Overview

jCleanCim is an open source tool for validation and documentation generation from Enterprise Architect CIM and IEC61850 UML models.

Quick intro to jCleanCim

Note: The content below is hard to keep up to date. Please refer to the jCleanCim presentation, available in the documentation directory.

This is jCleanCim open source application, initially developed to perform validation of CIM EA model file, then extended to do clean-up of left-overs from Rose and to show some basic statistics (thus addressing combined CIM issue #1103). Finally, it has been extended to allow for IEC (and custom) document generation, mainly driven by special needs of generating IEC 61850-7-3 and IEC 61850-7-4 from UML model developed by ABB and handed over to IEC TC57 WG19 in October 2009. On the fly, the support for CIM-based IEC documents (IEC 61970-301 and IEC 61968-11) has been added - after all, this was an easy part :-).

Since February 2010, jCleanCim has been used by CIM editors of IEC TC57 WG13, WG14 and recently WG16 to generate IEC 61970-301, IEC 61968-11 and IEC 62325-301 documents, respectively, as well as for the documentation describing CIM extensions developed in the European FP7 project ADDRESS.

Since September 2010, it has also been used by 61850 UML taskforce of WG10 to validate IEC 61850 UML and to automatically generate IEC 61850-7-4 and IEC 61850-7-3. As other IEC working groups start moving their IEC 61850-based specifications to UML as master model, jCleanCim will allow for automatic document generation for these, as well. Furthermore, since 01v06, the tool supports serialisation of the UML model of IEC 61850 to an XML format, that would be suitable for publishing on the web, and in support of the WebAccess taskforce of WG10. As a by-product, the same feature is available for CIM UML model.

NOTE: From this point on, the documentation is likely obsolete; we try to maintain up to date the jCleanCim presentation, so have a look there.

Main functions

jCleanCim has four main functions:

1. validation of a UML model provided in an .eap file, bulk or per IEC TC57 WG,
2. calculation and printing of statistics of the UML model,
3. generation of MS Word documentation from the UML model,
4. generation of XML documentation from the UML model.

In addition to support for the standard IEC TC57 CIM and custom CIM extensions compliant with CIM desing rules, jCleanCim supports the same functionality for the UML model of IEC 61850 being developed by the 61850 UML taskforce. Having both families of standards in UML is expected to facilitate one day harmonisation efforts between CIM and 61850, lead by IEC TC57 WG19. Since the IEC 61850 *UML model* is still not a standard, in the following we refer to CIM only, although the descriptions apply to the UML model of IEC 61850 as well.

Intended users

(TODO: needs update)

Intended users are primarily those who edit CIM UML and publish its documentation, thus:

- official IEC CIM model editors, responsible for maintaining the CIM information model (UML) and for generating official IEC documents, and,
- those who define custom non-standard CIM extensions who want to ensure they have followed standard CIM rules and who want to generate documentation for those extensions.

(continued on next page)

(continued from last page)

If you are already a user of the excellent CIMTool, you may wonder where within the process of CIM development and maintenance the jCleanCim fits. The answer is:

1. You would first use jCleanCim to validate correctness of the CIM information model (UML), and if required, to generate the information model documentation in MS Word format, as required by the IEC process.
2. You would then use CIMTool to create CIM profiles (XSD, RDF, OWL) and their documentation (HTML) from the imported CIM UML model, and to validate instance files created based on those profiles.

Available distributions

See readme file.

Prerequisites

See readme file.

Installation

See readme file.

Sample files

Project's input directory contains a very small model file (you need at least one model file to run jCleanCim at all), and a sample template file (required for doc generation only).

base-small.eap is a tiny subset of IEC 61970 UML plus home-made dumb extensions, with intentionally-introduced several modelling bugs and constructs never supposed to be found in CIM - this is for testing purposes. You can use base-small-template.doc and tailor it for your needs to produce the MS Word documentation.

Tailoring the template for your needs means modifying, adding or removing different kinds of `Placeholder`s.

If you want to run jCleanCim with your own models (and potentially template files), you need to copy them to the input directory.

This directory also contains a blank image file that must be available when running jCleanCim for certain scenarios (e.g., if you don't generate documentation, but only run validation, this file will be used in place of real UML diagrams).

Running jCleanCim out of the box

Binary distribution

After you have unzipped the **eclipse-independent binary** distribution (jCleanCim-[version]-bin.zip), you will be using the run script `run.bat` from console.

1. In the Windows Start menu, select "Run..." and type `cmd`. This will open a console window.
2. In the console window, type `cd` followed by a space. From the file explorer, just drag and drop the directory where you have unzipped the binary distribution onto the console window (so you don't have to type the whole path), then press enter. This will change the directory to where your jCleanCim has been installed.
3. In the console window, type `run` and press enter. This will run the jCleanCim jar with default configuration (validation and statistics) and the provided example files.

(continued on next page)

(continued from last page)

Source distribution

After you have unzipped the **eclipse-independent source** distribution (jCleanCim-[version]-src.zip) and installed Apache ant, you will be using the Apache ant script `build.xml` from console.

1. In the Windows Start menu, select "Run..." and type `cmd`. This will open a console window.
2. In the console window, type `cd` followed by a space. From the file explorer, just drag and drop the directory where you have unzipped the source distribution onto the console window (so you don't have to type the whole path), then press enter. This will change the directory to where your jCleanCim has been installed.
3. In the console window, type `ant jCleanCim` and press enter. This will build the jCleanCim jar from sources and run it with the default configuration (validation and statistics) and the provided example files. To see all available ant targets contained in the ant build file, type `ant -p` (or `ant -projecthelp`) and have a look at the graph of targets and their dependencies.

Alternatively, you may want to unzip the directory and import the project into your eclipse installation. *Note: If you already have an earlier version of jCleanCim in your eclipse workspace, you will first have to rename old jCleanCim project before importing new one. For example, if you want to import new version 01v04, rename first your existing jCleanCim project to jCleanCim-01v03, then import the new one. If you want to keep the old project jCleanCim-01v03 and use it, you will also have to update the build path for the directory where dll-s reside, otherwise eclipse shows you classpath error (eclipse does not use relative paths for dlls directories!).*

After you have imported the **eclipse existing project** from unzipped source distribution (jCleanCim-[version]-src.zip), you should create the default eclipse launch configuration as follows:

- Navigate to `src` directory
- locate the source file `org.tanjakostic.jcleancim.JCleanCim`
- right click on the class and select "Run as / Java Application".

This will launch the application and also create the run configuration, that you can later on copy to create custom configurations. Cached launch configurations are available from the eclipse "Run" icon (green icon with white arrow).

Note that you can also open the ant build file from within eclipse (Window / Show Views / ant) and run any of its tasks from the eclipse Outline window, the same way as from the console.

Configuring jCleanCim

Application configuration

To configure any run of jCleanCim application, you use the standard Java properties file available in the project's config directory. Default name for that properties file is config.properties and you can override this default with a command line argument if you want to use different stable configurations for different jCleanCim runs - see documentation in the application class `JCleanCim` and the configuration class `Config`.

The supplied default properties file contains reasonable defaults, and several tested configurations are commented. By default (out of the box), jCleanCim will run validation and statistics on base-small.eap model file, and will *not* generate any documentation.

Logging configuration

Project's config directory contains also the logging configuration file log4j.xml. We have set up the console output level to INFO (within the element `appender name="CONSOLE"...`), and the level for everything else to the most verbose, TRACE (within the element `logger name="org.tanjakostic.jcleancim"`). The first time you run jCleanCim application, the project's log directory gets created automatically.

If you get too noisy log files, you can dicrease the jCleanCim logger level in the log4j.xml file, from TRACE to DEBUG. If you want to post the jCleanCim log file with a model release, you can either:

(continued on next page)

(continued from last page)

- Copy the console output to a jCleanCim-[version]-[UMLpackageVersions].log file - this is easy when running jCleanCim from within eclipse, but may be impossible if running from console window.
- First dicrease the log level for the jCleanCim logger in log4j.xml file, from TRACE to INFO, then remove old log files, run jCleanCim and save the produced jCleanCim.log file as jCleanCim-[version]-[UMLpackageVersions].log; then revert the log level change in the log4j.xml file.

Typical usage patterns

UML model validation and statistics

To select what functions of jCleanCim to run, you need to set one or more of the main properties in the config/config.properties file: `validation.on` , `statistics.on` and `docgen.on` . In every case, you have to provide a valid EA model file name to work with, in the property `model.filename` . That file is expected to be on the jCleanCim classpath, so the best is to put the file in the input directory which is already set to be on the classpath.

Typical usage will be to first enable validation and statistics mode after you have edited the model, then address the problems in the model, and revalidate before releasing. Here is an example of a minimum config.properties file to do that:

```
model.filename    = base-small.eap
validation.on     = true
statistics.on     = true

validation.scope  =
```

If you are validating IEC 61850 UML models, there are several other properties; see documentation in `Config`.

If you have a big model, with potentially parts that are informative/buggy, you may want to set a filter and perform initial validation of your changes for only some top-level packages. For instance, to validate only standard CIM packages IEC61970 and IEC61968, you would set the `validation.scope` property so:

```
validation.scope = WG13, WG14
```

and to validate only custom (non-IEC) extensions:

```
validation.scope = OTHER_CIM, OTHER_IEC61850
```

It is recommended to *validate the full content of the EA model (by leaving the value of `validation.scope` property empty) at least before releasing the model* , to ensure there are no cross-package issues. See classes in the package `org.tanjakostic.jcleancim.validation` for available validators and rules they fire - the names of classes should be descriptive enough.

MS Word documentation generation

If you want to generate IEC (or custom) MS Word documentation from the UML model, in addition to the model file name in config/config.properties you must provide the names for template (input) file, the resulting (output) file, and enable document generation by setting the property `docgen.on = true` .

The template file is a regular MS Word document (*not* Word template with .dot extension), in which you put placeholders to control what jCleanCim should pick from the UML model and print into MS Word document. Detailed description of available placeholders and their usage is provided in the `Placeholder` class, and the templates distributed with jCleanCim in the project's

(continued on next page)

(continued from last page)

input directory should serve you as examples (of what is correct and what is not). If using your own template, you should put it into that directory before running jCleanCim for document generation.

When generating documentation, jCleanCim does the following:

- copies your template file into the projects `output` directory, created automatically the first time you run the document generation,
- renames the copied file to the name given in the properties file, and
- fills that copy with the contents from the EA model in place of the placeholders.

You can safely run document generation several times with the same name of the output file, without overwriting existing output files - if the output file exists, jCleanCim will rename it by appending a system nanosecond time. The disadvantage is that you will need to delete those discarded files from the `output` directory from time to time, but at least nothing gets lost without your control.

You may want to disable validation and statistics when enabling document generation to have the console log focused on document generation only.

Because document generation takes pretty long, you will first want to ensure that the placeholders in your template are correct, without generating the full package content. Here the minimum configuration to do this for a CIM model (IEC61850 model needs more properties; see config/config.properties file):

```
model.filename = base-small.eap

docgen.on = true
profiles.docgen.on =
docgen.inTemplate = base-small-template.doc
docgen.outDocument = base-small.doc

docgen.analysePlaceholders = true
```

Running only placeholder analysis (`docgen.analysePlaceholders=true`) will still produce the output document, but without UML package contents (classes, attributes, etc.). More importantly, that half-baked output document will contain placeholder errors, if any - do text search for string "$ERROR".

After you have fixed the placeholders in the template, you can reset `docgen.analysePlaceholders` to empty string to generate the full documentation.

There are further options documented in `Config` and in properties file config/config.properties. Playing with the provided sample small model and template files will hopefully get you started.

Other considerations

When generating official IEC documentation, the template should contain the IEC styles (this is probably already the case with CDV or FDIS documents that you as editor already have). To prevent MS Word exceptions when generating non-IEC documentation for extensions, jCleanCim defines default MS Word styles as replacement for the IEC styles. So, for example, 'Caption' is used if 'FIGURE-title' and 'TABLE-title' are not present in the template, or 'Normal' is used if 'PARAGRAPH' is not present. Below is the code snippet of the static initialiser for `Style` for the full list of default mappings: first argument is IEC style name and the second is the MS Word default style:

(continued on next page)

(continued from last page)

```
para("PARAGRAPH", "Normal"),
fig("Picture", "Normal"),
figcapt("FIGURE-title", "Caption"),
tabcapt("TABLE-title", "Caption"),
tabhead("TABLE-col-heading", "Normal"),
tabcell("TABLE-cell", "Normal"),
h1("Heading 1", "Heading 1"),
h2("Heading 2", "Heading 2"),
h3("Heading 3", "Heading 3"),
h4("Heading 4", "Heading 4"),
h5("Heading 5", "Heading 5"),
h6("Heading 6", "Heading 6"),
h7("Heading 7", "Heading 7"),
h8("Heading 8", "Heading 8"),
h9("Heading 9", "Heading 9");
```

It is essential to use correct styles for paragraphs containing figure and table captions in the template, because jCleanCim must deduce the number of figures and tables already existing in the template to calculate on the fly the correct numbering for new figures and tables (when inserting/appending the documentation for the UML model elements and diagrams). If jCleanCim throws an exception during document generation, it is very likely that the MS Word threw exception due to wrong/inexisting/negative number for the figure or table caption. *Note: We cannot check those numbers from within the code, because the MS Word automation API does not provide reliable access to them. In the worst case, when we catch an exception from MS Word, we gracefully close the MS Word document and exit the MS Word application, before exiting jCleanCim.*

Document generation may take pretty long, depending on how many classes the UML model has. The reason is that MS Word updates its fields evey time there is a table or any numbered paragraph (heading or figure/table caption) added to the document. To make that time somewhat shorter, consider the following when editing the template:

- disable automatic spell checking in the styles 'PARAGRAPH', 'TABLE-cell' and 'Normal'
- disable overall change tracking

Since version 01v03, jCleanCim has the MS Word application run in background by default (which is faster than having the window visible and updating all the time).

Known issues

Doc generation obviously relies on MS Word automation API, accessed from within Java through Java-COM bridge (Jacob). With certain MS Word files (used as jCleanCim template), we encounter from time to time issues when invoking COM objects for unknown reason, and with undetermined patterns (= an absolute horror for a programmer!). Therefore, the Java implementation of the writer catches those COM exceptions, prints the stack trace and attempts to continue, so you get at least some of the desired output.

Here some known issues related to MS Word automation API:

- When you print a relatively large part of the UML model, you may get the Word pop-up window "memory insufficient. Do you want to continue?" several times. To prevent this, jCleanCim regularly invokes the COM method (`UndoClear`) in attempt to clear cache of the running Word instance, but this call sometimes fails for an unknown reason. Disabling change tracking, and spell checking in styles 'PARAGRAPH', 'TABLE-cell' and 'Normal' in the template document may help here.
- In most cases, despite the above COM exceptions (and those that follow), your generated Word document will be complete, even if it stays open. Just save it and see what is in it. In cases it does not work, try to create a fresh Word document as a jCleanCim template, and copy only necessary styles from the original template, disable change tracking and spell checking.
- We also suspect issues with localised versions of MS Word related to style definition. Michael Specht (OFFIS, Germany) reported COM exceptions with base-small-template.doc bundled in the input with JCleanCim-01v01. He also reported that installing English language pack for MS Office solves the problem.

(continued on next page)

(continued from last page)

If the problems persist, feel free to help fixing jCleanCim/Jacob/MS Word stuff.

If despite that we cannot find a workaround, we should convince IEC to allow us to generate our documents at least as HTML (or still better, as XML), to get rid of binary dependencies on MS Word.

Success stories

All that said, we have sucessfully generated:

- IEC 61970-301 since Ed.4 (base CIM14)
- IEC 61968-11 since Ed.1 (DCIM10), and
- IEC 62325-301 since CDV (market CIM01), and various EU profile documents

We have also demonstrated the generation of Ed.2 IEC 61850-7-4 and IEC 61850-7-3, but these auto-generated documents from UML are not the official IEC documents yet.

The requirements for generation of the above documents (and what will be needed for IEC 61850-7-2 and other IEC CIM and 61850 family of standards) are the highest priority ones at this moment.

Checklist for the developer that produces jCleanCim distributions

Once after you've fixed bugs or added new features to jCleanCim, follow these steps to build and publish the three jCleanCim distributions:

1. update release notes (including date and version) in the readme file.
2. in the version properties file, update the property `project.version`.
3. if you provide new libraries or upgraded versions, ensure you update appropriate ant properties.
4. clean-up local config.properties file (keep only public defaults).
5. run `ant clean, unzip-all` and verify that the content unzipped under the build directory is ok.
6. run jCleanCim from within both unzipped directories (to verify it actually runs as described)
7. (copy locally the content of dist directory into jCleanCim releases directory)
8. on CIMug SharePoint, create new sub-directory within jCleanCim directory under CIM Methods & Tools for Enterprise Integration group Shared Documents; call it `jCleanCim-[version]`
9. upload into that new sub-directory all artefacts from dist directory
10. notify CIM model managers, IEC61850 UML task force and known users

Package
org.gigipugni.jcleancim.mibgen

org.gigipugni.jcleancim.mibgen
Class MibGen

```
java.lang.Object
  |
  +-org.gigipugni.jcleancim.mibgen.MibGen
```

public class **MibGen**
extends java.lang.Object

Constructor Summary

public	MibGen(UmlModel model) Constructs the collector from the UML model.

Method Summary

void	collectMib(UmlModel model, boolean liteMib)
DocgenConfig	getDocgenCfg()
boolean	isFromUml()

Methods inherited from class `java.lang.Object`

`clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait`

Constructors

MibGen

`public MibGen(UmlModel model)`

Constructs the collector from the UML model. After construction, call TODO to obtain the input for document generation.

Parameters:
`model`

Methods

isFromUml

`public boolean isFromUml()`

(continued on next page)

(continued from last page)

getDocgenCfg

```
public DocgenConfig getDocgenCfg()
```

collectMib

```
public void collectMib(UmlModel model,
         boolean liteMib)
```

org.gigipugni.jcleancim.mibgen
Class MibTemplate

```
java.lang.Object
  |
  +-org.gigipugni.jcleancim.mibgen.MibTemplate
```

public class **MibTemplate**
extends java.lang.Object

Constructor Summary

public	MibTemplate()

Method Summary

java.lang.String	getHeader()
java.lang.String	getModuleCompliance()
java.lang.String	getObject()
java.lang.String	getObjectGroup()
java.lang.String	getObjectIdentity()
java.lang.String	getTable()
java.lang.String	getTrap()
java.lang.String	getType()

Methods inherited from class `java.lang.Object`
clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Constructors

MibTemplate

public **MibTemplate**()

Methods

(continued from last page)

getHeader

```
public java.lang.String getHeader()
```

getObject

```
public java.lang.String getObject()
```

getTable

```
public java.lang.String getTable()
```

getType

```
public java.lang.String getType()
```

getTrap

```
public java.lang.String getTrap()
```

getObjectIdentity

```
public java.lang.String getObjectIdentity()
```

getObjectGroup

```
public java.lang.String getObjectGroup()
```

getModuleCompliance

```
public java.lang.String getModuleCompliance()
```

org.gigipugni.jcleancim.mibgen
Class MibWriter

```
java.lang.Object
  |
  +-org.gigipugni.jcleancim.mibgen.MibWriter
```

public class **MibWriter**
extends java.lang.Object

Constructor Summary

public	MibWriter(Config cfg, boolean lite, MibTemplate mibTemplate)

Method Summary

void	closeMib(int branchId)
void	closeMibFile()
java.lang.String	getCurrentMibIdentity()
void	resetBranchId()
void	resetRootPackages()
void	writeClassObjectIdentity(java.lang.String name, UmlClass c)
void	writeModuleCompliance(java.lang.String name, java.lang.String status, java.lang.String description, java.lang.String branch, java.lang.String branchid, java.lang.String groups)
void	writeModuleHeader(java.lang.String mibName, java.lang.String moduleName, java.lang.String description, java.lang.String branch, java.lang.String branchid, java.lang.String enums)
void	writeObject(java.lang.String mibprefix, java.lang.String branch, UmlAttribute a)
void	writeObjectGroup(java.lang.String name, java.lang.String status, java.lang.String description, java.lang.String branch, java.lang.String branchid, java.util.List objectList)
void	writeObjectIdentifier(java.lang.String name, java.lang.String branch, java.lang.String branchid)
void	writePackageObjectIdentity(java.lang.String name, UmlPackage p, java.lang.String branchid)

void	writeTable(java.lang.String name, java.lang.String type, java.lang.String access, java.lang.String status, java.lang.String description, java.lang.String branch, java.lang.String index, java.lang.String tableEntrySequence)
void	writeTrap(java.lang.String name, java.lang.String status, java.lang.String description, java.lang.String branch, java.lang.String enumerations)
void	writeType(java.lang.String name, java.lang.String type, java.lang.String status, java.lang.String description, java.lang.String enumerations)

Methods inherited from class `java.lang.Object`
`clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait`

Constructors

MibWriter

```
public MibWriter(Config cfg,
                 boolean lite,
                 MibTemplate mibTemplate)
```

Methods

writeTrap

```
public void writeTrap(java.lang.String name,
         java.lang.String status,
         java.lang.String description,
         java.lang.String branch,
         java.lang.String enumerations)
```

writeModuleCompliance

```
public void writeModuleCompliance(java.lang.String name,
         java.lang.String status,
         java.lang.String description,
         java.lang.String branch,
         java.lang.String branchid,
         java.lang.String groups)
```

(continued from last page)

writeObjectGroup

```
public void writeObjectGroup(java.lang.String name,
         java.lang.String status,
         java.lang.String description,
         java.lang.String branch,
         java.lang.String branchid,
         java.util.List objectList)
```

writeType

```
public void writeType(java.lang.String name,
         java.lang.String type,
         java.lang.String status,
         java.lang.String description,
         java.lang.String enumerations)
```

writeTable

```
public void writeTable(java.lang.String name,
         java.lang.String type,
         java.lang.String access,
         java.lang.String status,
         java.lang.String description,
         java.lang.String branch,
         java.lang.String index,
         java.lang.String tableEntrySequence)
```

writeObject

```
public void writeObject(java.lang.String mibprefix,
         java.lang.String branch,
         UmlAttribute a)
```

writeModuleHeader

```
public void writeModuleHeader(java.lang.String mibName,
         java.lang.String moduleName,
         java.lang.String description,
         java.lang.String branch,
         java.lang.String branchid,
         java.lang.String enums)
```

writeClassObjectIdentity

```
public void writeClassObjectIdentity(java.lang.String name,
         UmlClass c)
```

(continued from last page)

writePackageObjectIdentity

```
public void writePackageObjectIdentity(java.lang.String name,
         UmlPackage p,
         java.lang.String branchid)
```

resetRootPackages

```
public void resetRootPackages()
```

writeObjectIdentifier

```
public void writeObjectIdentifier(java.lang.String name,
         java.lang.String branch,
         java.lang.String branchid)
```

resetBranchId

```
public void resetBranchId()
```

closeMib

```
public void closeMib(int branchId)
```

closeMibFile

```
public void closeMibFile()
```

getCurrentMibIdentity

```
public java.lang.String getCurrentMibIdentity()
```

Package
org.tanjakostic.jcleancim

jCleanCim is an open source tool for validation and documentation generation from Enterprise Architect CIM and IEC61850 UML models.

org.tanjakostic.jcleancim
Class JCleanCim

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.JCleanCim
```

public class **JCleanCim**
extends java.lang.Object

jCleanCim command-line application.

Most of configuration currently needs to be specified in ./config/ org.tanjakostic.jcleancim.common.Config#DEFAULT_PROPS_FILE_NAME file. Command line arguments allow you to:

- change the name of that file, i.e., to specify different configurations for different runs with the same input model .eap, and/or
- override the input model .eap file specified in configuration properties file - i.e., to use the same configuration for different input models

We use apache command line argument library here as it gives nice help :-) If we need more configuration/filtering, best would be to do that in ./config/ org.tanjakostic.jcleancim.common.Config#DEFAULT_PROPS_FILE_NAME file and **not** with command line options (we could end up in a mess of what is defined on cmd line, and what in properties file).

Implementation note: To add new command line arguments, follow examples in the constructor. To add functionality for validation, statistics, profile crosschek and doc generation (from UML or from profiles), implement methods on `UmlModel` class and call them from `validate(UmlModel)`, `collectStatistics(UmlModel)`, `crossCheck(UmlModel, UmlModel)` and `generateDoc(UmlModel)` methods, respectively.

Method Summary

`UmlModel`	`buildFromEA()` Builds the model from EA file given in configuration or on command line.
`UmlModel`	`buildFromProfiles()` Builds the model from all profiles found under the /input/profiles directory.
`void`	`collectStatistics(UmlModel model)` Collects statistics for the model and logs them.
`UmlModel`	`createEmptyModel()` Creates empty model.
`void`	`crossCheck(UmlModel profilesModel, UmlModel umlModel)` Performs cross-check between the set of profiles and the UML model.
`void`	`generateDoc(UmlModel model)` Generates documentation for the model in the format specified in configuration through output file extension.
`void`	`generateMib(UmlModel model)` Generates full and light MIBs from the model.
`static void`	`main(java.lang.String[] args)` This command-line application first populates its model from full .eap file and/or profiles (to allow for different kinds of analysis afterwards), then selectively runs validation, statistics, profile-model cross-checking and MS Word document generation.

void	validate(UmlModel model) Validates the model.

Methods inherited from class java.lang.Object
clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods

main

```
public static void main(java.lang.String[] args)
  throws ApplicationException,
         java.io.IOException
```

This command-line application first populates its model from full .eap file and/or profiles (to allow for different kinds of analysis afterwards), then selectively runs validation, statistics, profile-model cross-checking and MS Word document generation. These operations can be enabled/disabled and they apply to the scope as configured in ./config/org.tanjakostic.jcleancim.common.Config#DEFAULT_PROPS_FILE_NAME file (or in a file you specify with `-propFile` command line option).

```
usage: jCleanCim
 -help                      print this message
 -version                   print application version
 -modelFile <*.eap>         name of the model file to use instead of one
                            defined in config properties
 -propFile <*.properties>   name of the config properties file to use
                            instead of default
```

Throws:
ApplicationException

buildFromEA

```
public UmlModel buildFromEA()
  throws ApplicationException
```

Builds the model from EA file given in configuration or on command line.

Throws:
ApplicationException

buildFromProfiles

```
public UmlModel buildFromProfiles()
  throws ApplicationException
```

Builds the model from all profiles found under the /input/profiles directory.

Throws:

(continued from last page)

ApplicationException

createEmptyModel

```
public UmlModel createEmptyModel()
  throws ApplicationException
```

Creates empty model.

Throws:
ApplicationException

validate

```
public void validate(UmlModel model)
```

Validates the model.

Parameters:
model

collectStatistics

```
public void collectStatistics(UmlModel model)
```

Collects statistics for the model and logs them.

Parameters:
model

crossCheck

```
public void crossCheck(UmlModel profilesModel,
         UmlModel umlModel)
```

Performs cross-check between the set of profiles and the UML model.

Parameters:
profilesModel
umlModel

generateMib

```
public void generateMib(UmlModel model)
  throws ApplicationException,
         java.io.IOException
```

Generates full and light MIBs from the model.

Throws:
UnsupportedOutputFormatException - if the requested format (extension) of the output file is not supported.
IOException - on any file system-related problem.

generateDoc

```
public void generateDoc(UmlModel model)
  throws ApplicationException,
         java.io.IOException
```

(continued from last page)

Generates documentation for the model in the format specified in configuration through output file extension.

Throws:

`UnsupportedOutputFormatException` - if the requested format (extension) of the output file is not supported.
`IOException` - on any file system-related problem.

Package
org.tanjakostic.jcleancim.builder

org.tanjakostic.jcleancim.builder
Class AbstractDiagramExporter

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.builder.AbstractDiagramExporter
```

All Implemented Interfaces:
DiagramExporter

Direct Known Subclasses:
EmptyDiagramExporter

---

public abstract class **AbstractDiagramExporter**
extends java.lang.Object
implements DiagramExporter

Common default implementation, as for "empty exporter.

Constructor Summary

public	AbstractDiagramExporter(Config cfg) Constructor.

Method Summary

Config	getCfg()
java.io.File	saveToFile(UmlObjectBuilder dia, Util.ImageFormat format, boolean throughClipboard)

Methods inherited from class java.lang.Object
clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.builder.DiagramExporter
getCfg, saveToFile

Constructors

AbstractDiagramExporter

public **AbstractDiagramExporter**(Config cfg)

Constructor.

Methods

(continued from last page)

getCfg

```
public final Config getCfg()
```

saveToFile

```
public java.io.File saveToFile(UmlObjectBuilder dia,
         Util.ImageFormat format,
         boolean throughClipboard)
  throws java.io.IOException
```

This default implementation always returns null, without actually exporting anything; ensure to override if you can export diagrams.

org.tanjakostic.jcleancim.builder
Class AbstractModelBuilder

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.builder.AbstractModelBuilder
```

All Implemented Interfaces:
ModelBuilder

Direct Known Subclasses:
ModelBuilderFromProfiles, EaModelBuilder, EmptyModelBuilder

---

public abstract class **AbstractModelBuilder**
extends java.lang.Object
implements ModelBuilder

Constructor Summary

protected	AbstractModelBuilder(Config cfg) Constructor.

Method Summary

abstract UmlModel	build()
abstract DiagramExporter	createDiagramExporter() Creates exporter of UML diagrams where applicable (otherwise, can be just a stub).
abstract XMIExporter	createXMIExporter() Returns exporter to XMI where applicable (otherwise, can be just a stub).
Config	getCfg()
DiagramExporter	getDiagramExporter()
XMIExporter	getXMIExporter()

Methods inherited from class `java.lang.Object`
clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface `org.tanjakostic.jcleancim.builder.ModelBuilder`
build, getCfg, getDiagramExporter, getXMIExporter

Constructors

(continued from last page)

AbstractModelBuilder

```
protected AbstractModelBuilder(Config cfg)
```

Constructor.

Methods

getDiagramExporter

```
public final DiagramExporter getDiagramExporter()
```

getXMIExporter

```
public final XMIExporter getXMIExporter()
```

createDiagramExporter

```
protected abstract DiagramExporter createDiagramExporter()
```

Creates exporter of UML diagrams where applicable (otherwise, can be just a stub).

createXMIExporter

```
protected abstract XMIExporter createXMIExporter()
```

Returns exporter to XMI where applicable (otherwise, can be just a stub).

getCfg

```
public final Config getCfg()
```

build

```
public abstract UmlModel build()
  throws ApplicationException
```

org.tanjakostic.jcleancim.builder
Class AbstractXMIExporter

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.builder.AbstractXMIExporter
```

All Implemented Interfaces:
XMIExporter

Direct Known Subclasses:
EmptyXMIExporter

---

public abstract class **AbstractXMIExporter**
extends java.lang.Object
implements XMIExporter

Here we implement all the logic except for actual exporting.

Constructor Summary

protected	AbstractXMIExporter(Config cfg) Constructor.

Method Summary

void	exportToXMIs(java.lang.String rootUuid)
Config	getCfg()
abstract void	toXmi(java.lang.String rootUuid, XMIDialect dialect, boolean exportDiagrams, java.io.File file, java.lang.String detail) Actual export to XMI.

Methods inherited from class `java.lang.Object`
clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface `org.tanjakostic.jcleancim.builder.XMIExporter`
exportToXMIs, getCfg

Constructors

AbstractXMIExporter

protected **AbstractXMIExporter**(Config cfg)

Constructor.

(continued from last page)

Methods

getCfg

```
public final Config getCfg()
```

exportToXMIs

```
public final void exportToXMIs(java.lang.String rootUuid)
  throws ApplicationException
```

toXmi

```
protected abstract void toXmi(java.lang.String rootUuid,
         XMIDialect dialect,
         boolean exportDiagrams,
         java.io.File file,
         java.lang.String detail)
  throws ApplicationException
```

Actual export to XMI.

Parameters:

`rootUuid`
`dialect`
`exportDiagrams`
`file`
`detail`

Throws:

`ApplicationException`

org.tanjakostic.jcleancim.builder
Interface DiagramExporter

All Known Implementing Classes:
AbstractDiagramExporter

public interface **DiagramExporter**
extends

Method Summary

abstract Config	getCfg() Returns the configuration, containing also diagram export options.
abstract java.io.File	saveToFile(UmlObjectBuilder dia, Util.ImageFormat format, boolean throughClipboard) Copies diagram dia to a file with format, in the pics output directory (per configuration) and returns the created file.

Methods

getCfg

```
public abstract Config getCfg()
```

Returns the configuration, containing also diagram export options.

saveToFile

```
public abstract java.io.File saveToFile(UmlObjectBuilder dia,
         Util.ImageFormat format,
         boolean throughClipboard)
  throws java.io.IOException
```

Copies diagram `dia` to a file with `format`, in the pics output directory (per configuration) and returns the created file.

Parameters:
`dia` - diagram representation
`format` - image format
`throughClipboard` - if true, image will be copied to clipboard and then saved as bitmap file (i.e., `format` will be ignored).

Returns:
created file with the diagram; null if file creation failed.

Throws:
`IOException` - if file creation failed.

org.tanjakostic.jcleancim.builder
Class EmptyDiagramExporter

```
java.lang.Object
  |-org.tanjakostic.jcleancim.builder.AbstractDiagramExporter
     |-org.tanjakostic.jcleancim.builder.EmptyDiagramExporter
```

All Implemented Interfaces:
DiagramExporter

---

public class **EmptyDiagramExporter**
extends AbstractDiagramExporter

Constructor Summary

public	EmptyDiagramExporter(Config cfg)

Methods inherited from class org.tanjakostic.jcleancim.builder.AbstractDiagramExporter
getCfg, saveToFile

Methods inherited from class java.lang.Object
clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.builder.DiagramExporter
getCfg, saveToFile

Constructors

EmptyDiagramExporter

public **EmptyDiagramExporter**(Config cfg)

org.tanjakostic.jcleancim.builder
Class EmptyModelBuilder

```
java.lang.Object
  +-org.tanjakostic.jcleancim.builder.AbstractModelBuilder
    +-org.tanjakostic.jcleancim.builder.EmptyModelBuilder
```

All Implemented Interfaces:
ModelBuilder

---

public class **EmptyModelBuilder**
extends AbstractModelBuilder

Creates an empty model, only with configuration. Useful when we want to generate documentation for profiles, where we don't need to read in the UML model from a repository.

Constructor Summary

public	EmptyModelBuilder(Config cfg)

Method Summary

UmlModel	build()
DiagramExporter	createDiagramExporter()
XMIExporter	createXMIExporter()

Methods inherited from class org.tanjakostic.jcleancim.builder.AbstractModelBuilder

build, createDiagramExporter, createXMIExporter, getCfg, getDiagramExporter, getXMIExporter

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.builder.ModelBuilder

build, getCfg, getDiagramExporter, getXMIExporter

Constructors

EmptyModelBuilder

public **EmptyModelBuilder**(Config cfg)

Methods

build

`public UmlModel build()`

createDiagramExporter

`protected DiagramExporter createDiagramExporter()`

Creates exporter of UML diagrams where applicable (otherwise, can be just a stub).

createXMIExporter

`protected XMIExporter createXMIExporter()`

Returns exporter to XMI where applicable (otherwise, can be just a stub).

org.tanjakostic.jcleancim.builder
Class EmptyXMIExporter

```
java.lang.Object
  |-org.tanjakostic.jcleancim.builder.AbstractXMIExporter
    |-org.tanjakostic.jcleancim.builder.EmptyXMIExporter
```

All Implemented Interfaces:
XMIExporter

public class **EmptyXMIExporter**
extends AbstractXMIExporter

This one doesn't know how to export XMI; useful when working without EA repository.

Constructor Summary

public	EmptyXMIExporter(Config cfg)

Method Summary

void	toXmi(java.lang.String rootUuid, XMIDialect dialect, boolean exportDiagrams, java.io.File file, java.lang.String detail) This default implementation does nothing (well, that's why we are empty exporter).

Methods inherited from class org.tanjakostic.jcleancim.builder.AbstractXMIExporter
exportToXMIs, getCfg, toXmi

Methods inherited from class java.lang.Object
clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.builder.XMIExporter
exportToXMIs, getCfg

Constructors

EmptyXMIExporter

public **EmptyXMIExporter**(Config cfg)

Methods

(continued from last page)

toXmi

```
protected void toXmi(java.lang.String rootUuid,
         XMIDialect dialect,
         boolean exportDiagrams,
         java.io.File file,
         java.lang.String detail)
```

This default implementation does nothing (well, that's why we are empty exporter).

org.tanjakostic.jcleancim.builder
Interface ModelBuilder

All Known Implementing Classes:
AbstractModelBuilder

public interface **ModelBuilder**
extends

Builds an in-memory representation of the model.

Method Summary

abstract UmlModel	build() Builds the in-memory model.
abstract Config	getCfg() Returns configuration.
abstract DiagramExporter	getDiagramExporter()
abstract XMIExporter	getXMIExporter()

Methods

getCfg

```
public abstract Config getCfg()
```

Returns configuration.

build

```
public abstract UmlModel build()
  throws ApplicationException
```

Builds the in-memory model. It is expected to release any resources it may have acquired.

getDiagramExporter

```
public abstract DiagramExporter getDiagramExporter()
```

getXMIExporter

```
public abstract XMIExporter getXMIExporter()
```

org.tanjakostic.jcleancim.builder
Interface UmlObjectBuilder

public interface **UmlObjectBuilder**
extends

To avoid interface bloat, we follow the design pattern of Java collections API: to provide "optional" methods and let implementations select which one they implement.

FIXME: doc

Method Summary

abstract UmlObject	build()
abstract UmlObject	build(UmlModel model)
abstract UmlObjectData	getObjData()

Methods

build

```
public abstract UmlObject build()
  throws java.lang.UnsupportedOperationException
```

Returns:
FIXME

build

```
public abstract UmlObject build(UmlModel model)
  throws java.lang.UnsupportedOperationException
```

Parameters:
`model`

Returns:
FIXME

getObjData

```
public abstract UmlObjectData getObjData()
```

Returns:

(continued from last page)

FIXME

org.tanjakostic.jcleancim.builder
Interface XMIExporter

All Known Implementing Classes:
AbstractXMIExporter

public interface **XMIExporter**
extends

Interface to implement for XMI export capability.

Method Summary

abstract void	exportToXMIs(java.lang.String rootUuid) Actually performs export for all the configured dialects.
abstract Config	getCfg() Returns the configuration, containing also XMI export options.

Methods

getCfg

```
public abstract Config getCfg()
```

Returns the configuration, containing also XMI export options.

exportToXMIs

```
public abstract void exportToXMIs(java.lang.String rootUuid)
  throws ApplicationException
```

Actually performs export for all the configured dialects.

Package
org.tanjakostic.jcleancim.builder.ea

Classes responsible for building in-memory UML model from EA repository (.eap model file).

The classes in this package have been factored out of the initial, simpler but less flexible implementation (in which these classes were initialising themselves from the EA repository and were used further by application for everything). These builder classes now are the only ones that "talk" to the EA repository, through a terribly slow EA API or, since 01v07 through bulk SQL queries, and they cache all the data we are interested in for a UML model. After they fetch all the data and diagrams from the EA repository, and potentially export diagrams for document generation, or export XMI, they create (or "build") a simple in-memory UML model that the application then uses for everything else. From that moment on, the application is totally independent of the EA repository, as it works with the in-memory UML model.

Important classes and interfaces are:

- `ModelBuilder` - interface implemented by model builders from various model sources.
- `EaHelper` - interface defining methods that rely on EA repository or project objects, such as copying diagrams to system clipboard or saving them to files, or the formatted documentation of the UML elements in EA repository.
- All the other *Builder classes.

org.tanjakostic.jcleancim.builder.ea
Class AssociationBuilder

```
java.lang.Object
  |-org.tanjakostic.jcleancim.builder.ea.AbstractObjectBuilderFromEA
    |-org.tanjakostic.jcleancim.builder.ea.AssociationBuilder
```

All Implemented Interfaces:
UmlObjectBuilder

---

public abstract class **AssociationBuilder**
extends AbstractObjectBuilderFromEA

Parameters:
`O` - Source data for association, `T` - Source data for association tagged values

---

Fields inherited from class `org.tanjakostic.jcleancim.builder.ea.AbstractObjectBuilderFromEA`
`CTOR_LOG_LEVEL`

Constructor Summary

protected	`AssociationBuilder(java.lang.Object inData, java.lang.Object tagsSrc, ClassBuilder source, ClassBuilder target, EaModelBuilder model, EaHelper eaHelper)` Constructor.

Method Summary

abstract AssociationEndBuilder	`createAssociationEnd(java.lang.Object inData, java.lang.Object tagsSrc, boolean isSource, ClassBuilder type, EaHelper eaHelper)`
void	`doBuild()`
void	`ensureAssociationsOfEndClassesInitialised()` Model builder should call this method to cross-check initialisation is correct.
abstract java.util.List	`fetchTaggedValues(java.lang.Object inDataTags)`
abstract java.lang.String	`getConnectorAlias(java.lang.Object inData)`
abstract java.lang.String	`getConnectorDirection(java.lang.Object inData)`
abstract java.lang.String	`getConnectorGUID(java.lang.Object inData)`
abstract java.lang.Integer	`getConnectorID(java.lang.Object inData)`
abstract java.lang.String	`getConnectorName(java.lang.Object inData)`

abstract java.lang.String	getConnectorNotes(java.lang.Object inData)
abstract java.lang.String	getConnectorStereotypes(java.lang.Object inData)
java.lang.String	getDirection()
UmlObjectData	getObjData()
AssociationEndBuilder	getSourceEnd()
java.util.Map	getTaggedValues()
AssociationEndBuilder	getTargetEnd()
void	initTaggedValues(java.util.List myTaggedValuesFields)
static boolean	isAssociationOrAggregation(java.lang.String type) Returns whether the EA connector is either an association or an aggregation (and thus needs to be retained for processing).
boolean	isBiDirectional()
boolean	isDirectionUnspecified()
java.lang.String	toString()

Methods inherited from class org.tanjakostic.jcleancim.builder.ea.AbstractObjectBuilderFromEA
build, build, doBuild, doBuild, getResult, setResult

Methods inherited from class java.lang.Object
clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.builder.UmlObjectBuilder
build, build, getObjData

Constructors

AssociationBuilder

```
protected AssociationBuilder(java.lang.Object inData,
                             java.lang.Object tagsSrc,
                             ClassBuilder source,
                             ClassBuilder target,
                             EaModelBuilder model,
                             EaHelper eaHelper)
```

Constructor.

(continued from last page)

Parameters:

`inData`
`tagsSrc`
`source`
`target`
`model`
`eaHelper`

Throws:

`NullPointerException` - if both source and target are null.

Methods

isAssociationOrAggregation

`public static boolean` **`isAssociationOrAggregation`**`(java.lang.String type)`

Returns whether the EA connector is either an association or an aggregation (and thus needs to be retained for processing).

getConnectorID

`protected abstract java.lang.Integer` **`getConnectorID`**`(java.lang.Object inData)`

getConnectorGUID

`protected abstract java.lang.String` **`getConnectorGUID`**`(java.lang.Object inData)`

getConnectorName

`protected abstract java.lang.String` **`getConnectorName`**`(java.lang.Object inData)`

getConnectorAlias

`protected abstract java.lang.String` **`getConnectorAlias`**`(java.lang.Object inData)`

getConnectorStereotypes

`protected abstract java.lang.String` **`getConnectorStereotypes`**`(java.lang.Object inData)`

getConnectorNotes

`protected abstract java.lang.String` **`getConnectorNotes`**`(java.lang.Object inData)`

getConnectorDirection

`protected abstract java.lang.String` **`getConnectorDirection`**`(java.lang.Object inData)`

(continued from last page)

createAssociationEnd

```
protected abstract AssociationEndBuilder createAssociationEnd(java.lang.Object inData,
         java.lang.Object tagsSrc,
         boolean isSource,
         ClassBuilder type,
         EaHelper eaHelper)
```

getSourceEnd

```
public final AssociationEndBuilder getSourceEnd()
```

getTargetEnd

```
public final AssociationEndBuilder getTargetEnd()
```

getDirection

```
public final java.lang.String getDirection()
```

isDirectionUnspecified

```
public final boolean isDirectionUnspecified()
```

isBiDirectional

```
public final boolean isBiDirectional()
```

fetchTaggedValues

```
protected abstract java.util.List fetchTaggedValues(java.lang.Object inDataTags)
```

initTaggedValues

```
protected final void initTaggedValues(java.util.List myTaggedValuesFields)
```

getTaggedValues

```
public final java.util.Map getTaggedValues()
```

(continued from last page)

ensureAssociationsOfEndClassesInitialised

`public final void ensureAssociationsOfEndClassesInitialised()`

Model builder should call this method to cross-check initialisation is correct.

toString

`public java.lang.String toString()`

getObjData

`public final UmlObjectData getObjData()`

doBuild

`public final void doBuild()`

org.tanjakostic.jcleancim.builder.ea
Class AssociationEndBuilder

```
java.lang.Object
  +-org.tanjakostic.jcleancim.builder.ea.AbstractObjectBuilderFromEA
    +-org.tanjakostic.jcleancim.builder.ea.AssociationEndBuilder
```

All Implemented Interfaces:
UmlObjectBuilder

public abstract class **AssociationEndBuilder**
extends AbstractObjectBuilderFromEA

Parameters:
`O` - Source data for association end, `T` - Source data for association end tagged values

Fields inherited from class `org.tanjakostic.jcleancim.builder.ea.AbstractObjectBuilderFromEA`
`CTOR_LOG_LEVEL`

Constructor Summary

protected	AssociationEndBuilder(java.lang.Object inData, java.lang.Object tagsSrc, AssociationBuilder containingAssociation, boolean isSource, ClassBuilder type, EaHelper eaHelper) Constructs association end from EA object.

Method Summary

void	doBuild()
abstract java.util.List	fetchTaggedValues(java.lang.Object srcTags)
AssociationBuilder	getContainingAssociation()
UmlAssociationEnd.Kind	getKind()
UmlMultiplicity	getMultiplicity()
UmlAssociationEnd.Navigable	getNavigable()
UmlObjectData	getObjData()
abstract java.lang.String	getRoleAggregation(java.lang.Object inData)
abstract java.lang.String	getRoleAlias(java.lang.Object inData)
abstract java.lang.String	getRoleCardinality(java.lang.Object inData)

abstract java.lang.String	getRoleName(java.lang.Object inData)
abstract java.lang.String	getRoleNavigable(java.lang.Object inData)
abstract java.lang.String	getRoleNotes(java.lang.Object inData)
abstract java.lang.String	getRoleStereotypes(java.lang.Object inData)
abstract java.lang.String	getRoleVisibility(java.lang.Object inData)
java.util.Map	getTaggedValues()
ClassBuilder	getType()
void	initObjData(java.lang.String name, java.lang.String alias, java.lang.String stereotype, java.lang.String visibility, java.lang.String notes, EaHelper eaHelper)
void	initOwnData(java.lang.String kind, java.lang.String cardinality, java.lang.String direction)
boolean	isAggregation()
boolean	isAssociation()
boolean	isComposition()
boolean	isOther()
boolean	isSource()
void	setType(ClassBuilder type)
java.lang.String	toString()

Methods inherited from class org.tanjakostic.jcleancim.builder.ea.AbstractObjectBuilderFromEA
build, build, doBuild, doBuild, getResult, setResult

Methods inherited from class java.lang.Object
clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.builder.UmlObjectBuilder
build, build, getObjData

(continued from last page)

Constructors

AssociationEndBuilder

```
protected AssociationEndBuilder(java.lang.Object inData,
                                java.lang.Object tagsSrc,
                                AssociationBuilder containingAssociation,
                                boolean isSource,
                                ClassBuilder type,
                                EaHelper eaHelper)
```

Constructs association end from EA object. Sets id to 0 and randomly generated uuid (these are not defined in EA).

If multiplicity for a composite end is empty, sets it to `UmlMultiplicity.OPT_ONE`, otherwise just uses whatever is found in the model.

Parameters:

`inData`
`tagsSrc`
`containingAssociation`
`type`
`eaHelper` - required for getting formatted doc

Throws:

`NullPointerException` - if any argument is null.

Methods

getRoleName

```
protected abstract java.lang.String getRoleName(java.lang.Object inData)
```

getRoleAlias

```
protected abstract java.lang.String getRoleAlias(java.lang.Object inData)
```

getRoleStereotypes

```
protected abstract java.lang.String getRoleStereotypes(java.lang.Object inData)
```

getRoleVisibility

```
protected abstract java.lang.String getRoleVisibility(java.lang.Object inData)
```

getRoleNotes

```
protected abstract java.lang.String getRoleNotes(java.lang.Object inData)
```

(continued from last page)

initObjData

```
protected final void initObjData(java.lang.String name,
         java.lang.String alias,
         java.lang.String stereotype,
         java.lang.String visibility,
         java.lang.String notes,
         EaHelper eaHelper)
```

getRoleAggregation

```
protected abstract java.lang.String getRoleAggregation(java.lang.Object inData)
```

getRoleCardinality

```
protected abstract java.lang.String getRoleCardinality(java.lang.Object inData)
```

getRoleNavigable

```
protected abstract java.lang.String getRoleNavigable(java.lang.Object inData)
```

initOwnData

```
protected final void initOwnData(java.lang.String kind,
         java.lang.String cardinality,
         java.lang.String direction)
```

fetchTaggedValues

```
protected abstract java.util.List fetchTaggedValues(java.lang.Object srcTags)
```

getTaggedValues

```
public final java.util.Map getTaggedValues()
```

isSource

```
public final boolean isSource()
```

getContainingAssociation

```
public final AssociationBuilder getContainingAssociation()
```

(continued from last page)

getKind

```
public final UmlAssociationEnd.Kind getKind()
```

getMultiplicity

```
public final UmlMultiplicity getMultiplicity()
```

getNavigable

```
public final UmlAssociationEnd.Navigable getNavigable()
```

setType

```
public final void setType(ClassBuilder type)
```

getType

```
public final ClassBuilder getType()
```

isAssociation

```
public final boolean isAssociation()
```

isAggregation

```
public final boolean isAggregation()
```

isComposition

```
public final boolean isComposition()
```

isOther

```
public final boolean isOther()
```

(continued from last page)

toString

```
public java.lang.String toString()
```

getObjData

```
public final UmlObjectData getObjData()
```

doBuild

```
protected final void doBuild()
```

org.tanjakostic.jcleancim.builder.ea
Class AttributeBuilder

```
java.lang.Object
  |-org.tanjakostic.jcleancim.builder.ea.AbstractObjectBuilderFromEA
    |-org.tanjakostic.jcleancim.builder.ea.AttributeBuilder
```

All Implemented Interfaces:
UmlObjectBuilder

---

public abstract class **AttributeBuilder**
extends AbstractObjectBuilderFromEA

It is the responsibility of the model builder to call `assignType(Map)` for every created attribute, after all the classes have been loaded.
Parameters:
`O` - Source data for attribute, `T` - Source data for attribute tagged values

---

Fields inherited from class `org.tanjakostic.jcleancim.builder.ea.AbstractObjectBuilderFromEA`
`CTOR_LOG_LEVEL`

Constructor Summary

protected	`AttributeBuilder(java.lang.Object inData, java.lang.Object tagsSrc, ClassBuilder containingClass, EaHelper eaHelper)` Constructor for attribute from EA object.

Method Summary

void	`assignType(java.util.Map classes)` It is the responsibility of the model builder to call this method after all the classes in the model have been initialised.
void	`doBuild()`
abstract java.util.List	`fetchAttrConstraints(java.lang.Object tagsSrc)`
abstract java.util.List	`fetchTaggedValues(java.lang.Object tagsSrc)`
abstract java.lang.String	`getAttributeAlias(java.lang.Object inData)`
abstract int	`getAttributeClassifierID(java.lang.Object inData)`
abstract java.lang.String	`getAttributeDefaultValue(java.lang.Object inData)`
abstract java.lang.String	`getAttributeGUID(java.lang.Object inData)`

abstract java.lang.Integer	getAttributeID(java.lang.Object inData)
abstract boolean	getAttributeIsConst(java.lang.Object inData)
abstract boolean	getAttributeIsStatic(java.lang.Object inData)
abstract java.lang.String	getAttributeLowerBound(java.lang.Object inData)
abstract java.lang.String	getAttributeName(java.lang.Object inData)
abstract java.lang.String	getAttributeNotes(java.lang.Object inData)
abstract int	getAttributePosition(java.lang.Object inData)
abstract java.lang.String	getAttributeStereotypes(java.lang.Object inData)
abstract java.lang.String	getAttributeType(java.lang.Object inData)
abstract java.lang.String	getAttributeUpperBound(java.lang.Object inData)
abstract java.lang.String	getAttributeVisibility(java.lang.Object inData)
java.util.List	getConstraints()
ClassBuilder	getContainingClass()
int	getEaTypeId()
java.lang.String	getEaTypeName()
java.lang.String	getInitValue()
UmlMultiplicity	getMultiplicity()
UmlObjectData	getObjData()
int	getPos()
java.util.Map	getTaggedValues()
ClassBuilder	getType()
void	initObjData(java.lang.Integer id, java.lang.String guid, java.lang.String name, java.lang.String alias, java.lang.String stereotype, java.lang.String visibility, java.lang.String notes, EaHelper eaHelper)

void	initOwnData(int pos, boolean isEnum, boolean isConst, boolean isStatic, java.lang.String lowerBound, java.lang.String upperBound, java.lang.String defaultVal, int classifierID, java.lang.String type)
boolean	isConst()
boolean	isLiteral()
boolean	isStatic()
boolean	isTypeSuperfluous()
java.lang.String	toString()

Methods inherited from class org.tanjakostic.jcleancim.builder.ea.AbstractObjectBuilderFromEA
build, build, doBuild, doBuild, getResult, setResult

Methods inherited from class java.lang.Object
clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.builder.UmlObjectBuilder
build, build, getObjData

Constructors

AttributeBuilder

```
protected AttributeBuilder(java.lang.Object inData,
                           java.lang.Object tagsSrc,
                           ClassBuilder containingClass,
                           EaHelper eaHelper)
```

Constructor for attribute from EA object.

Parameters:

inData
tagsSrc
containingClass
eaHelper

Throws:

NullPointerException - if containingClass, eaAttr or helper is null.

Methods

getAttributeID

```
protected abstract java.lang.Integer getAttributeID(java.lang.Object inData)
```

getAttributeGUID

```
protected abstract java.lang.String getAttributeGUID(java.lang.Object inData)
```

getAttributeName

```
protected abstract java.lang.String getAttributeName(java.lang.Object inData)
```

getAttributeAlias

```
protected abstract java.lang.String getAttributeAlias(java.lang.Object inData)
```

getAttributeStereotypes

```
protected abstract java.lang.String getAttributeStereotypes(java.lang.Object inData)
```

getAttributeVisibility

```
protected abstract java.lang.String getAttributeVisibility(java.lang.Object inData)
```

getAttributeNotes

```
protected abstract java.lang.String getAttributeNotes(java.lang.Object inData)
```

initObjData

```
protected final void initObjData(java.lang.Integer id,
         java.lang.String guid,
         java.lang.String name,
         java.lang.String alias,
         java.lang.String stereotype,
         java.lang.String visibility,
         java.lang.String notes,
         EaHelper eaHelper)
```

getAttributeIsConst

```
protected abstract boolean getAttributeIsConst(java.lang.Object inData)
```

getAttributeIsStatic

```
protected abstract boolean getAttributeIsStatic(java.lang.Object inData)
```

(continued from last page)

getAttributeLowerBound

```
protected abstract java.lang.String getAttributeLowerBound(java.lang.Object inData)
```

getAttributeUpperBound

```
protected abstract java.lang.String getAttributeUpperBound(java.lang.Object inData)
```

getAttributeDefaultValue

```
protected abstract java.lang.String getAttributeDefaultValue(java.lang.Object inData)
```

getAttributeClassifierID

```
protected abstract int getAttributeClassifierID(java.lang.Object inData)
```

getAttributeType

```
protected abstract java.lang.String getAttributeType(java.lang.Object inData)
```

getAttributePosition

```
protected abstract int getAttributePosition(java.lang.Object inData)
```

initOwnData

```
protected final void initOwnData(int pos,
         boolean isEnum,
         boolean isConst,
         boolean isStatic,
         java.lang.String lowerBound,
         java.lang.String upperBound,
         java.lang.String defaultVal,
         int classifierID,
         java.lang.String type)
```

getType

```
public final ClassBuilder getType()
```

(continued from last page)

assignType

```
public final void assignType(java.util.Map classes)
```

It is the responsibility of the model builder to call this method after all the classes in the model have been initialised.

getContainingClass

```
public final ClassBuilder getContainingClass()
```

getPos

```
public final int getPos()
```

isConst

```
public final boolean isConst()
```

isStatic

```
public final boolean isStatic()
```

getMultiplicity

```
public final UmlMultiplicity getMultiplicity()
```

getInitValue

```
public final java.lang.String getInitValue()
```

getEaTypeId

```
public final int getEaTypeId()
```

getEaTypeName

```
public final java.lang.String getEaTypeName()
```

isLiteral

```
public final boolean isLiteral()
```

(continued from last page)

isTypeSuperfluous

```
public final boolean isTypeSuperfluous()
```

fetchAttrConstraints

```
protected abstract java.util.List fetchAttrConstraints(java.lang.Object tagsSrc)
```

getConstraints

```
public final java.util.List getConstraints()
```

fetchTaggedValues

```
protected abstract java.util.List fetchTaggedValues(java.lang.Object tagsSrc)
```

getTaggedValues

```
public final java.util.Map getTaggedValues()
```

toString

```
public java.lang.String toString()
```

getObjData

```
public final UmlObjectData getObjData()
```

doBuild

```
protected final void doBuild()
```

org.tanjakostic.jcleancim.builder.ea
Class ClassBuilder

```
java.lang.Object
  |-org.tanjakostic.jcleancim.builder.ea.AbstractObjectBuilderFromEA
    |-org.tanjakostic.jcleancim.builder.ea.ClassBuilder
```

All Implemented Interfaces:
UmlObjectBuilder

---

public abstract class **ClassBuilder**
extends AbstractObjectBuilderFromEA

Parameters:
`E` - Type for element data, `S` - Type for element as source, `D` - Type for diagrams data, `A` - Type for attributes data, `O` - Type for operations data, `C` - Type for connectors data

---

Fields inherited from class `org.tanjakostic.jcleancim.builder.ea.AbstractObjectBuilderFromEA`
`CTOR_LOG_LEVEL`

Constructor Summary

protected	`ClassBuilder(java.lang.Object inData, java.lang.Object itemsSrc, PackageBuilder containingPackage, EaHelper eaHelper)` Constructor that stores EA class data and initialises attribute and association builders.

Method Summary

void	`addAttributeAfferentClass(ClassBuilder clazz)`
void	`addAttributeEfferentClass(ClassBuilder clazz)`
void	`addOperationAfferentClass(ClassBuilder clazz)`
void	`addOperationEfferentClass(ClassBuilder clazz)`
boolean	`bothEndsAreClass(java.util.Map connIds, EaModelBuilder model)`
UmlClass	`build()` This default implementatation ; package builder should override it by throwing exception.
abstract java.util.List	`collectAttributes(java.lang.Object itemsSrc)`
abstract java.util.List	`collectClassConstraints(java.lang.Object itemsSrc)`
abstract java.util.List	`collectConnectors(java.lang.Object itemsSrc)`
abstract java.util.List	`collectContainedElements(java.lang.Object itemsSrc)`

abstract java.util.List	collectDiagrams(java.lang.Object itemsSrc)
abstract java.util.List	collectOperations(java.lang.Object itemsSrc)
abstract java.util.List	collectTaggedValues(java.lang.Object itemsSrc)
abstract AssociationBuilder	createAssociation(java.lang.Object item, ClassBuilder source, ClassBuilder target, EaModelBuilder model, EaHelper eaHelper)
abstract AttributeBuilder	createAttribute(java.lang.Object item, EaHelper eaHelper)
abstract DependencyBuilder	createDependency(java.lang.Object item, EaModelBuilder model, ClassBuilder source, ClassBuilder target, EaHelper eaHelper)
abstract DiagramBuilder	createDiagram(java.lang.Object item, EaHelper eaHelper)
abstract ClassBuilder	createEmbeddedClass(java.lang.Object item, EaHelper eaHelper) Creates class embedded into this class; containing package of the new embedded class is the same as the containing package of this class.
abstract OperationBuilder	createOperation(java.lang.Object item, EaHelper eaHelper)
abstract SkippedBuilder	createSkippedConnector(java.lang.Object item, EaModelBuilder model, EaHelper eaHelper)
abstract SkippedBuilder	createSkippedElement(java.lang.Object item, EaModelBuilder model, EaHelper eaHelper)
void	doBuild()
abstract java.util.Map	eaConnectorIDsToFields(java.lang.Object item)
abstract java.lang.String	fetchConnectorType(java.lang.Object item)
java.util.List	getAssociationSourceEndClasses() For associations where I'm on the target end, returns the list of source end classes.
java.util.List	getAssociationTargetEndClasses() For associations where I'm on the source end, returns the list of target end classes.
java.util.List	getAttributes()
java.util.Map	getConstraints()
PackageBuilder	getContainingPackage()
java.util.List	getDependencyAfferentClasses() Returns classes that depend on me through explicit UML dependency in the model.
java.util.List	getDependencyEfferentClasses() Returns classes on which I depend through explicit UML dependency in the model.

java.util.List	getDiagrams()
abstract java.lang.String	getElementAbstract(java.lang.Object inData)
abstract java.lang.String	getElementAlias(java.lang.Object inData)
abstract java.lang.String	getElementGUID(java.lang.Object inData)
abstract java.lang.Integer	getElementID(java.lang.Object inData)
abstract boolean	getElementIsLeaf(java.lang.Object inData)
abstract boolean	getElementIsRoot(java.lang.Object inData)
abstract java.lang.String	getElementName(java.lang.Object inData)
abstract java.lang.String	getElementNotes(java.lang.Object inData)
abstract java.lang.String	getElementPersistence(java.lang.Object inData)
abstract int	getElementPosition(java.lang.Object inData)
abstract java.lang.String	getElementStereotypes(java.lang.Object inData)
abstract int	getElementSubtypeVal(java.lang.Object inData)
abstract java.lang.String	getElementType(java.lang.Object inData)
abstract java.lang.String	getElementVisibility(java.lang.Object inData)
UmlObjectData	getObjData()
java.util.List	getOperations()
int	getPos()
java.lang.String	getQualifiedName()
java.util.List	getSkippedEaItems()
java.util.List	getSubclasses()
java.util.List	getSuperclasses()
java.util.Map	getTaggedValues()

boolean	isAbstract()
boolean	isAssociationClass()
static boolean	isClassOrEaInterface(java.lang.String eaType) Returns whether EA type is an EA class or an EA interface.
boolean	isEaInterface()
boolean	isEaLeafPropSet()
boolean	isEaPersistentPropSet()
boolean	isEaRootPropSet()
boolean	isEnumeratedType()
boolean	isSelfDependent()
boolean	isSelfInherited()

Methods inherited from class org.tanjakostic.jcleancim.builder.ea.AbstractObjectBuilderFromEA
build, build, doBuild, doBuild, getResult, setResult

Methods inherited from class java.lang.Object
clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.builder.UmlObjectBuilder
build, build, getObjData

Constructors

ClassBuilder

```
protected ClassBuilder(java.lang.Object inData,
                       java.lang.Object itemsSrc,
                       PackageBuilder containingPackage,
                       EaHelper eaHelper)
```

Constructor that stores EA class data and initialises attribute and association builders.

Parameters:

inData
itemsSrc
containingPackage
eaHelper

Throws:

NullPointerException - if any argument is null.

Methods

isClassOrEaInterface

`public static boolean **isClassOrEaInterface**(java.lang.String eaType)`

Returns whether EA type is an EA class or an EA interface. E.g., a class with stereotype 'interface' is **not** EA interface, because you cannot show it, for example, with the circle notation; an EA interface has some kind of special treatment.

getElementID

`protected abstract java.lang.Integer **getElementID**(java.lang.Object inData)`

getElementGUID

`protected abstract java.lang.String **getElementGUID**(java.lang.Object inData)`

getElementName

`protected abstract java.lang.String **getElementName**(java.lang.Object inData)`

getElementAlias

`protected abstract java.lang.String **getElementAlias**(java.lang.Object inData)`

getElementStereotypes

`protected abstract java.lang.String **getElementStereotypes**(java.lang.Object inData)`

getElementVisibility

`protected abstract java.lang.String **getElementVisibility**(java.lang.Object inData)`

getElementNotes

`protected abstract java.lang.String **getElementNotes**(java.lang.Object inData)`

getElementPosition

`protected abstract int **getElementPosition**(java.lang.Object inData)`

getElementType

protected abstract java.lang.String **getElementType**(java.lang.Object inData)

getElementAbstract

protected abstract java.lang.String **getElementAbstract**(java.lang.Object inData)

getElementIsLeaf

protected abstract boolean **getElementIsLeaf**(java.lang.Object inData)

getElementIsRoot

protected abstract boolean **getElementIsRoot**(java.lang.Object inData)

getElementPersistence

protected abstract java.lang.String **getElementPersistence**(java.lang.Object inData)

getElementSubtypeVal

protected abstract int **getElementSubtypeVal**(java.lang.Object inData)

collectClassConstraints

protected abstract java.util.List **collectClassConstraints**(java.lang.Object itemsSrc)

getConstraints

public final java.util.Map **getConstraints**()

collectTaggedValues

protected abstract java.util.List **collectTaggedValues**(java.lang.Object itemsSrc)

getTaggedValues

public final java.util.Map **getTaggedValues**()

(continued from last page)

collectDiagrams

```
protected abstract java.util.List collectDiagrams(java.lang.Object itemsSrc)
```

createDiagram

```
protected abstract DiagramBuilder createDiagram(java.lang.Object item,
         EaHelper eaHelper)
```

createEmbeddedClass

```
protected abstract ClassBuilder createEmbeddedClass(java.lang.Object item,
         EaHelper eaHelper)
```

Creates class embedded into this class; containing package of the new embedded class is the same as the containing package of this class.

collectContainedElements

```
protected abstract java.util.List collectContainedElements(java.lang.Object itemsSrc)
```

createSkippedElement

```
protected abstract SkippedBuilder createSkippedElement(java.lang.Object item,
         EaModelBuilder model,
         EaHelper eaHelper)
```

collectAttributes

```
protected abstract java.util.List collectAttributes(java.lang.Object itemsSrc)
```

createAttribute

```
protected abstract AttributeBuilder createAttribute(java.lang.Object item,
         EaHelper eaHelper)
```

collectOperations

```
protected abstract java.util.List collectOperations(java.lang.Object itemsSrc)
```

(continued from last page)

createOperation

```
protected abstract OperationBuilder createOperation(java.lang.Object item,
        EaHelper eaHelper)
```

collectConnectors

```
protected abstract java.util.List collectConnectors(java.lang.Object itemsSrc)
```

fetchConnectorType

```
protected abstract java.lang.String fetchConnectorType(java.lang.Object item)
```

eaConnectorIDsToFields

```
protected abstract java.util.Map eaConnectorIDsToFields(java.lang.Object item)
```

bothEndsAreClass

```
protected boolean bothEndsAreClass(java.util.Map connIds,
        EaModelBuilder model)
```

createAssociation

```
protected abstract AssociationBuilder createAssociation(java.lang.Object item,
        ClassBuilder source,
        ClassBuilder target,
        EaModelBuilder model,
        EaHelper eaHelper)
```

createDependency

```
protected abstract DependencyBuilder createDependency(java.lang.Object item,
        EaModelBuilder model,
        ClassBuilder source,
        ClassBuilder target,
        EaHelper eaHelper)
```

createSkippedConnector

```
protected abstract SkippedBuilder createSkippedConnector(java.lang.Object item,
        EaModelBuilder model,
        EaHelper eaHelper)
```

getContainingPackage

```
public final PackageBuilder getContainingPackage()
```

isSelfDependent

```
public final boolean isSelfDependent()
```

isSelfInherited

```
public final boolean isSelfInherited()
```

getPos

```
public final int getPos()
```

isEnumeratedType

```
public final boolean isEnumeratedType()
```

isAbstract

```
public final boolean isAbstract()
```

isEaPersistentPropSet

```
public final boolean isEaPersistentPropSet()
```

isEaLeafPropSet

```
public final boolean isEaLeafPropSet()
```

isEaRootPropSet

```
public final boolean isEaRootPropSet()
```

isEaInterface

```
public final boolean isEaInterface()
```

(continued from last page)

isAssociationClass

```
public final boolean isAssociationClass()
```

getSkippedEaItems

```
public final java.util.List getSkippedEaItems()
```

getAttributes

```
public final java.util.List getAttributes()
```

getOperations

```
public final java.util.List getOperations()
```

getDiagrams

```
public final java.util.List getDiagrams()
```

getSuperclasses

```
public final java.util.List getSuperclasses()
```

getSubclasses

```
public final java.util.List getSubclasses()
```

addAttributeAfferentClass

```
public final void addAttributeAfferentClass(ClassBuilder clazz)
```

addAttributeEfferentClass

```
public final void addAttributeEfferentClass(ClassBuilder clazz)
```

(continued from last page)

getAssociationSourceEndClasses

`public final java.util.List getAssociationSourceEndClasses()`

For associations where I'm on the target end, returns the list of source end classes. The result may includes this if the association is recursive (both ends of the same type).

Implementation note: If you call `Object.toString()` from within this method, ensure you add a condition to avoid recursion (because `Object.toString()` calls this method).

getAssociationTargetEndClasses

`public final java.util.List getAssociationTargetEndClasses()`

For associations where I'm on the source end, returns the list of target end classes. The result may includes this if the association is recursive (both ends of the same type).

Implementation note: If you call `Object.toString()` from within this method, ensure you add a condition to avoid recursion (because `Object.toString()` calls this method).

getDependencyAfferentClasses

`public final java.util.List getDependencyAfferentClasses()`

Returns classes that depend on me through explicit UML dependency in the model.

getDependencyEfferentClasses

`public final java.util.List getDependencyEfferentClasses()`

Returns classes on which I depend through explicit UML dependency in the model.

addOperationAfferentClass

`public final void addOperationAfferentClass(ClassBuilder clazz)`

addOperationEfferentClass

`public final void addOperationEfferentClass(ClassBuilder clazz)`

getQualifiedName

`public final java.lang.String getQualifiedName()`

getObjData

`public final UmlObjectData getObjData()`

build

`public final UmlClass build()`

(continued from last page)

This default implementatation ; package builder should override it by throwing exception.

Returns class with its data, tagged values, skipped items, constraints, diagrams and literals (in case this is an enumerated type). Model builder must add other class features (attributes, opearations, associations and dependencies) once all the classes in the model have been created.

doBuild

```
protected final void doBuild()
```

org.tanjakostic.jcleancim.builder.ea
Class ConstraintBuilder

```
java.lang.Object
  |-org.tanjakostic.jcleancim.builder.ea.AbstractObjectBuilderFromEA
    |-org.tanjakostic.jcleancim.builder.ea.ConstraintBuilder
```

All Implemented Interfaces:
UmlObjectBuilder

---

public class **ConstraintBuilder**
extends AbstractObjectBuilderFromEA

Fields inherited from class `org.tanjakostic.jcleancim.builder.ea.AbstractObjectBuilderFromEA`
`CTOR_LOG_LEVEL`

Constructor Summary

public	`ConstraintBuilder(ClassBuilder containingClass, java.lang.String name, java.lang.String notes, EaHelper helper)` Constructor for class constraint from EA object.

Method Summary

void	`doBuild()`
java.util.List	`getAttrNames()`
java.lang.String	`getCondition()`
AttributeBuilder	`getContainingAttribute()`
ClassBuilder	`getContainingClass()`
UmlKind	`getKind()`
UmlObjectData	`getObjData()`
java.lang.String	`toString()`

Methods inherited from class `org.tanjakostic.jcleancim.builder.ea.AbstractObjectBuilderFromEA`
`build, build, doBuild, doBuild, getResult, setResult`

Methods inherited from class `java.lang.Object`
`clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait`

Methods inherited from interface org.tanjakostic.jcleancim.builder.UmlObjectBuilder
build, build, getObjData

Constructors

ConstraintBuilder

```
public ConstraintBuilder(ClassBuilder containingClass,
                         java.lang.String name,
                         java.lang.String notes,
                         EaHelper helper)
```

Constructor for class constraint from EA object. Sets id to 0, and uuid, visibility, alias and stereotype to their default values - they are not defined for constraints in EA.

Throws:

`NullPointerException` - if `containingClass` or `helper` is null.

Methods

getContainingClass

```
public final ClassBuilder getContainingClass()
```

getContainingAttribute

```
public final AttributeBuilder getContainingAttribute()
```

getKind

```
public final UmlKind getKind()
```

getAttrNames

```
public final java.util.List getAttrNames()
```

getCondition

```
public final java.lang.String getCondition()
```

toString

```
public java.lang.String toString()
```

getObjData

```
public final UmlObjectData getObjData()
```

doBuild

```
protected void doBuild()
```

org.tanjakostic.jcleancim.builder.ea
Class DependencyBuilder

```
java.lang.Object
  |-org.tanjakostic.jcleancim.builder.ea.AbstractObjectBuilderFromEA
    |-org.tanjakostic.jcleancim.builder.ea.DependencyBuilder
```

All Implemented Interfaces:
UmlObjectBuilder

---

public abstract class **DependencyBuilder**
extends AbstractObjectBuilderFromEA

Parameters:
`O` - Source data for dependency, `T` - Source data for dependency tagged values

---

Fields inherited from class `org.tanjakostic.jcleancim.builder.ea.AbstractObjectBuilderFromEA`
`CTOR_LOG_LEVEL`

Constructor Summary

protected	`DependencyBuilder(java.lang.Object inData, EaModelBuilder model, java.lang.Object tagsSrc, PackageBuilder sourcePackage, PackageBuilder targetPackage, ClassBuilder sourceClass, ClassBuilder targetClass, EaHelper eaHelper)` Creates dependency between two packages or two classes.

Method Summary

void	`doBuild()`
void	`ensureClass2ClassOrPackage2PackageDependenciesEndsInitialised()` Model builder may want to call this method to cross-check initialisation is correct.
abstract java.util.List	`fetchTaggedValues(java.lang.Object taggedValues)`
abstract java.lang.String	`getConnectorAlias(java.lang.Object inData)`
abstract java.lang.String	`getConnectorGUID(java.lang.Object inData)`
abstract java.lang.Integer	`getConnectorID(java.lang.Object inData)`
abstract java.lang.String	`getConnectorName(java.lang.Object inData)`
abstract java.lang.String	`getConnectorNotes(java.lang.Object inData)`
abstract java.lang.String	`getConnectorStereotypes(java.lang.Object inData)`

UmlDependency.Kind	getKind()
UmlObjectData	getObjData()
java.lang.String	getQualifiedName()
ClassBuilder	getSourceClass()
PackageBuilder	getSourcePackage()
java.util.Map	getTaggedValues()
ClassBuilder	getTargetClass()
PackageBuilder	getTargetPackage()
static boolean	isDependency(java.lang.String type)
void	setSourceClass(ClassBuilder sourceClass)
void	setSourcePackage(PackageBuilder sourcePackage)
void	setTargetClass(ClassBuilder targetClass)
void	setTargetPackage(PackageBuilder targetPackage)
java.lang.String	toString()

Methods inherited from class org.tanjakostic.jcleancim.builder.ea.AbstractObjectBuilderFromEA
build, build, doBuild, doBuild, getResult, setResult

Methods inherited from class java.lang.Object
clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.builder.UmlObjectBuilder
build, build, getObjData

Constructors

(continued from last page)

DependencyBuilder

```
protected DependencyBuilder(java.lang.Object inData,
                            EaModelBuilder model,
                            java.lang.Object tagsSrc,
                            PackageBuilder sourcePackage,
                            PackageBuilder targetPackage,
                            ClassBuilder sourceClass,
                            ClassBuilder targetClass,
                            EaHelper eaHelper)
```

Creates dependency between two packages or two classes. Visibility is always set to `UmlVisibility.PUBLIC`. At least one of source*, target* must be non-null.

Note that because at present we don't care about characteristics of dependency ends other than the elements they connect, we store tagged values on the ends into the tagged values of the dependency itself.

Methods

isDependency

```
public static boolean isDependency(java.lang.String type)
```

---

getConnectorID

```
protected abstract java.lang.Integer getConnectorID(java.lang.Object inData)
```

---

getConnectorGUID

```
protected abstract java.lang.String getConnectorGUID(java.lang.Object inData)
```

---

getConnectorName

```
protected abstract java.lang.String getConnectorName(java.lang.Object inData)
```

---

getConnectorAlias

```
protected abstract java.lang.String getConnectorAlias(java.lang.Object inData)
```

---

getConnectorStereotypes

```
protected abstract java.lang.String getConnectorStereotypes(java.lang.Object inData)
```

---

getConnectorNotes

```
protected abstract java.lang.String getConnectorNotes(java.lang.Object inData)
```

(continued from last page)

fetchTaggedValues

```
protected abstract java.util.List fetchTaggedValues(java.lang.Object taggedValues)
```

getSourcePackage

```
public final PackageBuilder getSourcePackage()
```

setSourcePackage

```
public final void setSourcePackage(PackageBuilder sourcePackage)
```

getTargetPackage

```
public final PackageBuilder getTargetPackage()
```

setTargetPackage

```
public final void setTargetPackage(PackageBuilder targetPackage)
```

getSourceClass

```
public final ClassBuilder getSourceClass()
```

setSourceClass

```
public final void setSourceClass(ClassBuilder sourceClass)
```

getTargetClass

```
public final ClassBuilder getTargetClass()
```

setTargetClass

```
public final void setTargetClass(ClassBuilder targetClass)
```

(continued from last page)

getKind

```
public final UmlDependency.Kind getKind()
```

getTaggedValues

```
public final java.util.Map getTaggedValues()
```

getQualifiedName

```
public final java.lang.String getQualifiedName()
```

ensureClass2ClassOrPackage2PackageDependenciesEndsInitialised

```
public final void ensureClass2ClassOrPackage2PackageDependenciesEndsInitialised()
```

Model builder may want to call this method to cross-check initialisation is correct.

toString

```
public java.lang.String toString()
```

getObjData

```
public final UmlObjectData getObjData()
```

doBuild

```
protected final void doBuild()
```

org.tanjakostic.jcleancim.builder.ea
Class DiagramBuilder

```
java.lang.Object
  |-org.tanjakostic.jcleancim.builder.ea.AbstractObjectBuilderFromEA
    |-org.tanjakostic.jcleancim.builder.ea.DiagramBuilder
```

All Implemented Interfaces:
UmlObjectBuilder

---

public abstract class **DiagramBuilder**
extends AbstractObjectBuilderFromEA

Parameters:
`O` - Source data for diagram

Field Summary

public static final	DEFAULT_FILE_FORMAT

Fields inherited from class `org.tanjakostic.jcleancim.builder.ea.AbstractObjectBuilderFromEA`
CTOR_LOG_LEVEL

Constructor Summary

protected	DiagramBuilder(java.lang.Object inData, PackageBuilder containingPackage, ClassBuilder containingClass, EaHelper eaHelper) Creates diagram that belongs to a package or a class.

Method Summary

void	doBuild()
ClassBuilder	getContainingClass()
PackageBuilder	getContainingPackage()
abstract java.lang.String	getDiagramGUID(java.lang.Object inData)
abstract java.lang.Integer	getDiagramID(java.lang.Object inData)
abstract java.lang.String	getDiagramName(java.lang.Object inData)
abstract java.lang.String	getDiagramNotes(java.lang.Object inData)
abstract java.lang.String	getDiagramOrientation(java.lang.Object inData)

abstract java.lang.String	getDiagramStereotypes(java.lang.Object inData)
abstract java.lang.String	getDiagramType(java.lang.Object inData)
UmlDiagram.Kind	getKind()
UmlObjectData	getObjData()
boolean	isPortrait()
java.lang.String	toString()

Methods inherited from class org.tanjakostic.jcleancim.builder.ea.AbstractObjectBuilderFromEA
build, build, doBuild, doBuild, getResult, setResult

Methods inherited from class java.lang.Object
clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.builder.UmlObjectBuilder
build, build, getObjData

Fields

DEFAULT_FILE_FORMAT

```
public static final org.tanjakostic.jcleancim.util.Util.ImageFormat
DEFAULT_FILE_FORMAT
```

Constructors

DiagramBuilder

```
protected DiagramBuilder(java.lang.Object inData,
                         PackageBuilder containingPackage,
                         ClassBuilder containingClass,
                         EaHelper eaHelper)
```

Creates diagram that belongs to a package or a class. Visibility is always set to UmlVisibility.PUBLIC and alias to empty string (these are not defined in EA).

Parameters:

inData
containingPackage
containingClass
eaHelper

Throws:

(continued from last page)

`NullPointerException` - if both containingPackage and containingClass null, or if eaDiagram is null, or if helper is null.

Methods

getDiagramID

`protected abstract java.lang.Integer` **`getDiagramID`**`(java.lang.Object inData)`

getDiagramGUID

`protected abstract java.lang.String` **`getDiagramGUID`**`(java.lang.Object inData)`

getDiagramName

`protected abstract java.lang.String` **`getDiagramName`**`(java.lang.Object inData)`

getDiagramStereotypes

`protected abstract java.lang.String` **`getDiagramStereotypes`**`(java.lang.Object inData)`

getDiagramNotes

`protected abstract java.lang.String` **`getDiagramNotes`**`(java.lang.Object inData)`

getDiagramOrientation

`protected abstract java.lang.String` **`getDiagramOrientation`**`(java.lang.Object inData)`

getDiagramType

`protected abstract java.lang.String` **`getDiagramType`**`(java.lang.Object inData)`

getContainingPackage

`public final PackageBuilder` **`getContainingPackage`**`()`

getContainingClass

`public final ClassBuilder` **`getContainingClass`**`()`

isPortrait

```
public final boolean isPortrait()
```

getKind

```
public final UmlDiagram.Kind getKind()
```

toString

```
public java.lang.String toString()
```

getObjData

```
public final UmlObjectData getObjData()
```

doBuild

```
protected final void doBuild()
```

org.tanjakostic.jcleancim.builder.ea
Class EA

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.builder.ea.EA
```

public class **EA**
extends java.lang.Object

Utility class with various constants applicable to Enterprise Architect internal data model.

Field Summary

public static final	ATTR_CLASSIF Value: **Classifier**
public static final	ATTR_CONST Value: **Const**
public static final	ATTR_CONSTR_NAME Value: **Constraint**
public static final	ATTR_CONSTR_NOTE Value: **Notes**
public static final	ATTR_DEFAULT Value: **Default**
public static final	ATTR_ID Value: **ID**
public static final	ATTR_LOBOUND Value: **LowerBound**
public static final	ATTR_NAME Value: **Name**
public static final	ATTR_NOTE Value: **Notes**
public static final	ATTR_POSITION Value: **Pos**
public static final	ATTR_SCOPE Value: **Scope**

public static final	ATTR_STATIC Value: **IsStatic**
public static final	ATTR_STYLE Value: **Style**
protected static final	ATTR_TAGS_OUT
public static final	ATTR_TGVAL_NAME Value: **Property**
protected static final	ATTR_TGVAL_OWNER_ID Value: **ElementID**
public static final	ATTR_TGVAL_VALUE Value: **VALUE**
public static final	ATTR_TYPE Value: **Type**
public static final	ATTR_UPBOUND Value: **UpperBound**
public static final	CLASS_CONSTR_NAME Value: **Constraint**
public static final	CLASS_CONSTR_NOTE Value: **Notes**
public static final	CONN_ALIAS Value: **Alias**
public static final	CONN_DIR Value: **Direction**
public static final	CONN_FROM_AGGREG Value: **SourceIsAggregate**
public static final	CONN_FROM_ALIAS Value: **SrcAlias**
public static final	CONN_FROM_CARD Value: **SourceCard**
public static final	CONN_FROM_ID Value: **Start_Object_ID**

public static final	CONN_FROM_NAME Value: **SourceRole**
public static final	CONN_FROM_NAV Value: **SrcNav**
public static final	CONN_FROM_NOTE Value: **SourceRoleNote**
public static final	CONN_FROM_SCOPE Value: **SourceAccess**
public static final	CONN_FROM_STEREOS Value: **SrcStereo**
public static final	CONN_FROM_STYLE Value: **SourceStyle**
public static final	CONN_ID Value: **Connector_ID**
public static final	CONN_NAME Value: **Name**
public static final	CONN_NOTE Value: **Notes**
protected static final	CONN_STYLEEX Value: **StyleEx**
public static final	CONN_TGVAL_NAME Value: **Property**
protected static final	CONN_TGVAL_OWNER_ID Value: **ElementID**
public static final	CONN_TGVAL_VALUE Value: **VALUE**
public static final	CONN_TO_AGGREG Value: **DestIsAggregate**
public static final	CONN_TO_ALIAS Value: **EndAlias**
public static final	CONN_TO_CARD Value: **DestCard**

public static final	CONN_TO_ID Value: **End_Object_ID**
public static final	CONN_TO_NAME Value: **DestRole**
public static final	CONN_TO_NAV Value: **DestNav**
public static final	CONN_TO_NOTE Value: **DestRoleNote**
public static final	CONN_TO_SCOPE Value: **DestAccess**
public static final	CONN_TO_STEREOS Value: **DestStereo**
public static final	CONN_TO_STYLE Value: **DestStyle**
public static final	CONN_TYPE Value: **Connector_Type**
public static final	DEDUCED_STEREOS Value: **DeducedStereotypes**
public static final	DIA_ID Value: **Diagram_ID**
public static final	DIA_NAME Value: **Name**
public static final	DIA_NOTE Value: **Notes**
public static final	DIA_ORIENT Value: **Orientation**
protected static final	DIA_OWNER_ID Value: **ParentID**
protected static final	DIA_PCKG_ID Value: **Package_ID**
protected static final	DIA_POS Value: **TPos**

public static final	DIA_STEREO Value: **Stereotype**
public static final	DIA_TYPE Value: **Diagram_Type**
public static final	EA_GUID Value: **ea_guid**
public static final	ELEM_ABSTRACT Value: **Abstract**
public static final	ELEM_ALIAS Value: **Alias**
public static final	ELEM_ID Value: **Object_ID**
public static final	ELEM_LEAF Value: **IsLeaf**
public static final	ELEM_NAME Value: **Name**
public static final	ELEM_NOTE Value: **Note**
public static final	ELEM_PERSIST Value: **Persistence**
public static final	ELEM_POS Value: **TPos**
public static final	ELEM_ROOT Value: **IsRoot**
public static final	ELEM_SCOPE Value: **Scope**
public static final	ELEM_SUBTYPE Value: **NType**
protected static final	ELEM_TAGS_OUT
public static final	ELEM_TGVAL_NAME Value: **Property**

public static final	ELEM_TGVAL_VALUE Value: **Value**
public static final	ELEM_TYPE Value: **Object_Type**
public static final	OP_ABSTRACT Value: **Abstract**
public static final	OP_ALIAS Value: **Style**
public static final	OP_FINAL Value: **IsLeaf**
public static final	OP_ID Value: **OperationID**
public static final	OP_NAME Value: **Name**
public static final	OP_NOTE Value: **Notes**
protected static final	OP_OWNER_ID Value: **Object_ID**
public static final	OP_POS Value: **Pos**
public static final	OP_RET_ARRAY Value: **ReturnArray**
public static final	OP_RET_TYPE_ID Value: **Classifier**
public static final	OP_RET_TYPE_NAME Value: **Type**
public static final	OP_SCOPE Value: **Scope**
public static final	OP_STATIC Value: **IsStatic**
protected static final	OP_TAGS_OUT

public static final	OP_TGVAL_NAME Value: **Property**
protected static final	OP_TGVAL_OWNER_ID Value: **ElementID**
public static final	OP_TGVAL_VALUE Value: **VALUE**
public static final	PACKAGE_ID Value: **Package_ID**
public static final	PACKAGE_NAME Value: **Name**
public static final	PACKAGE_NOTE Value: **Notes**
public static final	PACKAGE_OWNER_ID Value: **Parent_ID**
public static final	PACKAGE_POS Value: **TPos**
public static final	PAR_ALIAS Value: **Alias**
public static final	PAR_CLASSIF Value: **Classifier**
public static final	PAR_NAME Value: **Name**
public static final	PAR_NOTE Value: **Notes**
protected static final	PAR_OWNER_ID Value: **OperationID**
public static final	PAR_POS Value: **Pos**
protected static final	PAR_STYLE Value: **StyleEx**
protected static final	PAR_TAGS_OUT

public static final	PAR_TYPE Value: **Type**
protected static final	PARENT_ID Value: **ParentID**
protected static final	ROLE_TGVAL_BASECLASS Value: **BaseClass**
public static final	ROLE_TGVAL_NAME Value: **TagValue**
protected static final	ROLE_TGVAL_OWNER_ID Value: **ElementID**
public static final	ROLE_TGVAL_VALUE Value: **Notes**
protected static final	XREF_CLIENT Value: **Client**
protected static final	XREF_DESCRIPTION Value: **Description**
protected static final	XREF_NAME Value: **Name**
protected static final	XREF_NAME_STEREOS Value: **Stereotypes**
protected static final	XREF_TYPE Value: **Type**
protected static final	XREF_TYPE_CONN_DEST Value: **connectorDestEnd property**
protected static final	XREF_TYPE_CONN_SRC Value: **connectorSrcEnd property**

Constructor Summary

protected	EA()

Method Summary

static java.lang.String	extractAlias(java.lang.String burried) EA burries in certain cases alias information in its Style/StyleEx table columns; this method interprets and returns such an item.

static java.lang.String	extractNavigability(java.lang.String burried) EA burries navigability information for association ends in the connector table's Style/StyleEx columns; this method interprets and returns such an item.
static java.lang.String	extractStereotypes(java.lang.String description) Exctracts stereotypes if existing, otherwise returns empty string.

Methods inherited from class java.lang.Object
clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Fields

DEDUCED_STEREOS

public static final java.lang.String **DEDUCED_STEREOS**

Constant value: **DeducedStereotypes**

EA_GUID

public static final java.lang.String **EA_GUID**

Constant value: **ea_guid**

PARENT_ID

protected static final java.lang.String **PARENT_ID**

Constant value: **ParentID**

XREF_NAME

protected static final java.lang.String **XREF_NAME**

Constant value: **Name**

XREF_NAME_STEREOS

protected static final java.lang.String **XREF_NAME_STEREOS**

Constant value: **Stereotypes**

XREF_TYPE

protected static final java.lang.String **XREF_TYPE**

Constant value: **Type**

(continued from last page)

XREF_TYPE_CONN_SRC

`protected static final java.lang.String` **`XREF_TYPE_CONN_SRC`**

Constant value: **`connectorSrcEnd property`**

XREF_TYPE_CONN_DEST

`protected static final java.lang.String` **`XREF_TYPE_CONN_DEST`**

Constant value: **`connectorDestEnd property`**

XREF_CLIENT

`protected static final java.lang.String` **`XREF_CLIENT`**

Constant value: **`Client`**

XREF_DESCRIPTION

`protected static final java.lang.String` **`XREF_DESCRIPTION`**

Constant value: **`Description`**

PACKAGE_OWNER_ID

`public static final java.lang.String` **`PACKAGE_OWNER_ID`**

Constant value: **`Parent_ID`**

PACKAGE_ID

`public static final java.lang.String` **`PACKAGE_ID`**

Constant value: **`Package_ID`**

PACKAGE_NAME

`public static final java.lang.String` **`PACKAGE_NAME`**

Constant value: **`Name`**

PACKAGE_NOTE

`public static final java.lang.String` **`PACKAGE_NOTE`**

Constant value: **`Notes`**

PACKAGE_POS

`public static final java.lang.String` **`PACKAGE_POS`**

(continued from last page)

Constant value: **TPos**

DIA_OWNER_ID

`protected static final java.lang.String` **DIA_OWNER_ID**

Constant value: **ParentID**

DIA_PCKG_ID

`protected static final java.lang.String` **DIA_PCKG_ID**

Constant value: **Package_ID**

DIA_ID

`public static final java.lang.String` **DIA_ID**

Constant value: **Diagram_ID**

DIA_NAME

`public static final java.lang.String` **DIA_NAME**

Constant value: **Name**

DIA_NOTE

`public static final java.lang.String` **DIA_NOTE**

Constant value: **Notes**

DIA_STEREO

`public static final java.lang.String` **DIA_STEREO**

Constant value: **Stereotype**

DIA_ORIENT

`public static final java.lang.String` **DIA_ORIENT**

Constant value: **Orientation**

DIA_TYPE

`public static final java.lang.String` **DIA_TYPE**

Constant value: **Diagram_Type**

DIA_POS

`protected static final java.lang.String` **`DIA_POS`**

Constant value: **`TPos`**

ELEM_ID

`public static final java.lang.String` **`ELEM_ID`**

Constant value: **`Object_ID`**

ELEM_NAME

`public static final java.lang.String` **`ELEM_NAME`**

Constant value: **`Name`**

ELEM_ALIAS

`public static final java.lang.String` **`ELEM_ALIAS`**

Constant value: **`Alias`**

ELEM_NOTE

`public static final java.lang.String` **`ELEM_NOTE`**

Constant value: **`Note`**

ELEM_SCOPE

`public static final java.lang.String` **`ELEM_SCOPE`**

Constant value: **`Scope`**

ELEM_TYPE

`public static final java.lang.String` **`ELEM_TYPE`**

Constant value: **`Object_Type`**

ELEM_ABSTRACT

`public static final java.lang.String` **`ELEM_ABSTRACT`**

Constant value: **`Abstract`**

(continued from last page)

ELEM_ROOT

`public static final java.lang.String` **`ELEM_ROOT`**

Constant value: **`IsRoot`**

ELEM_LEAF

`public static final java.lang.String` **`ELEM_LEAF`**

Constant value: **`IsLeaf`**

ELEM_PERSIST

`public static final java.lang.String` **`ELEM_PERSIST`**

Constant value: **`Persistence`**

ELEM_SUBTYPE

`public static final java.lang.String` **`ELEM_SUBTYPE`**

Constant value: **`NType`**

ELEM_POS

`public static final java.lang.String` **`ELEM_POS`**

Constant value: **`TPos`**

ELEM_TAGS_OUT

`protected static final java.lang.String` **`ELEM_TAGS_OUT`**

CLASS_CONSTR_NAME

`public static final java.lang.String` **`CLASS_CONSTR_NAME`**

Constant value: **`Constraint`**

CLASS_CONSTR_NOTE

`public static final java.lang.String` **`CLASS_CONSTR_NOTE`**

Constant value: **`Notes`**

ELEM_TGVAL_NAME

`public static final java.lang.String` **`ELEM_TGVAL_NAME`**

(continued from last page)

Constant value: **`Property`**

ELEM_TGVAL_VALUE

`public static final java.lang.String` **`ELEM_TGVAL_VALUE`**

Constant value: **`Value`**

ATTR_ID

`public static final java.lang.String` **`ATTR_ID`**

Constant value: **`ID`**

ATTR_POSITION

`public static final java.lang.String` **`ATTR_POSITION`**

Constant value: **`Pos`**

ATTR_NAME

`public static final java.lang.String` **`ATTR_NAME`**

Constant value: **`Name`**

ATTR_NOTE

`public static final java.lang.String` **`ATTR_NOTE`**

Constant value: **`Notes`**

ATTR_STYLE

`public static final java.lang.String` **`ATTR_STYLE`**

Constant value: **`Style`**

ATTR_SCOPE

`public static final java.lang.String` **`ATTR_SCOPE`**

Constant value: **`Scope`**

ATTR_CONST

`public static final java.lang.String` **`ATTR_CONST`**

Constant value: **`Const`**

(continued from last page)

ATTR_STATIC

`public static final java.lang.String` **`ATTR_STATIC`**

Constant value: **`IsStatic`**

ATTR_LOBOUND

`public static final java.lang.String` **`ATTR_LOBOUND`**

Constant value: **`LowerBound`**

ATTR_UPBOUND

`public static final java.lang.String` **`ATTR_UPBOUND`**

Constant value: **`UpperBound`**

ATTR_DEFAULT

`public static final java.lang.String` **`ATTR_DEFAULT`**

Constant value: **`Default`**

ATTR_CLASSIF

`public static final java.lang.String` **`ATTR_CLASSIF`**

Constant value: **`Classifier`**

ATTR_TYPE

`public static final java.lang.String` **`ATTR_TYPE`**

Constant value: **`Type`**

ATTR_TAGS_OUT

`protected static final java.lang.String` **`ATTR_TAGS_OUT`**

ATTR_TGVAL_OWNER_ID

`protected static final java.lang.String` **`ATTR_TGVAL_OWNER_ID`**

Constant value: **`ElementID`**

ATTR_TGVAL_NAME

`public static final java.lang.String` **`ATTR_TGVAL_NAME`**

(continued from last page)

Constant value: **Property**

ATTR_TGVAL_VALUE

`public static final java.lang.String` **ATTR_TGVAL_VALUE**

Constant value: **VALUE**

ATTR_CONSTR_NAME

`public static final java.lang.String` **ATTR_CONSTR_NAME**

Constant value: **Constraint**

ATTR_CONSTR_NOTE

`public static final java.lang.String` **ATTR_CONSTR_NOTE**

Constant value: **Notes**

OP_OWNER_ID

`protected static final java.lang.String` **OP_OWNER_ID**

Constant value: **Object_ID**

OP_ID

`public static final java.lang.String` **OP_ID**

Constant value: **OperationID**

OP_NAME

`public static final java.lang.String` **OP_NAME**

Constant value: **Name**

OP_ALIAS

`public static final java.lang.String` **OP_ALIAS**

Constant value: **Style**

OP_NOTE

`public static final java.lang.String` **OP_NOTE**

Constant value: **Notes**

(continued from last page)

OP_SCOPE

`public static final java.lang.String` **`OP_SCOPE`**

Constant value: **`Scope`**

OP_POS

`public static final java.lang.String` **`OP_POS`**

Constant value: **`Pos`**

OP_RET_TYPE_NAME

`public static final java.lang.String` **`OP_RET_TYPE_NAME`**

Constant value: **`Type`**

OP_RET_ARRAY

`public static final java.lang.String` **`OP_RET_ARRAY`**

Constant value: **`ReturnArray`**

OP_RET_TYPE_ID

`public static final java.lang.String` **`OP_RET_TYPE_ID`**

Constant value: **`Classifier`**

OP_STATIC

`public static final java.lang.String` **`OP_STATIC`**

Constant value: **`IsStatic`**

OP_ABSTRACT

`public static final java.lang.String` **`OP_ABSTRACT`**

Constant value: **`Abstract`**

OP_FINAL

`public static final java.lang.String` **`OP_FINAL`**

Constant value: **`IsLeaf`**

OP_TAGS_OUT

`protected static final java.lang.String` **`OP_TAGS_OUT`**

(continued from last page)

OP_TGVAL_OWNER_ID

`protected static final java.lang.String` **`OP_TGVAL_OWNER_ID`**

Constant value: **`ElementID`**

OP_TGVAL_NAME

`public static final java.lang.String` **`OP_TGVAL_NAME`**

Constant value: **`Property`**

OP_TGVAL_VALUE

`public static final java.lang.String` **`OP_TGVAL_VALUE`**

Constant value: **`VALUE`**

PAR_OWNER_ID

`protected static final java.lang.String` **`PAR_OWNER_ID`**

Constant value: **`OperationID`**

PAR_NAME

`public static final java.lang.String` **`PAR_NAME`**

Constant value: **`Name`**

PAR_POS

`public static final java.lang.String` **`PAR_POS`**

Constant value: **`Pos`**

PAR_NOTE

`public static final java.lang.String` **`PAR_NOTE`**

Constant value: **`Notes`**

PAR_STYLE

`protected static final java.lang.String` **`PAR_STYLE`**

Constant value: **`StyleEx`**

(continued from last page)

PAR_ALIAS

`public static final java.lang.String` **`PAR_ALIAS`**

Constant value: **`Alias`**

PAR_TYPE

`public static final java.lang.String` **`PAR_TYPE`**

Constant value: **`Type`**

PAR_CLASSIF

`public static final java.lang.String` **`PAR_CLASSIF`**

Constant value: **`Classifier`**

PAR_TAGS_OUT

`protected static final java.lang.String` **`PAR_TAGS_OUT`**

CONN_ID

`public static final java.lang.String` **`CONN_ID`**

Constant value: **`Connector_ID`**

CONN_NAME

`public static final java.lang.String` **`CONN_NAME`**

Constant value: **`Name`**

CONN_STYLEEX

`protected static final java.lang.String` **`CONN_STYLEEX`**

Constant value: **`StyleEx`**

CONN_ALIAS

`public static final java.lang.String` **`CONN_ALIAS`**

Constant value: **`Alias`**

CONN_NOTE

`public static final java.lang.String` **`CONN_NOTE`**

(continued from last page)

Constant value: **Notes**

CONN_TYPE

`public static final java.lang.String` **CONN_TYPE**

Constant value: **Connector_Type**

CONN_DIR

`public static final java.lang.String` **CONN_DIR**

Constant value: **Direction**

CONN_FROM_ID

`public static final java.lang.String` **CONN_FROM_ID**

Constant value: **Start_Object_ID**

CONN_TO_ID

`public static final java.lang.String` **CONN_TO_ID**

Constant value: **End_Object_ID**

CONN_FROM_NAME

`public static final java.lang.String` **CONN_FROM_NAME**

Constant value: **SourceRole**

CONN_TO_NAME

`public static final java.lang.String` **CONN_TO_NAME**

Constant value: **DestRole**

CONN_FROM_STYLE

`public static final java.lang.String` **CONN_FROM_STYLE**

Constant value: **SourceStyle**

CONN_TO_STYLE

`public static final java.lang.String` **CONN_TO_STYLE**

Constant value: **DestStyle**

(continued from last page)

CONN_FROM_ALIAS

`public static final java.lang.String` **`CONN_FROM_ALIAS`**

Constant value: **`SrcAlias`**

CONN_TO_ALIAS

`public static final java.lang.String` **`CONN_TO_ALIAS`**

Constant value: **`EndAlias`**

CONN_FROM_SCOPE

`public static final java.lang.String` **`CONN_FROM_SCOPE`**

Constant value: **`SourceAccess`**

CONN_TO_SCOPE

`public static final java.lang.String` **`CONN_TO_SCOPE`**

Constant value: **`DestAccess`**

CONN_FROM_STEREOS

`public static final java.lang.String` **`CONN_FROM_STEREOS`**

Constant value: **`SrcStereo`**

CONN_TO_STEREOS

`public static final java.lang.String` **`CONN_TO_STEREOS`**

Constant value: **`DestStereo`**

CONN_FROM_NOTE

`public static final java.lang.String` **`CONN_FROM_NOTE`**

Constant value: **`SourceRoleNote`**

CONN_TO_NOTE

`public static final java.lang.String` **`CONN_TO_NOTE`**

Constant value: **`DestRoleNote`**

CONN_FROM_AGGREG

`public static final java.lang.String` **`CONN_FROM_AGGREG`**

(continued from last page)

Constant value: **SourceIsAggregate**

CONN_TO_AGGREG

public static final java.lang.String **CONN_TO_AGGREG**

Constant value: **DestIsAggregate**

CONN_FROM_CARD

public static final java.lang.String **CONN_FROM_CARD**

Constant value: **SourceCard**

CONN_TO_CARD

public static final java.lang.String **CONN_TO_CARD**

Constant value: **DestCard**

CONN_FROM_NAV

public static final java.lang.String **CONN_FROM_NAV**

Constant value: **SrcNav**

CONN_TO_NAV

public static final java.lang.String **CONN_TO_NAV**

Constant value: **DestNav**

CONN_TGVAL_OWNER_ID

protected static final java.lang.String **CONN_TGVAL_OWNER_ID**

Constant value: **ElementID**

CONN_TGVAL_NAME

public static final java.lang.String **CONN_TGVAL_NAME**

Constant value: **Property**

CONN_TGVAL_VALUE

public static final java.lang.String **CONN_TGVAL_VALUE**

Constant value: **VALUE**

ROLE_TGVAL_OWNER_ID

`protected static final java.lang.String` **`ROLE_TGVAL_OWNER_ID`**

Constant value: **`ElementID`**

ROLE_TGVAL_BASECLASS

`protected static final java.lang.String` **`ROLE_TGVAL_BASECLASS`**

Constant value: **`BaseClass`**

ROLE_TGVAL_NAME

`public static final java.lang.String` **`ROLE_TGVAL_NAME`**

Constant value: **`TagValue`**

ROLE_TGVAL_VALUE

`public static final java.lang.String` **`ROLE_TGVAL_VALUE`**

Constant value: **`Notes`**

Constructors

EA

`protected` **`EA`**`()`

Methods

extractStereotypes

`public static java.lang.String` **`extractStereotypes`**`(java.lang.String description)`

Exctracts stereotypes if existing, otherwise returns empty string.

extractAlias

`public static java.lang.String` **`extractAlias`**`(java.lang.String burried)`

EA burries in certain cases alias information in its Style/StyleEx table columns; this method interprets and returns such an item.

Parameters:
`burried`

extractNavigability

`public static java.lang.String` **`extractNavigability`**`(java.lang.String burried)`

(continued from last page)

EA burries navigability information for association ends in the connector table's Style/StyleEx columns; this method interprets and returns such an item.

Parameters:

`burried`

org.tanjakostic.jcleancim.builder.ea
Class EaHelper

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.builder.ea.EaHelper
```

public class **EaHelper**
extends java.lang.Object

Constructor Summary

public	EaHelper()

Method Summary

TextDescription	getHtmlText(java.lang.String text)
TextDescription	getRawText(java.lang.String text)

Methods inherited from class `java.lang.Object`
clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Constructors

EaHelper

public **EaHelper**()

Methods

getRawText

public TextDescription **getRawText**(java.lang.String text)

getHtmlText

public TextDescription **getHtmlText**(java.lang.String text)

org.tanjakostic.jcleancim.builder.ea
Class EaModelBuilder

```
java.lang.Object
  |-org.tanjakostic.jcleancim.builder.AbstractModelBuilder
    |-org.tanjakostic.jcleancim.builder.ea.EaModelBuilder
```

All Implemented Interfaces:
ModelBuilder

Direct Known Subclasses:
SqlXmlModelBuilder, JapiModelBuilder, DbModelBuilder

---

public abstract class **EaModelBuilder**
extends AbstractModelBuilder

Class that wraps the EA repository; currently supports a single root (in the EA project browser), i.e., if there are more than one roots, all but the first will be ignored.

Parameters:
`P` - Type for package data, `S` - Type for element as source

---

Constructor Summary

protected	EaModelBuilder(Config cfg) Constructor.

Method Summary

void	addAssociation(AssociationBuilder builder)
void	addAttribute(AttributeBuilder builder)
void	addClass(ClassBuilder builder)
void	addDependency(DependencyBuilder builder)
void	addDiagram(DiagramBuilder builder)
void	addOperation(OperationBuilder builder)
void	addPackage(PackageBuilder builder)
static void	assertModelNotEmptyWarnIfMultipleRoots(int count)
UmlModel	build()
abstract void	bulkLoad()

abstract void	closeRepo()
abstract PackageBuilder	createModelPackage(java.lang.Object m)
abstract java.lang.String	fetchPackageGuid(java.lang.Object inData)
AssociationBuilder	findAssociation(java.lang.Integer assocId)
ClassBuilder	findClass(java.lang.Integer typeId)
ClassBuilder	findClass(java.lang.String name)
DependencyBuilder	findDependency(java.lang.Integer depId)
abstract java.lang.String	findElementType(java.lang.Integer id) Returns the EA type for object ID.
abstract java.lang.String	findElementTypeAndName(java.lang.Integer id)
abstract java.lang.Object	getFirstRoot()
abstract java.lang.String	getLogSubtitleEndPopulateBuilders()
abstract java.lang.String	getLogSubtitleStartPopulateBuilders()
abstract java.util.List	getModels(java.lang.Object root)
abstract EaTables	getTables() Returns tables resulting from the bulk initialisation (if applicable).
abstract java.lang.String	initRepoAndGetVersion()
boolean	isEaElementClass(java.lang.Integer objId)
boolean	isEaElementPackage(java.lang.Integer objId)
abstract void	openRepo(java.lang.String modelFileAbsPath)

Methods inherited from class org.tanjakostic.jcleancim.builder.AbstractModelBuilder

build, createDiagramExporter, createXMIExporter, getCfg, getDiagramExporter, getXMIExporter

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.builder.ModelBuilder

build, getCfg, getDiagramExporter, getXMIExporter

Constructors

EaModelBuilder

```
protected EaModelBuilder(Config cfg)
```

Constructor.

Parameters:
cfg

Methods

initRepoAndGetVersion

```
protected abstract java.lang.String initRepoAndGetVersion()
```

openRepo

```
protected abstract void openRepo(java.lang.String modelFileAbsPath)
  throws ApplicationException
```

bulkLoad

```
protected abstract void bulkLoad()
  throws ApplicationException
```

closeRepo

```
protected abstract void closeRepo()
  throws ApplicationException
```

getLogSubtitleStartPopulateBuilders

```
protected abstract java.lang.String getLogSubtitleStartPopulateBuilders()
```

getLogSubtitleEndPopulateBuilders

```
protected abstract java.lang.String getLogSubtitleEndPopulateBuilders()
```

(continued from last page)

createModelPackage

```
protected abstract PackageBuilder createModelPackage(java.lang.Object m)
```

getFirstRoot

```
protected abstract java.lang.Object getFirstRoot()
  throws ApplicationException
```

fetchPackageGuid

```
protected abstract java.lang.String fetchPackageGuid(java.lang.Object inData)
```

getModels

```
protected abstract java.util.List getModels(java.lang.Object root)
```

assertModelNotEmptyWarnIfMultipleRoots

```
protected static void assertModelNotEmptyWarnIfMultipleRoots(int count)
  throws ApplicationException
```

addPackage

```
public final void addPackage(PackageBuilder builder)
```

addDependency

```
public final void addDependency(DependencyBuilder builder)
```

findDependency

```
public final DependencyBuilder findDependency(java.lang.Integer depId)
```

addClass

```
public final void addClass(ClassBuilder builder)
```

(continued from last page)

findClass

```
public final ClassBuilder findClass(java.lang.Integer typeId)
```

findElementType

```
public abstract java.lang.String findElementType(java.lang.Integer id)
```

Returns the EA type for object ID.

findElementTypeAndName

```
public abstract java.lang.String findElementTypeAndName(java.lang.Integer id)
```

isEaElementClass

```
public boolean isEaElementClass(java.lang.Integer objId)
```

isEaElementPackage

```
public boolean isEaElementPackage(java.lang.Integer objId)
```

findClass

```
public final ClassBuilder findClass(java.lang.String name)
```

addAssociation

```
public final void addAssociation(AssociationBuilder builder)
```

findAssociation

```
public final AssociationBuilder findAssociation(java.lang.Integer assocId)
```

addDiagram

```
public final void addDiagram(DiagramBuilder builder)
```

addAttribute

```
public final void addAttribute(AttributeBuilder builder)
```

(continued from last page)

addOperation

```
public final void addOperation(OperationBuilder builder)
```

getTables

```
public abstract EaTables getTables()
  throws java.lang.UnsupportedOperationException
```

Returns tables resulting from the bulk initialisation (if applicable).

build

```
public UmlModel build()
  throws ApplicationException
```

This implementation is mainly working with EA model files. opens the EA model file, reads in all it needs, closes the EA model file and creates the in-memory model.

org.tanjakostic.jcleancim.builder.ea
Interface EaSelector

All Known Implementing Classes:
SqlXmlSelector

public interface **EaSelector**
extends

Method Summary

abstract java.util.List	select(java.lang.String tableName, java.lang.String[] columnNames, boolean skipTiming) Select columnNames from tableName.

Methods

select

```
public abstract java.util.List select(java.lang.String tableName,
         java.lang.String[] columnNames,
         boolean skipTiming)
  throws ApplicationException
```

Select columnNames from tableName.

org.tanjakostic.jcleancim.builder.ea
Interface EaSql2Xml

All Known Implementing Classes:
JapiRepo

public interface **EaSql2Xml**
extends

Isolates EA mechanism for SQL queries on the open repository (allows us to pass in a mock instead of the EA repository for testing).

Method Summary

abstract java.lang.String	sqlResultAsXml(java.lang.String queryStatement) Returns result of the SQL queryStatement as EA XML.

Methods

sqlResultAsXml

`public abstract java.lang.String` **`sqlResultAsXml`**`(java.lang.String queryStatement)`

Returns result of the SQL `queryStatement` as EA XML.

org.tanjakostic.jcleancim.builder.ea
Class EaTables

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.builder.ea.EaTables
```

public class **EaTables**
extends java.lang.Object

An attempt to speed up reading the .eap model.

The constructor takes an instance of `EaSelector` that performs access to the underlying repository and produces tables (maps) as a simple initial in-memory model. For those scenarios where we don't need to export diagrams or XMI from EA (with its repository/project methods), after construction of this instance we can safely close the EA repository.

Constructor Summary

public	`EaTables(EaSelector selector, boolean skipTiming)` Constructor; loads all the relevant content from the repository into simple data structures (maps).

Method Summary

`java.util.List`	`findAttributeConstraints(java.lang.Integer containingAttrId)` Returns constraints for `containingAttrId` if existing, empty list otherwise.
`java.util.List`	`findAttributes(java.lang.Integer containingClassId)` Returns attributes for `containingClassId` if existing, empty list otherwise.
`java.util.List`	`findAttributeTags(java.lang.Integer containingElemId)` Returns tagged values for `containingElemId` if existing, empty list otherwise.
`java.util.List`	`findClassEmbeddedElements(java.lang.Integer containingObjectId)` Returns elements embedded in `containingObjectId` if existing, empty list otherwise.
`java.util.List`	`findConnectors(boolean include, java.util.List typeNames, java.lang.Integer elementId)` Returns connectors that include or exclude `typeNames` for `elementId` if found, empty list otherwise.
`java.util.List`	`findConnectors(java.lang.Integer elementId)` Returns all connectors for `elementId` if found, empty list otherwise.
`java.util.List`	`findConnectorSourceEndTags(java.lang.Integer containingConnId)` Returns tagged values for source end of `containingConnId` if existing, empty list otherwise.
`java.util.List`	`findConnectorTags(java.lang.Integer containingElemId)` Returns tagged values for `containingElemId` if existing, empty list otherwise.
`java.util.List`	`findConnectorTargetEndTags(java.lang.Integer containingConnId)` Returns tagged values for target end of `containingConnId` if existing, empty list otherwise.

java.lang.String	findElementType(java.lang.Integer objectId) Returns type (as string) for objectId if found, null otherwise.
java.lang.String	findElementTypeAndName(java.lang.Integer objectId) Returns type and name (as string) for objectId if found, null otherwise.
java.util.List	findObjectConstraints(java.lang.Integer containingObjectId) Returns constraints for containingObjectId if existing, empty list otherwise.
java.util.List	findObjectDiagrams(java.lang.Integer containingObjectId, java.lang.String containerName) Returns ordered diagrams under object containingObjectId if existing, empty list otherwise; if containingObjectId = 0, then the returned diagrams belong to packages and it makes no sense to order them here, because you need to further filter the items for their packageId first, then order the result (sorry, that's how EA stores diagrams...).
java.util.List	findObjectTaggedValues(java.lang.Integer containingObjectId) Returns constraints for containingElemId if existing, empty list otherwise.
java.util.List	findOperations(java.lang.Integer containingClassId) Returns operations for containingClassId if existing, empty list otherwise.
java.util.List	findOperationTags(java.lang.Integer containingOpId) Returns tagged values for containingOpId if existing, empty list otherwise.
java.util.List	findOrderedParameters(java.lang.Integer containingOpId) Returns ordered parameters for containingOpId if existing, empty list otherwise.
java.util.List	findPackageClasses(java.lang.Integer containingPackageId, java.lang.String name) Returns ordered classifiers in containingPackageId if existing, empty list otherwise.
java.util.List	findPackageDiagrams(java.lang.Integer containingPackageId, java.lang.String containerName) Returns ordered diagrams under package containingPackageId if existing, empty list otherwise.
java.util.List	findPackageEmbeddedElements(java.lang.Integer containingPackageId) Returns non-classifiers and non-packages in package containingPackageId if existing, empty list otherwise.
java.util.List	findPackageSubpackages(java.lang.Integer containingPackageId, java.lang.String name) Returns ordered packages in containingPackageId if existing, empty list otherwise.

Methods inherited from class java.lang.Object
clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Constructors

EaTables

```
public EaTables(EaSelector selector,
                boolean skipTiming)
```

Constructor; loads all the relevant content from the repository into simple data structures (maps).

(continued from last page)

Parameters:

`selector` - accesses EA data

Throws:

`ApplicationException`

Methods

findPackageSubpackages

```
public java.util.List findPackageSubpackages(java.lang.Integer containingPackageId,
         java.lang.String name)
```

Returns ordered packages in `containingPackageId` if existing, empty list otherwise.

findObjectDiagrams

```
public java.util.List findObjectDiagrams(java.lang.Integer containingObjectId,
         java.lang.String containerName)
```

Returns ordered diagrams under object `containingObjectId` if existing, empty list otherwise; if `containingObjectId` = 0, then the returned diagrams belong to packages and it makes no sense to order them here, because you need to further filter the items for their packageId first, then order the result (sorry, that's how EA stores diagrams...).

findPackageDiagrams

```
public java.util.List findPackageDiagrams(java.lang.Integer containingPackageId,
         java.lang.String containerName)
```

Returns ordered diagrams under package `containingPackageId` if existing, empty list otherwise.

findPackageClasses

```
public java.util.List findPackageClasses(java.lang.Integer containingPackageId,
         java.lang.String name)
```

Returns ordered classifiers in `containingPackageId` if existing, empty list otherwise.

findPackageEmbeddedElements

```
public java.util.List findPackageEmbeddedElements(java.lang.Integer
containingPackageId)
```

Returns non-classifiers and non-packages in package `containingPackageId` if existing, empty list otherwise.

findConnectors

```
public java.util.List findConnectors(boolean include,
         java.util.List typeNames,
         java.lang.Integer elementId)
```

Returns connectors that include or exclude `typeNames` for `elementId` if found, empty list otherwise.

findConnectors

```
public java.util.List findConnectors(java.lang.Integer elementId)
```

Returns all connectors for `elementId` if found, empty list otherwise.

findElementTypeAndName

`public java.lang.String **findElementTypeAndName**(java.lang.Integer objectId)`

Returns type and name (as string) for `objectId` if found, null otherwise.

findElementType

`public java.lang.String **findElementType**(java.lang.Integer objectId)`

Returns type (as string) for `objectId` if found, null otherwise.

findClassEmbeddedElements

`public java.util.List **findClassEmbeddedElements**(java.lang.Integer containingObjectId)`

Returns elements embedded in `containingObjectId` if existing, empty list otherwise.

findObjectConstraints

`public java.util.List **findObjectConstraints**(java.lang.Integer containingObjectId)`

Returns constraints for `containingObjectId` if existing, empty list otherwise.

findObjectTaggedValues

`public java.util.List **findObjectTaggedValues**(java.lang.Integer containingObjectId)`

Returns constraints for `containingElemId` if existing, empty list otherwise.

findAttributes

`public java.util.List **findAttributes**(java.lang.Integer containingClassId)`

Returns attributes for `containingClassId` if existing, empty list otherwise.

findAttributeConstraints

`public java.util.List **findAttributeConstraints**(java.lang.Integer containingAttrId)`

Returns constraints for `containingAttrId` if existing, empty list otherwise.

findAttributeTags

`public java.util.List **findAttributeTags**(java.lang.Integer containingElemId)`

Returns tagged values for `containingElemId` if existing, empty list otherwise.

findConnectorTags

`public java.util.List **findConnectorTags**(java.lang.Integer containingElemId)`

Returns tagged values for `containingElemId` if existing, empty list otherwise.

findConnectorSourceEndTags

`public java.util.List **findConnectorSourceEndTags**(java.lang.Integer containingConnId)`

(continued from last page)

Returns tagged values for source end of `containingConnId` if existing, empty list otherwise.

findConnectorTargetEndTags

`public java.util.List` **`findConnectorTargetEndTags`**`(java.lang.Integer containingConnId)`

Returns tagged values for target end of `containingConnId` if existing, empty list otherwise.

findOperations

`public java.util.List` **`findOperations`**`(java.lang.Integer containingClassId)`

Returns operations for `containingClassId` if existing, empty list otherwise.

findOrderedParameters

`public java.util.List` **`findOrderedParameters`**`(java.lang.Integer containingOpId)`

Returns ordered parameters for `containingOpId` if existing, empty list otherwise.

findOperationTags

`public java.util.List` **`findOperationTags`**`(java.lang.Integer containingOpId)`

Returns tagged values for `containingOpId` if existing, empty list otherwise.

org.tanjakostic.jcleancim.builder.ea
Class OperationBuilder

```
java.lang.Object
  |-org.tanjakostic.jcleancim.builder.ea.AbstractObjectBuilderFromEA
    |-org.tanjakostic.jcleancim.builder.ea.OperationBuilder
```

All Implemented Interfaces:
UmlObjectBuilder

---

public abstract class **OperationBuilder**
extends AbstractObjectBuilderFromEA

Parameters:
`O` - Source data for operation, `T` - Source data for operation tagged values and parameters

---

Fields inherited from class `org.tanjakostic.jcleancim.builder.ea.AbstractObjectBuilderFromEA`
`CTOR_LOG_LEVEL`

Constructor Summary

protected	`OperationBuilder(java.lang.Object inData, java.lang.Object tagsSrc, ClassBuilder containingClass, EaHelper eaHelper)` Constructor for operation from EA object.

Method Summary

void	`assignTypeToParametersAndExceptions(EaModelBuilder model)` It is the responsibility of the model builder to call this method after all the classes in the model have been initialised.
java.util.List	`createEaExceptionTypeInfo()`
abstract void	`createParams(java.lang.Object parsSrc, EaHelper eaHelper)`
void	`doBuild()`
abstract java.util.List	`fetchTaggedValues(java.lang.Object tagsSrc)`
ClassBuilder	`getContainingClass()`
int	`getEaReturnTypeId()`
java.lang.String	`getEaReturnTypeName()`
java.util.List	`getExceptionNames()` Returns potentially empty list of exception names.
java.util.List	`getExceptions()`

java.lang.String	getExceptionsSignature() Returns potentially empty string containing comma-separated list of exceptions that follow the 'throw' statement in operation signature.
UmlOperation.ReturnKind	getKind()
UmlObjectData	getObjData()
abstract java.lang.String	getOperationAlias(java.lang.Object inData)
abstract java.lang.String	getOperationClassifierID(java.lang.Object inData)
abstract java.lang.String	getOperationGUID(java.lang.Object inData)
abstract java.lang.Integer	getOperationID(java.lang.Object inData)
abstract boolean	getOperationIsAbstract(java.lang.Object inData)
abstract boolean	getOperationIsLeaf(java.lang.Object inData)
abstract boolean	getOperationIsReturnArray(java.lang.Object inData)
abstract boolean	getOperationIsStatic(java.lang.Object inData)
abstract java.lang.String	getOperationName(java.lang.Object inData)
abstract java.lang.String	getOperationNotes(java.lang.Object inData)
abstract int	getOperationPosition(java.lang.Object inData)
abstract java.lang.String	getOperationReturnType(java.lang.Object inData)
abstract java.lang.String	getOperationStereotypes(java.lang.Object inData)
abstract java.lang.String	getOperationVisibility(java.lang.Object inData)
java.util.List	getParameters()
int	getPos()
ClassBuilder	getReturnType()
java.util.Map	getTaggedValues()
boolean	isAbstract()

boolean	isFinal()
boolean	isStatic()
void	setReturnType(ClassBuilder returnType)
java.lang.String	toString()

Methods inherited from class org.tanjakostic.jcleancim.builder.ea.AbstractObjectBuilderFromEA
build, build, doBuild, doBuild, getResult, setResult

Methods inherited from class java.lang.Object
clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.builder.UmlObjectBuilder
build, build, getObjData

Constructors

OperationBuilder

```
protected OperationBuilder(java.lang.Object inData,
                           java.lang.Object tagsSrc,
                           ClassBuilder containingClass,
                           EaHelper eaHelper)
```

Constructor for operation from EA object.

Parameters:

inData
tagsSrc
containingClass
eaHelper

Throws:

NullPointerException - if any argument is null.

Methods

getOperationID

```
protected abstract java.lang.Integer getOperationID(java.lang.Object inData)
```

getOperationGUID

```
protected abstract java.lang.String getOperationGUID(java.lang.Object inData)
```

getOperationName

protected abstract java.lang.String **getOperationName**(java.lang.Object inData)

getOperationAlias

protected abstract java.lang.String **getOperationAlias**(java.lang.Object inData)

getOperationStereotypes

protected abstract java.lang.String **getOperationStereotypes**(java.lang.Object inData)

getOperationVisibility

protected abstract java.lang.String **getOperationVisibility**(java.lang.Object inData)

getOperationNotes

protected abstract java.lang.String **getOperationNotes**(java.lang.Object inData)

getOperationPosition

protected abstract int **getOperationPosition**(java.lang.Object inData)

getOperationIsAbstract

protected abstract boolean **getOperationIsAbstract**(java.lang.Object inData)

getOperationIsStatic

protected abstract boolean **getOperationIsStatic**(java.lang.Object inData)

getOperationIsLeaf

protected abstract boolean **getOperationIsLeaf**(java.lang.Object inData)

getOperationIsReturnArray

protected abstract boolean **getOperationIsReturnArray**(java.lang.Object inData)

(continued from last page)

getOperationReturnType

```
protected abstract java.lang.String getOperationReturnType(java.lang.Object inData)
```

getOperationClassifierID

```
protected abstract java.lang.String getOperationClassifierID(java.lang.Object inData)
```

getExceptionsSignature

```
public final java.lang.String getExceptionsSignature()
```

Returns potentially empty string containing comma-separated list of exceptions that follow the 'throw' statement in operation signature.

getExceptionNames

```
public final java.util.List getExceptionNames()
```

Returns potentially empty list of exception names.

createParams

```
protected abstract void createParams(java.lang.Object parsSrc,
        EaHelper eaHelper)
```

assignTypeToParametersAndExceptions

```
public final void assignTypeToParametersAndExceptions(EaModelBuilder model)
```

It is the responsibility of the model builder to call this method after all the classes in the model have been initialised.

getContainingClass

```
public final ClassBuilder getContainingClass()
```

getPos

```
public final int getPos()
```

isAbstract

```
public final boolean isAbstract()
```

isStatic

```
public final boolean isStatic()
```

isFinal

```
public final boolean isFinal()
```

getKind

```
public final UmlOperation.ReturnKind getKind()
```

getEaReturnTypeId

```
public final int getEaReturnTypeId()
```

getEaReturnTypeName

```
public final java.lang.String getEaReturnTypeName()
```

fetchTaggedValues

```
protected abstract java.util.List fetchTaggedValues(java.lang.Object tagsSrc)
```

getTaggedValues

```
public final java.util.Map getTaggedValues()
```

createEaExceptionTypeInfo

```
public final java.util.List createEaExceptionTypeInfo()
```

getExceptions

```
public final java.util.List getExceptions()
```

setReturnType

```
public final void setReturnType(ClassBuilder returnType)
```

(continued from last page)

getReturnType

```
public final ClassBuilder getReturnType()
```

getParameters

```
public final java.util.List getParameters()
```

toString

```
public java.lang.String toString()
```

getObjData

```
public final UmlObjectData getObjData()
```

doBuild

```
protected final void doBuild()
```

org.tanjakostic.jcleancim.builder.ea
Class PackageBuilder

```
java.lang.Object
  |-org.tanjakostic.jcleancim.builder.ea.AbstractObjectBuilderFromEA
    |-org.tanjakostic.jcleancim.builder.ea.PackageBuilder
```

All Implemented Interfaces:
UmlObjectBuilder

Direct Known Subclasses:
DbPackageBuilder

---

public abstract class **PackageBuilder**
extends AbstractObjectBuilderFromEA

Parameters:
`P` - Type for package data, `E` - Type for element data, `SP` - Type for package as source, `SE` - Type for element as source, `D` - Type for diagram data, `C` - Type for connector data

---

Fields inherited from class `org.tanjakostic.jcleancim.builder.ea.AbstractObjectBuilderFromEA`
`CTOR_LOG_LEVEL`

Constructor Summary

protected	`PackageBuilder(java.lang.Object inData, java.lang.Object inDataE, java.lang.Object itemsSrcP, java.lang.Object itemsSrcE, EaModelBuilder model, PackageBuilder containingPackage, int modelId, EaHelper eaHelper)` Constructor.

Method Summary

boolean	`bothEndsArePackage(java.util.Map connIds, EaModelBuilder model)`
UmlPackage	`build()`
UmlPackage	`build(UmlModel model)` This default implementatation throws only exception; package builder should override it.
java.util.List	`collectAfferentPackages()` Returns all packages that depend on me through an explicit UML dependency in the model.
abstract java.util.List	`collectConnectors(java.lang.Object itemsSrc)`
abstract java.util.List	`collectDiagrams(java.lang.Object itemsSrc)`
java.util.List	`collectEfferentPackages()` Returns all packages that I depend on through an explicit UML dependency in the model.

abstract java.util.List	collectPackageElements(java.lang.Object itemsSrcP)
abstract java.util.List	collectSubPackages(java.lang.Object itemsSrcP)
abstract java.util.List	collectTaggedValues(java.lang.Object itemsSrc)
abstract ClassBuilder	createClass(java.lang.Object item, EaHelper eaHelper)
abstract DependencyBuilder	createDependency(java.lang.Object item, EaModelBuilder model, PackageBuilder source, PackageBuilder target, EaHelper eaHelper)
abstract DiagramBuilder	createDiagram(java.lang.Object item, EaHelper eaHelper)
abstract SkippedBuilder	createSkippedConnector(java.lang.Object item, EaModelBuilder model, EaHelper eaHelper)
abstract SkippedBuilder	createSkippedElement(java.lang.Object item, EaModelBuilder model, EaHelper eaHelper)
abstract PackageBuilder	createSubPackage(java.lang.Object item, EaHelper eaHelper)
void	doBuild()
void	doBuild(UmlModel model)
abstract java.util.Map	eaConnectorIDsToFields(java.lang.Object item)
abstract java.lang.String	fetchConnectorType(java.lang.Object item)
abstract java.lang.String	fetchElementType(java.lang.Object item)
java.util.List	getChildPackages()
java.util.List	getClasses()
java.util.List	getClassUuids() Returns UUIDs of classes in the order they are defined in the repository.
PackageBuilder	getContainingPackage()
java.util.List	getDependenciesAsSource()
java.util.List	getDependenciesAsTarget()
int	getDepth()
java.util.List	getDiagrams()

java.lang.Integer	getEaElementID()
UmlPackage.Kind	getKind()
EaModelBuilder	getModel()
int	getModelId()
UmlObjectData	getObjData()
abstract java.lang.String	getPackageAlias(java.lang.Object inData)
abstract java.lang.Integer	getPackageElementID(java.lang.Object inDataE)
abstract java.lang.String	getPackageGUID(java.lang.Object inData)
abstract java.lang.Integer	getPackageID(java.lang.Object inData)
abstract java.lang.String	getPackageName(java.lang.Object inData)
abstract java.lang.String	getPackageNotes(java.lang.Object inData)
abstract java.lang.Integer	getPackageParentID(java.lang.Object inData)
abstract int	getPackagePos(java.lang.Object inData)
abstract java.lang.String	getPackageStereotypes(java.lang.Object inData)
abstract java.lang.String	getPackageVisibility(java.lang.Object inDataE)
int	getPos()
java.lang.String	getQualifiedName()
java.util.List	getSkippedEaItems()
java.util.Map	getTaggedValues()
static boolean	isEaPackage(java.lang.String eaType)
boolean	isSelfDependent()
java.lang.String	toString()

Methods inherited from class org.tanjakostic.jcleancim.builder.ea.AbstractObjectBuilderFromEA
build, build, doBuild, doBuild, getResult, setResult

Methods inherited from class `java.lang.Object`
`clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait`

Methods inherited from interface `org.tanjakostic.jcleancim.builder.UmlObjectBuilder`
`build, build, getObjData`

Constructors

PackageBuilder

```
protected PackageBuilder(java.lang.Object inData,
                         java.lang.Object inDataE,
                         java.lang.Object itemsSrcP,
                         java.lang.Object itemsSrcE,
                         EaModelBuilder model,
                         PackageBuilder containingPackage,
                         int modelId,
                         EaHelper eaHelper)
```

Constructor. Package is stored in EA DB table for elements, but there is also a specific package table, based on packageId; with respect to the data we use, it contains only one item not present in elements table: parent package ID. However, when using API, EA does the chatty queries, so we use eaPackage as much as possible (that table is smaller and queries are faster).

Parameters:

`inData`
`inDataE`
`itemsSrcP`
`itemsSrcE`
`model`
`containingPackage`
`modelId`
`eaHelper`

Methods

isEaPackage

```
public static boolean isEaPackage(java.lang.String eaType)
```

getPackageID

```
protected abstract java.lang.Integer getPackageID(java.lang.Object inData)
```

getPackageGUID

```
protected abstract java.lang.String getPackageGUID(java.lang.Object inData)
```

getPackageName

protected abstract java.lang.String **getPackageName**(java.lang.Object inData)

getPackageNotes

protected abstract java.lang.String **getPackageNotes**(java.lang.Object inData)

getPackageAlias

protected abstract java.lang.String **getPackageAlias**(java.lang.Object inData)

getPackageStereotypes

protected abstract java.lang.String **getPackageStereotypes**(java.lang.Object inData)

getPackageVisibility

protected abstract java.lang.String **getPackageVisibility**(java.lang.Object inDataE)

getPackagePos

protected abstract int **getPackagePos**(java.lang.Object inData)

getPackageParentID

protected abstract java.lang.Integer **getPackageParentID**(java.lang.Object inData)

getPackageElementID

protected abstract java.lang.Integer **getPackageElementID**(java.lang.Object inDataE)

collectTaggedValues

protected abstract java.util.List **collectTaggedValues**(java.lang.Object itemsSrc)

getTaggedValues

public final java.util.Map **getTaggedValues**()

(continued from last page)

collectDiagrams

```
protected abstract java.util.List collectDiagrams(java.lang.Object itemsSrc)
```

createDiagram

```
protected abstract DiagramBuilder createDiagram(java.lang.Object item,
         EaHelper eaHelper)
```

collectConnectors

```
protected abstract java.util.List collectConnectors(java.lang.Object itemsSrc)
```

fetchConnectorType

```
protected abstract java.lang.String fetchConnectorType(java.lang.Object item)
```

eaConnectorIDsToFields

```
protected abstract java.util.Map eaConnectorIDsToFields(java.lang.Object item)
```

bothEndsArePackage

```
protected boolean bothEndsArePackage(java.util.Map connIds,
         EaModelBuilder model)
```

createSkippedConnector

```
protected abstract SkippedBuilder createSkippedConnector(java.lang.Object item,
         EaModelBuilder model,
         EaHelper eaHelper)
```

createDependency

```
protected abstract DependencyBuilder createDependency(java.lang.Object item,
         EaModelBuilder model,
         PackageBuilder source,
         PackageBuilder target,
         EaHelper eaHelper)
```

(continued from last page)

collectPackageElements

```
protected abstract java.util.List collectPackageElements(java.lang.Object itemsSrcP)
```

fetchElementType

```
protected abstract java.lang.String fetchElementType(java.lang.Object item)
```

createSkippedElement

```
protected abstract SkippedBuilder createSkippedElement(java.lang.Object item,
         EaModelBuilder model,
         EaHelper eaHelper)
```

createClass

```
protected abstract ClassBuilder createClass(java.lang.Object item,
         EaHelper eaHelper)
```

collectSubPackages

```
protected abstract java.util.List collectSubPackages(java.lang.Object itemsSrcP)
```

createSubPackage

```
protected abstract PackageBuilder createSubPackage(java.lang.Object item,
         EaHelper eaHelper)
```

getModel

```
public final EaModelBuilder getModel()
```

getContainingPackage

```
public final PackageBuilder getContainingPackage()
```

getKind

```
public final UmlPackage.Kind getKind()
```

(continued from last page)

getDepth

```
public final int getDepth()
```

getModelId

```
public final int getModelId()
```

getPos

```
public final int getPos()
```

getEaElementID

```
public final java.lang.Integer getEaElementID()
```

isSelfDependent

```
public final boolean isSelfDependent()
```

getSkippedEaItems

```
public final java.util.List getSkippedEaItems()
```

getDependenciesAsSource

```
public final java.util.List getDependenciesAsSource()
```

getDependenciesAsTarget

```
public final java.util.List getDependenciesAsTarget()
```

getDiagrams

```
public final java.util.List getDiagrams()
```

getClasses

```
public final java.util.List getClasses()
```

(continued from last page)

getChildPackages

```
public final java.util.List getChildPackages()
```

collectEfferentPackages

```
public final java.util.List collectEfferentPackages()
```

Returns all packages that I depend on through an explicit UML dependency in the model.

collectAfferentPackages

```
public final java.util.List collectAfferentPackages()
```

Returns all packages that depend on me through an explicit UML dependency in the model.

getClassUuids

```
public final java.util.List getClassUuids()
```

Returns UUIDs of classes in the order they are defined in the repository.

getQualifiedName

```
public final java.lang.String getQualifiedName()
```

toString

```
public java.lang.String toString()
```

getObjData

```
public final UmlObjectData getObjData()
```

build

```
public UmlPackage build()
```

This default implementatation ; package builder should override it by throwing exception.

doBuild

```
protected void doBuild()
```

build

```
public final UmlPackage build(UmlModel model)
```

This default implementatation throws only exception; package builder should override it.

Recursively builds the model skeleton with packages and classes, and all other items that do not require references to classes. The in-memory model returned from here is thus only half-built. The model builder must finish the build process ("link") by calling the builders for class features (attributes, operations, associations, dependencies) and for packages (dependencies), because they all require existing, valid classes and packages.

doBuild

```
protected final void doBuild(UmlModel model)
```

This default implementatation throws only exception; package builder should override it.

org.tanjakostic.jcleancim.builder.ea
Class ParameterBuilder

```
java.lang.Object
  +-org.tanjakostic.jcleancim.builder.ea.AbstractObjectBuilderFromEA
      +-org.tanjakostic.jcleancim.builder.ea.ParameterBuilder
```

All Implemented Interfaces:
UmlObjectBuilder

---

public abstract class **ParameterBuilder**
extends AbstractObjectBuilderFromEA

Does not have tagged values.
Parameters:
o - Source data for operation parameter

Fields inherited from class `org.tanjakostic.jcleancim.builder.ea.AbstractObjectBuilderFromEA`
`CTOR_LOG_LEVEL`

Constructor Summary

protected	ParameterBuilder(java.lang.Object inData, OperationBuilder containingOperation, EaHelper eaHelper)

Method Summary

ClassBuilder	assignType(java.lang.String opFullyQualifiedName, EaModelBuilder model) It is the responsibility of the model builder or its delegate to call this method after all the classes in the model have been initialised.
void	doBuild()
OperationBuilder	getContainingOperation()
java.lang.String	getEaTypeIdAsString()
java.lang.String	getEaTypeInfo()
java.lang.String	getEaTypeName()
UmlKind	getKind()
UmlObjectData	getObjData()
abstract java.lang.String	getParameterAlias(java.lang.Object inData)
abstract java.lang.String	getParameterClassifierID(java.lang.Object inData)

abstract java.lang.String	getParameterGUID(java.lang.Object inData)
abstract java.lang.String	getParameterName(java.lang.Object inData)
abstract java.lang.String	getParameterNotes(java.lang.Object inData)
abstract int	getParameterPosition(java.lang.Object inData)
abstract java.lang.String	getParameterStereotypes(java.lang.Object inData)
abstract java.lang.String	getParameterType(java.lang.Object inData)
int	getPosition()
ClassBuilder	getType()
java.lang.String	toString()

Methods inherited from class org.tanjakostic.jcleancim.builder.ea.AbstractObjectBuilderFromEA
build, build, doBuild, doBuild, getResult, setResult

Methods inherited from class java.lang.Object
clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.builder.UmlObjectBuilder
build, build, getObjData

Constructors

ParameterBuilder

```
protected ParameterBuilder(java.lang.Object inData,
                           OperationBuilder containingOperation,
                           EaHelper eaHelper)
```

Methods

getParameterGUID

```
protected abstract java.lang.String getParameterGUID(java.lang.Object inData)
```

(continued from last page)

getParameterName

```
protected abstract java.lang.String getParameterName(java.lang.Object inData)
```

getParameterAlias

```
protected abstract java.lang.String getParameterAlias(java.lang.Object inData)
```

getParameterStereotypes

```
protected abstract java.lang.String getParameterStereotypes(java.lang.Object inData)
```

getParameterNotes

```
protected abstract java.lang.String getParameterNotes(java.lang.Object inData)
```

getParameterType

```
protected abstract java.lang.String getParameterType(java.lang.Object inData)
```

getParameterClassifierID

```
protected abstract java.lang.String getParameterClassifierID(java.lang.Object inData)
```

getParameterPosition

```
protected abstract int getParameterPosition(java.lang.Object inData)
```

assignType

```
public final ClassBuilder assignType(java.lang.String opFullyQualifiedName,
         EaModelBuilder model)
```

It is the responsibility of the model builder or its delegate to call this method after all the classes in the model have been initialised.

getContainingOperation

```
public final OperationBuilder getContainingOperation()
```

(continued from last page)

getEaTypeName

```
public final java.lang.String getEaTypeName()
```

getEaTypeIdAsString

```
public final java.lang.String getEaTypeIdAsString()
```

getPosition

```
public final int getPosition()
```

getKind

```
public final UmlKind getKind()
```

getType

```
public final ClassBuilder getType()
```

getEaTypeInfo

```
public final java.lang.String getEaTypeInfo()
```

toString

```
public java.lang.String toString()
```

getObjData

```
public final UmlObjectData getObjData()
```

doBuild

```
protected final void doBuild()
```

org.tanjakostic.jcleancim.builder.ea
Class SkippedBuilder

```
java.lang.Object
  +-org.tanjakostic.jcleancim.builder.ea.AbstractObjectBuilderFromEA
    +-org.tanjakostic.jcleancim.builder.ea.SkippedBuilder
```

All Implemented Interfaces:
UmlObjectBuilder

---

public abstract class **SkippedBuilder**
extends AbstractObjectBuilderFromEA

Parameters:
`E` - Source data for skipped element, `S` - Source data for skipped element's diagrams, `C` - Source data for skipped connector, `D` - Source data for diagram

---

Fields inherited from class `org.tanjakostic.jcleancim.builder.ea.AbstractObjectBuilderFromEA`
`CTOR_LOG_LEVEL`

Constructor Summary

protected	SkippedBuilder(java.lang.Object inDataE, java.lang.Object itemsSrc, java.lang.Object inDataC, PackageBuilder p, ClassBuilder c, EaModelBuilder model, EaHelper eaHelper) Constructor.

Method Summary

abstract java.util.List	collectDiagrams(java.lang.Object itemsSrc)
abstract DiagramBuilder	createDiagram(java.lang.Object item, EaHelper eaHelper)
void	doBuild()
abstract java.lang.String	getConnectorAlias(java.lang.Object inDataC)
abstract java.lang.Integer	getConnectorClientID(java.lang.Object inDataC)
abstract java.lang.String	getConnectorGUID(java.lang.Object inDataC)
abstract java.lang.Integer	getConnectorID(java.lang.Object inDataC)
abstract java.lang.String	getConnectorName(java.lang.Object inDataC)
abstract java.lang.String	getConnectorNotes(java.lang.Object inDataC)

abstract java.lang.String	getConnectorStereotypes(java.lang.Object inDataC)
abstract java.lang.Integer	getConnectorSupplierID(java.lang.Object inDataC)
abstract java.lang.String	getConnectorType(java.lang.Object inDataC)
ClassBuilder	getContainingClass()
PackageBuilder	getContainingPackage()
java.util.List	getDiagrams()
abstract java.lang.String	getElementAlias(java.lang.Object inDataE)
abstract java.lang.String	getElementGUID(java.lang.Object inDataE)
abstract java.lang.Integer	getElementID(java.lang.Object inDataE)
abstract java.lang.String	getElementName(java.lang.Object inDataE)
abstract java.lang.String	getElementNotes(java.lang.Object inDataE)
abstract java.lang.String	getElementStereotypes(java.lang.Object inDataE)
abstract java.lang.String	getElementType(java.lang.Object inDataE)
UmlSkipped.Kind	getKind()
UmlObjectData	getObjData()
java.lang.String	getOtherEndName()
boolean	isConnector()
java.lang.String	toString()

Methods inherited from class org.tanjakostic.jcleancim.builder.ea.AbstractObjectBuilderFromEA
build, build, doBuild, doBuild, getResult, setResult

Methods inherited from class java.lang.Object
clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.builder.UmlObjectBuilder
build, build, getObjData

Constructors

SkippedBuilder

```
protected SkippedBuilder(java.lang.Object inDataE,
                         java.lang.Object itemsSrc,
                         java.lang.Object inDataC,
                         PackageBuilder p,
                         ClassBuilder c,
                         EaModelBuilder model,
                         EaHelper eaHelper)
```

Constructor. Creates skipped relationship or element for class or package. Visibility is always set to `UmlVisibility.PUBLIC`.

Parameters:

`inDataE`
`itemsSrc`
`inDataC`
`p`
`c`
`model`
`eaHelper`

Methods

getElementID

```
protected abstract java.lang.Integer getElementID(java.lang.Object inDataE)
```

getElementGUID

```
protected abstract java.lang.String getElementGUID(java.lang.Object inDataE)
```

getElementName

```
protected abstract java.lang.String getElementName(java.lang.Object inDataE)
```

getElementAlias

```
protected abstract java.lang.String getElementAlias(java.lang.Object inDataE)
```

getElementStereotypes

```
protected abstract java.lang.String getElementStereotypes(java.lang.Object inDataE)
```

getElementNotes

protected abstract java.lang.String **getElementNotes**(java.lang.Object inDataE)

getElementType

protected abstract java.lang.String **getElementType**(java.lang.Object inDataE)

getConnectorID

protected abstract java.lang.Integer **getConnectorID**(java.lang.Object inDataC)

getConnectorGUID

protected abstract java.lang.String **getConnectorGUID**(java.lang.Object inDataC)

getConnectorName

protected abstract java.lang.String **getConnectorName**(java.lang.Object inDataC)

getConnectorAlias

protected abstract java.lang.String **getConnectorAlias**(java.lang.Object inDataC)

getConnectorStereotypes

protected abstract java.lang.String **getConnectorStereotypes**(java.lang.Object inDataC)

getConnectorNotes

protected abstract java.lang.String **getConnectorNotes**(java.lang.Object inDataC)

getConnectorType

protected abstract java.lang.String **getConnectorType**(java.lang.Object inDataC)

getConnectorClientID

protected abstract java.lang.Integer **getConnectorClientID**(java.lang.Object inDataC)

(continued from last page)

getConnectorSupplierID

```
protected abstract java.lang.Integer getConnectorSupplierID(java.lang.Object inDataC)
```

collectDiagrams

```
protected abstract java.util.List collectDiagrams(java.lang.Object itemsSrc)
```

createDiagram

```
protected abstract DiagramBuilder createDiagram(java.lang.Object item,
         EaHelper eaHelper)
```

getContainingPackage

```
public final PackageBuilder getContainingPackage()
```

getContainingClass

```
public final ClassBuilder getContainingClass()
```

isConnector

```
public final boolean isConnector()
```

getKind

```
public final UmlSkipped.Kind getKind()
```

getOtherEndName

```
public final java.lang.String getOtherEndName()
```

getDiagrams

```
public final java.util.List getDiagrams()
```

(continued from last page)

toString

```
public java.lang.String toString()
```

getObjData

```
public final UmlObjectData getObjData()
```

doBuild

```
protected final void doBuild()
```

Package

org.tanjakostic.jcleancim.builder.ea.db

org.tanjakostic.jcleancim.builder.ea.db
Class DbModelBuilder

```
java.lang.Object
  |-org.tanjakostic.jcleancim.builder.AbstractModelBuilder
    |-org.tanjakostic.jcleancim.builder.ea.EaModelBuilder
      |-org.tanjakostic.jcleancim.builder.ea.db.DbModelBuilder
```

All Implemented Interfaces:
ModelBuilder

---

public class **DbModelBuilder**
extends EaModelBuilder

The fastest builder of our in-memory model from EA. It is the new implementation based on Jackcess library that allows reading MS Access file in an OS-independent way, and independently of EA API.

Limitation: Note that with this implementation we don't have access to the EA repository (API) methods, so we cannot export diagrams or XMI - although we do provide "empty" exporters, so that this implementation can hook into the existing framework.

This implementation should be used for very fast {edit UML - validate} cycles. When you need to produce a UML release (with XMI) and/or generate any kind of documentation with diagrams, ensure you swap this implementation with the one that can export XMI and diagrams.

Constructor Summary

public	DbModelBuilder(Config cfg) Constructor.

Method Summary

void	bulkLoad()
void	closeRepo()
DiagramExporter	createDiagramExporter()
PackageBuilder	createModelPackage(java.util.Map inData)
XMIExporter	createXMIExporter()
java.lang.String	fetchPackageGuid(java.util.Map inData)
java.lang.String	findElementType(java.lang.Integer id)
java.lang.String	findElementTypeAndName(java.lang.Integer id)
java.util.Map	getFirstRoot()

java.lang.String	getLogSubtitleEndPopulateBuilders()
java.lang.String	getLogSubtitleStartPopulateBuilders()
java.util.List	getModels(java.util.Map rootPckRow)
EaTables	getTables()
java.lang.String	initRepoAndGetVersion()
void	openRepo(java.lang.String modelFileAbsPath)

Methods inherited from class org.tanjakostic.jcleancim.builder.ea.EaModelBuilder

addAssociation, addAttribute, addClass, addDependency, addDiagram, addOperation, addPackage, assertModelNotEmptyWarnIfMultipleRoots, build, bulkLoad, closeRepo, createModelPackage, fetchPackageGuid, findAssociation, findClass, findClass, findDependency, findElementType, findElementTypeAndName, getFirstRoot, getLogSubtitleEndPopulateBuilders, getLogSubtitleStartPopulateBuilders, getModels, getTables, initRepoAndGetVersion, isEaElementClass, isEaElementPackage, openRepo

Methods inherited from class org.tanjakostic.jcleancim.builder.AbstractModelBuilder

build, createDiagramExporter, createXMIExporter, getCfg, getDiagramExporter, getXMIExporter

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.builder.ModelBuilder

build, getCfg, getDiagramExporter, getXMIExporter

Constructors

DbModelBuilder

public **DbModelBuilder**(Config cfg)

Constructor.

Parameters:
cfg

Methods

initRepoAndGetVersion

protected java.lang.String **initRepoAndGetVersion**()

openRepo

```
protected void openRepo(java.lang.String modelFileAbsPath)
  throws ApplicationException
```

closeRepo

```
protected void closeRepo()
  throws ApplicationException
```

bulkLoad

```
protected void bulkLoad()
  throws ApplicationException
```

getFirstRoot

```
protected java.util.Map getFirstRoot()
  throws ApplicationException
```

getModels

```
protected java.util.List getModels(java.util.Map rootPckRow)
```

getLogSubtitleStartPopulateBuilders

```
protected java.lang.String getLogSubtitleStartPopulateBuilders()
```

getLogSubtitleEndPopulateBuilders

```
protected java.lang.String getLogSubtitleEndPopulateBuilders()
```

createModelPackage

```
protected PackageBuilder createModelPackage(java.util.Map inData)
```

findElementType

```
public java.lang.String findElementType(java.lang.Integer id)
```

Returns the EA type for object ID.

findElementTypeAndName

```
public java.lang.String findElementTypeAndName(java.lang.Integer id)
```

fetchPackageGuid

```
protected java.lang.String fetchPackageGuid(java.util.Map inData)
```

createDiagramExporter

```
protected final DiagramExporter createDiagramExporter()
```

Creates exporter of UML diagrams where applicable (otherwise, can be just a stub).

createXMIExporter

```
protected final XMIExporter createXMIExporter()
```

Returns exporter to XMI where applicable (otherwise, can be just a stub).

getTables

```
public EaTables getTables()
```

Returns tables resulting from the bulk initialisation (if applicable).

org.tanjakostic.jcleancim.builder.ea.db
Class DbPackageBuilder

```
java.lang.Object
  +-org.tanjakostic.jcleancim.builder.ea.AbstractObjectBuilderFromEA
    +-org.tanjakostic.jcleancim.builder.ea.PackageBuilder
      +-org.tanjakostic.jcleancim.builder.ea.db.DbPackageBuilder
```

All Implemented Interfaces:
UmlObjectBuilder

---

public class **DbPackageBuilder**
extends PackageBuilder

Fields inherited from class `org.tanjakostic.jcleancim.builder.ea.AbstractObjectBuilderFromEA`
`CTOR_LOG_LEVEL`

Constructor Summary

public	`DbPackageBuilder(java.util.Map inData, EaModelBuilder model, PackageBuilder containingPackage, int modelId, EaHelper eaHelper)` Constructor.

Method Summary

`java.util.List`	`collectConnectors(EaModelBuilder itemsSrcE)`
`java.util.List`	`collectDiagrams(EaModelBuilder itemsSrc)`
`java.util.List`	`collectPackageElements(EaModelBuilder itemsSrcP)`
`java.util.List`	`collectSubPackages(EaModelBuilder itemsSrcP)`
`java.util.List`	`collectTaggedValues(EaModelBuilder itemsSrc)`
`ClassBuilder`	`createClass(java.util.Map item, EaHelper eaHelper)`
`DependencyBuilder`	`createDependency(java.util.Map item, EaModelBuilder model, PackageBuilder source, PackageBuilder target, EaHelper eaHelper)`
`DiagramBuilder`	`createDiagram(java.util.Map item, EaHelper eaHelper)`
static `PackageBuilder`	`createModelPackageBuilder(java.util.Map item, EaModelBuilder model, EaHelper eaHelper)` Creates model package from EA object; loads all the model contents recursively.

SkippedBuilder	createSkippedConnector(java.util.Map item, EaModelBuilder model, EaHelper eaHelper)
SkippedBuilder	createSkippedElement(java.util.Map item, EaModelBuilder model, EaHelper eaHelper)
PackageBuilder	createSubPackage(java.util.Map item, EaHelper eaHelper)
java.util.Map	eaConnectorIDsToFields(java.util.Map item)
java.lang.String	fetchConnectorType(java.util.Map item)
java.lang.String	fetchElementType(java.util.Map item)
java.lang.String	getPackageAlias(java.util.Map inData)
java.lang.Integer	getPackageElementID(java.util.Map inDataE)
java.lang.String	getPackageGUID(java.util.Map inData)
java.lang.Integer	getPackageID(java.util.Map inData)
java.lang.String	getPackageName(java.util.Map inData)
java.lang.String	getPackageNotes(java.util.Map inData)
java.lang.Integer	getPackageParentID(java.util.Map inData)
int	getPackagePos(java.util.Map inData)
java.lang.String	getPackageStereotypes(java.util.Map inData)
java.lang.String	getPackageVisibility(java.util.Map inDataE)

Methods inherited from class org.tanjakostic.jcleancim.builder.ea.PackageBuilder

bothEndsArePackage, build, build, collectAfferentPackages, collectConnectors, collectDiagrams, collectEfferentPackages, collectPackageElements, collectSubPackages, collectTaggedValues, createClass, createDependency, createDiagram, createSkippedConnector, createSkippedElement, createSubPackage, doBuild, doBuild, eaConnectorIDsToFields, fetchConnectorType, fetchElementType, getChildPackages, getClasses, getClassUuids, getContainingPackage, getDependenciesAsSource, getDependenciesAsTarget, getDepth, getDiagrams, getEaElementID, getKind, getModel, getModelId, getObjData, getPackageAlias, getPackageElementID, getPackageGUID, getPackageID, getPackageName, getPackageNotes, getPackageParentID, getPackagePos, getPackageStereotypes, getPackageVisibility, getPos, getQualifiedName, getSkippedEaItems, getTaggedValues, isEaPackage, isSelfDependent, toString

Methods inherited from class org.tanjakostic.jcleancim.builder.ea.AbstractObjectBuilderFromEA

build, build, doBuild, doBuild, getResult, setResult

Methods inherited from class `java.lang.Object`
`clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait`

Methods inherited from interface `org.tanjakostic.jcleancim.builder.UmlObjectBuilder`
`build, build, getObjData`

Constructors

DbPackageBuilder

```
public DbPackageBuilder(java.util.Map inData,
                        EaModelBuilder model,
                        PackageBuilder containingPackage,
                        int modelId,
                        EaHelper eaHelper)
```

Constructor.

Parameters:

`inData`
`model`
`containingPackage`
`modelId`
`eaHelper`

Methods

createModelPackageBuilder

```
public static PackageBuilder createModelPackageBuilder(java.util.Map item,
         EaModelBuilder model,
         EaHelper eaHelper)
```

Creates model package from EA object; loads all the model contents recursively.

Parameters:

`item` - EA package that is wrapped by this UML package.
`model` - parent UML model (EA repository wrapper, needed for extracting diagrams for printing to clipboard and for formatted UML docs of elements and connectors)
`eaHelper` - we need this to save diagrams and formatted text.

Throws:

`NullPointerException` - if any argument is null.

getPackageID

```
protected java.lang.Integer getPackageID(java.util.Map inData)
```

getPackageGUID

```
protected java.lang.String getPackageGUID(java.util.Map inData)
```

(continued from last page)

getPackageName

protected java.lang.String **getPackageName**(java.util.Map inData)

getPackageAlias

protected java.lang.String **getPackageAlias**(java.util.Map inData)

getPackageStereotypes

protected java.lang.String **getPackageStereotypes**(java.util.Map inData)

getPackageVisibility

protected java.lang.String **getPackageVisibility**(java.util.Map inDataE)

getPackageNotes

protected java.lang.String **getPackageNotes**(java.util.Map inData)

getPackagePos

protected int **getPackagePos**(java.util.Map inData)

getPackageParentID

protected java.lang.Integer **getPackageParentID**(java.util.Map inData)

getPackageElementID

protected java.lang.Integer **getPackageElementID**(java.util.Map inDataE)

collectTaggedValues

protected java.util.List **collectTaggedValues**(EaModelBuilder itemsSrc)

(continued from last page)

collectDiagrams

```
protected java.util.List collectDiagrams(EaModelBuilder itemsSrc)
```

createDiagram

```
protected DiagramBuilder createDiagram(java.util.Map item,
         EaHelper eaHelper)
```

collectConnectors

```
protected java.util.List collectConnectors(EaModelBuilder itemsSrcE)
```

fetchConnectorType

```
protected java.lang.String fetchConnectorType(java.util.Map item)
```

eaConnectorIDsToFields

```
protected java.util.Map eaConnectorIDsToFields(java.util.Map item)
```

createSkippedConnector

```
protected SkippedBuilder createSkippedConnector(java.util.Map item,
         EaModelBuilder model,
         EaHelper eaHelper)
```

createDependency

```
protected DependencyBuilder createDependency(java.util.Map item,
         EaModelBuilder model,
         PackageBuilder source,
         PackageBuilder target,
         EaHelper eaHelper)
```

collectPackageElements

```
protected java.util.List collectPackageElements(EaModelBuilder itemsSrcP)
```

fetchElementType

```
protected java.lang.String fetchElementType(java.util.Map item)
```

(continued from last page)

createSkippedElement

```
protected SkippedBuilder createSkippedElement(java.util.Map item,
         EaModelBuilder model,
         EaHelper eaHelper)
```

createClass

```
protected ClassBuilder createClass(java.util.Map item,
         EaHelper eaHelper)
```

collectSubPackages

```
protected java.util.List collectSubPackages(EaModelBuilder itemsSrcP)
```

createSubPackage

```
protected PackageBuilder createSubPackage(java.util.Map item,
         EaHelper eaHelper)
```

Package

org.tanjakostic.jcleancim.builder.ea.japi

org.tanjakostic.jcleancim.builder.ea.japi
Class JapiModelBuilder

```
java.lang.Object
  +-org.tanjakostic.jcleancim.builder.AbstractModelBuilder
    +-org.tanjakostic.jcleancim.builder.ea.EaModelBuilder
      +-org.tanjakostic.jcleancim.builder.ea.japi.JapiModelBuilder
```

All Implemented Interfaces:
ModelBuilder

---

public class **JapiModelBuilder**
extends EaModelBuilder

The slowest builder of our in-memory model from EA: it uses the very slow EA API and iterates over its collections. It is the refactored version of the original implementation since 01v01. With this implementation we access the EA repository and can thus export diagrams and XMI if required.

We intentionally keep this implementation because we hope Sparx will one day provide a fast implementation...

Constructor Summary

public	JapiModelBuilder(Config cfg) Constructor.

Method Summary

void	bulkLoad()
void	closeRepo()
DiagramExporter	createDiagramExporter()
PackageBuilder	createModelPackage(org.sparx.Package m)
XMIExporter	createXMIExporter()
java.lang.String	fetchPackageGuid(org.sparx.Package inData)
java.lang.String	findElementType(java.lang.Integer objId)
java.lang.String	findElementTypeAndName(java.lang.Integer id)
org.sparx.Package	getFirstRoot()
java.lang.String	getLogSubtitleEndPopulateBuilders()
java.lang.String	getLogSubtitleStartPopulateBuilders()

java.util.List	getModels(org.sparx.Package root)
EaTables	getTables()
java.lang.String	initRepoAndGetVersion()
void	openRepo(java.lang.String modelFileAbsPath)

Methods inherited from class org.tanjakostic.jcleancim.builder.ea.EaModelBuilder

addAssociation, addAttribute, addClass, addDependency, addDiagram, addOperation, addPackage, assertModelNotEmptyWarnIfMultipleRoots, build, bulkLoad, closeRepo, createModelPackage, fetchPackageGuid, findAssociation, findClass, findClass, findDependency, findElementType, findElementTypeAndName, getFirstRoot, getLogSubtitleEndPopulateBuilders, getLogSubtitleStartPopulateBuilders, getModels, getTables, initRepoAndGetVersion, isEaElementClass, isEaElementPackage, openRepo

Methods inherited from class org.tanjakostic.jcleancim.builder.AbstractModelBuilder

build, createDiagramExporter, createXMIExporter, getCfg, getDiagramExporter, getXMIExporter

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.builder.ModelBuilder

build, getCfg, getDiagramExporter, getXMIExporter

Constructors

JapiModelBuilder

public **JapiModelBuilder**(Config cfg)

Constructor.

Parameters:
cfg

Methods

initRepoAndGetVersion

protected java.lang.String **initRepoAndGetVersion**()

openRepo

protected void **openRepo**(java.lang.String modelFileAbsPath)

(continued from last page)

closeRepo

```
protected void closeRepo()
  throws ApplicationException
```

bulkLoad

```
protected void bulkLoad()
```

getFirstRoot

```
protected org.sparx.Package getFirstRoot()
  throws ApplicationException
```

getModels

```
protected java.util.List getModels(org.sparx.Package root)
```

getLogSubtitleStartPopulateBuilders

```
protected java.lang.String getLogSubtitleStartPopulateBuilders()
```

getLogSubtitleEndPopulateBuilders

```
protected java.lang.String getLogSubtitleEndPopulateBuilders()
```

createModelPackage

```
protected PackageBuilder createModelPackage(org.sparx.Package m)
```

findElementTypeAndName

```
public java.lang.String findElementTypeAndName(java.lang.Integer id)
```

findElementType

```
public java.lang.String findElementType(java.lang.Integer objId)
```

Returns the EA type for object ID.

fetchPackageGuid

```
protected java.lang.String fetchPackageGuid(org.sparx.Package inData)
```

createDiagramExporter

```
protected final DiagramExporter createDiagramExporter()
```

Creates exporter of UML diagrams where applicable (otherwise, can be just a stub).

createXMIExporter

```
protected final XMIExporter createXMIExporter()
```

Returns exporter to XMI where applicable (otherwise, can be just a stub).

getTables

```
public EaTables getTables()
  throws java.lang.UnsupportedOperationException
```

Returns tables resulting from the bulk initialisation (if applicable).

org.tanjakostic.jcleancim.builder.ea.japi
Class JapiRepo

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.builder.ea.japi.JapiRepo
```

All Implemented Interfaces:
EaSql2Xml

public class **JapiRepo**
extends java.lang.Object
implements EaSql2Xml

Constructor Summary

public	JapiRepo() Constructor.

Method Summary

void	close()
static java.util.List	eaToJavaList(org.sparx.Collection eaCollection)
DiagramExporter	getDiagramExporter(Config cfg)
java.lang.String	getVersion()
XMIExporter	getXMIExporter(Config cfg)
void	open(java.lang.String modelFileAbsPath)
java.lang.String	sqlResultAsXml(java.lang.String queryStatement)

Methods inherited from class `java.lang.Object`

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface `org.tanjakostic.jcleancim.builder.ea.EaSql2Xml`

sqlResultAsXml

Constructors

(continued from last page)

JapiRepo

```
public JapiRepo()
```

Constructor.

Methods

getVersion

```
public java.lang.String getVersion()
```

open

```
public void open(java.lang.String modelFileAbsPath)
```

close

```
public void close()
  throws ApplicationException
```

getDiagramExporter

```
public DiagramExporter getDiagramExporter(Config cfg)
```

getXMIExporter

```
public XMIExporter getXMIExporter(Config cfg)
```

sqlResultAsXml

```
public java.lang.String sqlResultAsXml(java.lang.String queryStatement)
```

eaToJavaList

```
public static java.util.List eaToJavaList(org.sparx.Collection eaCollection)
```

Package

org.tanjakostic.jcleancim.builder.ea.sqlxml

org.tanjakostic.jcleancim.builder.ea.sqlxml
Class SqlXmlModelBuilder

```
java.lang.Object
  |-org.tanjakostic.jcleancim.builder.AbstractModelBuilder
    |-org.tanjakostic.jcleancim.builder.ea.EaModelBuilder
      |-org.tanjakostic.jcleancim.builder.ea.sqlxml.SqlXmlModelBuilder
```

All Implemented Interfaces:
ModelBuilder

---

public class **SqlXmlModelBuilder**
extends EaModelBuilder

The fast builder of our in-memory model from EA. It is the refactored version of the first implementation (in 01v07). With this implementation we access the EA repository and can thus export diagrams and XMI if required.

Constructor Summary

public	SqlXmlModelBuilder(Config cfg) Constructor.

Method Summary

void	bulkLoad()
void	closeRepo()
DiagramExporter	createDiagramExporter()
PackageBuilder	createModelPackage(java.util.Map inData)
XMIExporter	createXMIExporter()
java.lang.String	fetchPackageGuid(java.util.Map inData)
java.lang.String	findElementType(java.lang.Integer id)
java.lang.String	findElementTypeAndName(java.lang.Integer id)
java.util.Map	getFirstRoot()
java.lang.String	getLogSubtitleEndPopulateBuilders()
java.lang.String	getLogSubtitleStartPopulateBuilders()
java.util.List	getModels(java.util.Map rootPckRow)

EaTables	getTables()
java.lang.String	initRepoAndGetVersion()
void	openRepo(java.lang.String modelFileAbsPath)

Methods inherited from class org.tanjakostic.jcleancim.builder.ea.EaModelBuilder

addAssociation, addAttribute, addClass, addDependency, addDiagram, addOperation, addPackage, assertModelNotEmptyWarnIfMultipleRoots, build, bulkLoad, closeRepo, createModelPackage, fetchPackageGuid, findAssociation, findClass, findClass, findDependency, findElementType, findElementTypeAndName, getFirstRoot, getLogSubtitleEndPopulateBuilders, getLogSubtitleStartPopulateBuilders, getModels, getTables, initRepoAndGetVersion, isEaElementClass, isEaElementPackage, openRepo

Methods inherited from class org.tanjakostic.jcleancim.builder.AbstractModelBuilder

build, createDiagramExporter, createXMIExporter, getCfg, getDiagramExporter, getXMIExporter

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.builder.ModelBuilder

build, getCfg, getDiagramExporter, getXMIExporter

Constructors

SqlXmlModelBuilder

public **SqlXmlModelBuilder**(Config cfg)

Constructor.

Parameters:
cfg

Methods

initRepoAndGetVersion

protected java.lang.String **initRepoAndGetVersion**()

openRepo

protected void **openRepo**(java.lang.String modelFileAbsPath)

closeRepo

```
protected void closeRepo()
  throws ApplicationException
```

bulkLoad

```
protected void bulkLoad()
  throws ApplicationException
```

getFirstRoot

```
protected java.util.Map getFirstRoot()
  throws ApplicationException
```

getModels

```
protected java.util.List getModels(java.util.Map rootPckRow)
```

getLogSubtitleStartPopulateBuilders

```
protected java.lang.String getLogSubtitleStartPopulateBuilders()
```

getLogSubtitleEndPopulateBuilders

```
protected java.lang.String getLogSubtitleEndPopulateBuilders()
```

createModelPackage

```
protected PackageBuilder createModelPackage(java.util.Map inData)
```

findElementType

```
public java.lang.String findElementType(java.lang.Integer id)
```

Returns the EA type for object ID.

findElementTypeAndName

```
public java.lang.String findElementTypeAndName(java.lang.Integer id)
```

fetchPackageGuid

`protected java.lang.String fetchPackageGuid(java.util.Map inData)`

createDiagramExporter

`protected final DiagramExporter createDiagramExporter()`

Creates exporter of UML diagrams where applicable (otherwise, can be just a stub).

createXMIExporter

`protected final XMIExporter createXMIExporter()`

Returns exporter to XMI where applicable (otherwise, can be just a stub).

getTables

`public EaTables getTables()`

Returns tables resulting from the bulk initialisation (if applicable).

org.tanjakostic.jcleancim.builder.ea.sqlxml
Class SqlXmlSelector

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.builder.ea.sqlxml.SqlXmlSelector
```

All Implemented Interfaces:
EaSelector

public class **SqlXmlSelector**
extends java.lang.Object
implements EaSelector

EA repository supports a method to perform an SQL query and return the result set as XML. This class is a wrapper to that EA functionality without dependency on EA.

Constructor Summary

public	SqlXmlSelector(EaSql2Xml queror)

Method Summary

java.util.List	select(java.lang.String tableName, java.lang.String[] columnNames, boolean logTime)

Methods inherited from class java.lang.Object
clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.builder.ea.EaSelector
select

Constructors

SqlXmlSelector

```
public SqlXmlSelector(EaSql2Xml queror)
```

Methods

select

```
public java.util.List select(java.lang.String tableName,
         java.lang.String[] columnNames,
         boolean logTime)
```

Package
org.tanjakostic.jcleancim.common

Classes commonly used by several packages.

Important classes are:

- `Config` - contains information parsed from org.tanjakostic.jcleancim.common.Config#DEFAULT_PROPS_FILE_NAME file, according to which the whole application will run.
- `OwningWg` - contains definition for the ownership of top-level packages by IEC working groups.
- `Nature` - contains two values, used to "classify" the nature of the model (packages) and be able to correctly do validation, statistics and document generation.

TODO:

- Do we need to have more flexible ownership?

org.tanjakostic.jcleancim.common
Class Config

```
java.lang.Object
   |
   +-org.tanjakostic.jcleancim.common.Config
```

public class **Config**
extends java.lang.Object

Configuration is read from three configuration files.

1. main user configuration file, with the default name #DEFAULT_PROPS_FILE_NAME, contains main user-configurable definitions and options common to any UML model nature. It can be replaced by another one provided as command line argument. If no main configuration file has been specified or found on the classpath, default value #DEFAULT_PROPS_FILE_NAME is used. Such a sample file, with the valid defaults, is provided with every distribution of jCleanCim.
2. IEC 61850-specific configuration file, with the fixed name #IEC61850_PROPS_FILE_NAME , contains mainly rarely modified definitions, and thus removes the clutter from the main configuration that the user may edit frequently.
3. jCleanCim version configuration file, with the fixed name #VERSION_PROPS_FILENAME, is edited only when branching development of jCleanCim, to properly tag its current version. The user is not expected to edit it.

Files

Input files are mainly expected to be available in any place that is on the classpath. By convention, we configure the directory `input` under the project root to be on the classpath (implementation note: we simply search for input files as resources that must be on the classpath, and we set the classpath to include directory `input`; this eliminates the need for absolute paths). The application never modifies anything under `input` directory. Files searched on the classpath are for instance model file org.tanjakostic.jcleancim.common.Config#KEY_MODEL_FILENAME, XML schema for document generation (defined only as default: #DEFAULT_WEBACCESS_SCHEMA_FILENAME), or MS Word templace file `KEY_DOCGEN_WORD_IN_TEMPLATE`.

Output files are created in a separate directory, #OUTPUT_DIR_NAME, under the current execution path obtained with the Java system property org.tanjakostic.jcleancim.util.Util#USER_DIR_KEY. If such a directory is not available, like in case of a fresh installation, it gets automatically created (implementation note: using the Java system property org.tanjakostic.jcleancim.util.Util#USER_DIR_KEY ensures that the "home" for #OUTPUT_DIR_NAME directory will be under the project root directory when you run jCleanCim application, and under `test` directory when you run tests).

The constructor of this class completes all the tricks related to files, both input and output, and once it completed successfully, it is sure that all the file absolute paths are valid: if there is any problem related to resources, we will fail fast, before starting potentially lengthy operations. Furthermore, if an output file (to be produced by the current run of jCleanCim) already exists in the #OUTPUT_DIR_NAME directory, it will be renamed already in the constructor to ensure that it does not get overwritten later on.

Other properties

Several available properties allow you to configure how to run jCleanCim: which main functions to run or not (validation, statistics, document generation), and to fine tune their execution. At present, this is the most comfortable way to configure a run of jCleanCim.

Because this is a java application, every property found in the #DEFAULT_PROPS_FILE_NAME file can be overwritten when launching the application by providing one or more `-D<propertyName>=<propertyValue>` statements immediately after the `java` command, but if there are many properties to configure, it is simpler to do it in the #DEFAULT_PROPS_FILE_NAME file.

Implementation note: The values obtained from properties have been validated and stored in appropriate format in the constructor. For instance, "true" is read as string and stored as boolean; comma-separated string read from file is stored as a list of strings; absolute file paths are produced from the simple file names. These are then made available through methods to the application.

Follows the description of individual properties.

Controlling some aspect of the overall application

As of release 01v09, there is one such property:

- Set #KEY_APP_SKIP_TIMING = "true" when debugging overall application INFO log, to allow seemless text comparison of two consecutive runs of the application. By default, the timing of major steps get logged.

Top-level properties, to select the functionality to execute

You control what gets executed by enabling ("true") or disabling ("false" = value, "" = value omitted, null = whole property absent) one or more of the top level options. The values are shown below enclosed in "" to denote some text, but they should be typed in the properties file without "":

- Set #KEY_XMIEXPORT_ON = "true" to export the .eap model to the three XMI formats (XMI 1.1, XMI 2.1 and CIMTool XMI 1.4/Rose); this option is independent from other top-level options, but makes sense only if the .eap model file is available.
- Set #KEY_VALIDATION_ON = "true" to run model validation; this option is independent from other top-level options.
- Set #KEY_STATISTICS_ON = "true" to run model statistics; this option is independent from other top-level options.
- Set #KEY_PROFILES_CROSSCHECK_ON = "true" to run crosscheck between UML model and multiple profiles; this option is independent from other top-level options. Note: not yet implemented.
- Set #KEY_MIBGEN_ON = "true"to run MIBs generation; this option is independent from other top-level options.
- Set #KEY_DOCGEN_ON = "true" and #KEY_PROFILES_DOCGEN_ON = "false","", null to run document generation from UML model (as required for IEC61968-11, IEC61970-301 or IEC61850-7-4).
- Set #KEY_DOCGEN_ON = "true" and #KEY_PROFILES_DOCGEN_ON = "true" to run MS Word document generation from one or more CIM RDF/OWL profiles (as required for IEC61970-452 or IEC61968-13 documents). At present, it is impossible to create multiple Word documents. Note: not yet implemented.

Model-related properties

These properties specify the UML or other model to work with:

- Property org.tanjakostic.jcleancim.common.Config#KEY_MODEL_FILENAME holds the name of the EA file containing UML model. This file is expected to be found on the classpath. The value is ignored if you invoke the application with the `-m <myModel.eap>` command line argument. The value specified with this property is useful if you always use the same configuration file, so you need not type command line argument. A valid UML repository is required for every scenario, except:
  - [NOT YET IMPLEMENTED] when doing docgen for one or more profiles, the model is built from the files in profiles subdirectories #KEY_PROFILES_RELPATH/#KEY_PROFILES_DIRNAMES
  - when you populate the in-memory model through the API instead of reading from EA file (with version 01v04 or higher).
- Property #KEY_MODEL_BUILDER allows you to choose the most performant loading of the model .eap file given your usage requirements. For full support of diagram and XMI export use `ModelBuilderKind.sqlxml`. This implementation is based on SQL queries for reading the model .eap file (and replaces `model.useSql=true` option of 01v07). It is almost order of magnitude faster than the regular API calls (option `ModelBuilderKind.japi`). Since 01v08, we have a rocket-fast implementation, with `ModelBuilderKind.db` in case you don't need to export diagrams or XMI. Note that both non-API options work properly for the .eap file based on Acess RDBMS only.
- Property #KEY_MODEL_NATURE_IEC61850 allows you to specify a list of model packages (directly below the root in the model repository) that are IEC61850, or derive from it. Potential IEC61850-family model packages not specified in this list will simply be processed as if they were CIM (and this is normally not what one wants...).
- Property #KEY_PROFILES_DIRNAMES allows you to specify one or more subdirectories under relative path #KEY_PROFILES_RELPATH, under which to search for profile files, of format #XSD_EXT. This is required if running UML/profile cross-check or if generating MS Word documentation for profiles. The value should reflect one of more IEC `WGs` owning the profiles. These names are important as they are the only simple means to determine which WG owns the profile without requiring other configuration by the user. For all of the validation of cross-referencing and document generation, we are building an in-memory model with profiles as packages, and therefore we have to assign them the owning WG the same way we do when building the in-memory model from a UML repository. Any number of further subdirectories is allowed, as all the files below the selected subdirectory get scanned. If the value is left empty, profile files of all WGs found under the default profiles directory `DEFAULT_PROFILES_RELPATH` supported profile extension #XSD_EXT will be picked and processed.

XMI export properties

This set of properties controls XMI export functionality, and are relevant only if #KEY_XMIEXPORT_ON="true".

- Property #KEY_XMIEXPORT_DIALECTS, if not empty or absent, allows to select which XMI `dialects` to export.

Common validation properties

This set of properties controls validation functionality, and are relevant only if #KEY_VALIDATION_ON="true".

- Property #KEY_VALIDATION_SCOPE, if not empty or absent, allows to filter the scope for validation per IEC `WG` owning the top-level package. Note that the whole of the model must be read in order to determine the scope of associations and explicit UML dependencies, as well as inheritance among classes from different packages. The scope is then used for validation and statistics, but NOT for doc generation purposes. This filter is useful for model editors who want to do validation and see statistics for their field of concern only. However, before releasing any updated model, it is recommended to do the full validation, i.e., to leave this value empty.
- Properties #KEY_VALIDATION_PACKAGES_OFF, #KEY_VALIDATION_CLASSES_OFF, #KEY_VALIDATION_ATTRIBUTES_OFF, #KEY_VALIDATION_ASSOCIATIONS_OFF, #KEY_VALIDATION_OPERATIONS_OFF, #KEY_VALIDATION_DEPENDENCIES_OFF and #KEY_VALIDATION_DIAGRAMS_OFF, when set "true", allow to disable the whole family or validaton rules, applicable to the given type of UML element (package, ..., diagram, respectively). For a model editor, in case there are many validation error or warnings, it may be convenient to temporarily disable the validation for all except for 1 type of elements. However, before releasing any updated model, it is recommended to do the full validation, i.e., to leave these values empty.
- Property #KEY_VALIDATION_RULES_OFF, if not empty or absent, allows you to disable individual validation rules. Since version 01v04, console log (as well as log file) contain the full set of available rules, so you can just copy/paste the desired class names, separated with a comma, as value for this key. Note: Line escape character is backslash "\". This may be useful if there are "noisy" warnings that you cannot fix for a given release, so by disabling specific one or two rules, you can temporarily reduce the noise in the output.
- Property #KEY_VALIDATION_LOGGING_VERBOSE, if set "true", allows you to have on the console output all the validation rules displayed, as they are fired, even if they produce no error or warning. This may be handy until you get familiar with all the available rules or for debugging, but typically you'll have this option disabled ("false", "" or null).
- Property #KEY_VALIDATION_PACKAGES_DATA_INDEX is used for validation and documentation generation of parts of IEC61850-7-4 and IEC61850-7-3, but may be handy for CIM models (for debugging), and that is why it is not specified as IEC61850-specific property. Namely, if your MS Word document template has one or more data index placeholders (to generate an alphabetical index of all the attributes defined on classes within or below a given package), you must specify the names of the packages that you want to use in placeholders for data indexes. To support the primary need of IEC61850-7-3 or IEC61850-7-4 documents, the data indexes are built for the given package and all its content recursively. To avoid building those indexes for everything in the UML model, this option requires to specify one or more packages for which the data index should be built (i.e., made available for printing). Package names provided in the configuration file contain real values for IEC61850, and as an example the package Core from CIM - this is then used in the sample MS Word document template to illustrate functionality. If the MS Word document template does not include the data index placeholder, the data index (even if built) is never printed. However, if you do have a data index placeholder in the template but omit to specify the package name here, no content with data index will be printed in the output document (because it will not have been built).

IEC61850-specific validation properties

While CIM UML uses UML as its meta-model, and we generate the documentation for any CIM package and its elements the same way, IEC61850 has a pretty complex structure and the underlying meta-model. Almost every element of the IEC61850 concrete model needs different treatment in both UML and for document generation at present (in the future, still more to come for SCL modelling!). Consequently, validation of IEC61850 UML model and document generation from that model requires many special "hints" for the application to produce the desired format (and allow us to not hard-code these in the source code). The properties relevant for IEC61850 only, explicitly contain in their names "IEC61850". Strictly speaking, most of these options are mainly required for document generation only. However, the document gets generated from the UML model, and we want to ensure that we have performed validation of the model before generating documentation. Therefore, the majority of IEC61850-specific properties are applicable to both validation (if #KEY_VALIDATION_ON ="true") and document generation (if #KEY_DOCGEN_ON="true"). The values provided in the default #DEFAULT_PROPS_FILE_NAME file need not be modified, except if the packages get renamed in the UML model of IEC61850, or if WG10 decides to generate doc differently.

- Properties #KEY_VALIDATION_IEC61850_PACKAGES72, #KEY_VALIDATION_IEC61850_PACKAGES73 and #KEY_VALIDATION_IEC61850_PACKAGES74 must have as value a comma-separated list of names of IEC61850 sub-packages that have some special requirement for validation and doc generation of IEC61850-7-2, IEC61850-7-3 and IEC61850-7-4, respectively (for instance, tables for IEC61850-7-3 and IEC61850-7-4 have different format, because they document different elements of the meta-model, common data classes and logical nodes, respectively).
- Property #KEY_VALIDATION_IEC61850_PACKAGE_META_MODEL indicates the name of package where IEC61850 meta-model is defined: UML elements from this package must not be printed as inherited in the concrete definitions (tables) in IEC61850-7-3 and IEC61850-7-4.

- Property #KEY_VALIDATION_IEC61850_PACKAGE72_TOP indicates the name of top package where IEC6185007-2 is is defined: this is temporary thing, because we still don't have full and final 7-2 in UML, and we want to be able to skip validation of types from its sub-packages.
- Property #KEY_VALIDATION_IEC61850_PACKAGES_ENUMS_XML must have as value a comma-separated list of names of IEC61850 sub-packages that contain enumerations that are to be printed as XML (in addition to their normal printing as tables).
- Property #KEY_VALIDATION_IEC61850_PACKAGES_LN should have as value a comma-separated list of names of IEC61850 packages containing logical node classes (and other sub-packages). The packages and the classes typically have a short name in the UML model, but need a full name for the headings in the auto-generated documentation, and also some special heading formatting (e.g., heading for the XCBR class should look like "Logical node circuit breaker LNName: XCBR"). In UML, these human-readable names are defined as alias for the element. If no values are specified here, document generation runs normally, but those aliases will be simply ignored and the headings of the packages and classes will have just short names without any special formatting.
- Property #KEY_VALIDATION_IEC61850_PACKAGES_CDC is like #KEY_VALIDATION_IEC61850_PACKAGES_LN, but for common data classes.
- Property #KEY_VALIDATION_IEC61850_PACKAGES_DA is like #KEY_VALIDATION_IEC61850_PACKAGES_LN, but for constructed data attributes.
- Property #KEY_VALIDATION_IEC61850_PACKAGES_BASIC is like #KEY_VALIDATION_IEC61850_PACKAGES_LN, but for basic types.
- Property #KEY_VALIDATION_IEC61850_PACKAGE_PRES_COND must specify the name of the UML package that contains definition for presence conditions: defined in IEC61850-7-3 (in its own clause), and used to model the conditional presence of elements in both IEC61850-7-3 and IEC61850-7-4 (this is modelled in UML as constraints on classes).
- Property #KEY_VALIDATION_IEC61850_PACKAGE_FC must specify the name of the UML package that contains definition for functional constraints: defined in IEC61850-7-3 (in its own clause). This package is defined in IEC61850-7-***2***, but we must be able to generate Annex B in IEC61850-7-***3*** with that table.
- Property #KEY_VALIDATION_IEC61850_PACKAGE_TRGOP must specify the name of the UML package that contains definition for trigger options: defined in IEC61850-7-3 (in its own clause).
- Property #KEY_VALIDATION_IEC61850_PACKAGES_DO_ABBR should have as value a comma-separated list of names of IEC61850 packages containing definitions for valid abbreviations to use for names of data objects within logical nodes, and this table must be generated as clause 4 in IEC61850-7-4, -7-410 and -7-420. Also, we perform validation of data objects defined for logical nodes in UML (i.e. attribute names in concrete sub-classes of LN) and log errors.
- Property #KEY_VALIDATION_IEC61850_PACKAGE_LN_MAPS must specify the name of the UML package that contains (in its sub-packages) an extract from IEC61850-5, to be able to generate in IEC61850-7-4 a "special" clause with the table showing the mappings between LNs defined in requirements document (IEC61850-5) and actual normative LNs (IEC61850-7-4). In UML model, this info is contained as tagged values on classes representing IEC61850-5, and through dependencies between them and the actual LNs (IEC61850-7-4). If the value is left empty (or the property is not present at all), the content of that "special" clause in the generated Word document will be empty, even if there is a placeholder.

Statistics properties

These options control the *displaying* of statistics and are applicable only if value in property #KEY_STATISTICS_ON is set to "true".

There are two boolean options, both applicable to CIM only: #KEY_STATISTICS_CIM_IGNORE_ID_OBJECT_INHERITANCE and #KEY_STATISTICS_CIM_IGNORE_DOMAIN_CLASS_ATTRIBUTES. In CIM, almost every class inherits from IEC61970::Core::IdentifiedObject, and most of attributes have as a type some class from IEC61970::Domain package. These two options allow, when set to "true", to skip displaying these obvious cross-WG dependencies and avoid unnecessary noise in the output.

One more option has been added, applicable to any model family: #KEY_STATISTICS_TAGS_TO_IGNORE. You may want to use this in case you discover that the custom tool / add-on that you use heavily marks your model with tagged values and you don't want to end up with thousands of lines of log listing every single item in your model.

Document generation properties (MS Word or XML)

These options specify and control the generation of MS Word or XML document when property #KEY_DOCGEN_ON is set to "true". Depending on the value in #KEY_PROFILES_DOCGEN_ON, the document will be generated from the UML model as default (if #KEY_PROFILES_DOCGEN_ON="false", "", null), or from multiple profiles (if #KEY_PROFILES_DOCGEN_ON="true"; note that at present, this functionality is not implemented, but is planned for some of future releases).

Because MS Word document generation takes very long, we apply several optimisations depending on the value of #KEY_DOCGEN_ON. For instance, if it not enabled, we don't export diagrams from the UML model (reading model is almost

twice faster without exporting diagrams).

Several document generation properties are in fact used when collecting the content from the in-memory UML model, before outputing anything to files. Here are general document generation properties:

- Property #KEY_DOCGEN_INCLUDE_INFORMATIVE, if set "true", allows to include informative elements from UML model into generated document. By default ("false", "", null), these are skipped for document generation.
- Property #KEY_DOCGEN_INCLUDE_NON_PUBLIC, if set "true", allows to include private, package-private or protected UML elements into generated document. By default ("false", "", null), these are skipped for document generation.
- Property #KEY_DOCGEN_PRINT_HTML, if set "true", will allow to respect markup formatting in the documentation of elements in the UML repository when generating output documents. EA currently supports some simple markup (like italic, bold, underline, subscript, superscript, bulleted and numbered lists) in notes for diagram, package, class, attribute, operation, but not for association ends, constraints, operation parameters, tags etc. In CIM, we never use formated documentation, but it is heavilly used in IEC61850 model (e.g., for formulae). Consider that enabling fomatted output results in ~1.5 time longer MS Word document generation, while for XML output documents there is no impact to performance. (TODO: In this release of jCleanCim, this functionality is still not working properly for MS Word output documents, so you will not be happy with the output; this should be fixed in some of comming releases.)
- Property #KEY_DOCGEN_SHOW_CUSTOM_STEREOTYPES, if set "true", allows to show in the generated document custom UML stereotypes on UML model elements (in addition to built-in stereotypes already handled). By default ("false", "", null), these are skipped for document generation.
- Property #KEY_DOCGEN_SKIP_FOR_CUSTOM_STEREOTYPES holds a comma-separated list of custom UML stereotypes that you want to exclude from document generation. This list is matched against the built-in, standard stereotypes according to modelling rules and any built-in stereotype from this list is removed (i.e., no overriding of built-in stereotype consideration for document generation). Empty list is valid and reflects default behaviour, i.e., everything found in the model according to other specified filtering options is printed.
- Property #KEY_DOCGEN_SHOW_NAMESPACE_PACKAGES allows you to specify a comma-separated list of UML packages for which you want to explicitly print the namespace URI and prefix information, if existing.
- Property #KEY_DOCGEN_IEC61850_INCLUDE_METAMODEL_INHERITANCE, if set "true", allows for IEC61850 document generation, to include UML elements inherited from the IEC61850 meta-model package #KEY_VALIDATION_IEC61850_PACKAGE_META_MODEL. By default ("false", "", null), these are skipped for document generation.
- Property #KEY_DOCGEN_IEC61850_WRITE_UML_TYPES, if set "true", allows for IEC61850 document generation, to write documents in debug mode, i.e., instead of special processing of types for attributes of LNs, CDCs, DAs and the index tables for LNs and CDCs, this option makes to write the actual type, as it is in UML. This is useful for debugging only. By default ("false", "", null), document generation is for "real".

XML document generation properties

Starting with release 01v06, we started implementing support for printing UML model content to two XML files: so-called XML spec and XML doc. The first one is for all the content that is technically speaking specification (used for implementation), and the second one contains all the strings from the UML element description, special heading, caption or other titles, i.e., everything that needs translation. This decision has been taken in order to facilitate editing and translating process by IEC editors (once we move completely to web-based access and away from MS Word).

- Property #KEY_DOCGEN_XML_SCOPE, if not empty or absent, allows to filter the scope for generating XML documentation per IEC WG owning the top-level package, in IEC 61850 lingo, so-called name spaces. In contrast to MS Word documentation, we do not have any input template, so this is the means to select one or more namespaces, per WG. In CIM, we do not have name spaces, but we fabricate one per top level package (from the existing version class).

MS Word document generation properties

- Property #KEY_DOCGEN_WORD_ANALYSE_PLACEHOLDERS, if set "true", allows you to only analyse ("validate") your input template and get hint on errors. This is useful when e.g. updating template with placeholders for new diagrams: if you have a typing error and specify the value in the placeholder which does not exist in the UML model, an output MS Word document will be generated by replacing the placeholders *not* with real content from UML, but with the actual names from UML that would be used; or with ERROR description in case the placeholder value is invalid. This is very handy to run if you've updated the template, but before actually generating the full documentation (which takes long!) - the produced skeleton output document, if not containg "ERROR" indicates that all the placeholders are OK.
- Property #KEY_DOCGEN_WORD_SAVE_REOPEN_EVERY gives the number of tables (and implicitly, table captions) to write before saving, closing and reopening the auto-generated document. This is new option, introduced in release 01v05, to improve performance of MS Word document generation for extremely large documents (i.e., those that have more than ~200 tables). Ensure you set this value as indicated in the readme file when generating very large documentation! MS Word's "insertCaption" method, which is the only means to automatically have numbered figures, tables and their tables of contents,

slows down exponentially with the number of captions inserted (in particular, for tables). The only way we found to speed this up is to save the document at #KEY_DOCGEN_WORD_SAVE_REOPEN_EVERY number of tables, reopen it and continue printing from there. Default value is -1, which means no close/reopen will happen. If you set it to 0, it will perform close/reopen just after writing the first table, then never anymore. Any value greater than zero is applied to all the tables generated.
- Property #KEY_DOCGEN_WORD_USE_DOC_FORMAT, when enabled, forces usage of the slow COM API for MS Word. This mimics the original implementation, before we have provided much faster one, assuming we can always work with the OpenXML (.docx) MS Word documents. (TODO: OpenXML implementation not yet functional, so you have to set this to true for the time being...)
- Property #KEY_DOCGEN_WORD_INTRO_TO_FIGURE_BEFORE, when enabled, preserves original way of printing introduction to figures first in MS Word document generation, as opposed to referring to the figure caption below the figure and its caption. This latter is the new default behaviour.
- Property #KEY_DOCGEN_WORD_USE_HYPERLINKS, if set "true", will allow to hyperlink to the chapter describing the type for attributes, association ends and operation parameters. Consider that enabling fomatted output results in longer MS Word document generation!
- Property #KEY_DOCGEN_WORD_INCLUDE_INHERITANCE_PATH, if set "true", will include as a part of class documentation, immediately after the title, the class' inheritance path, i.e., all its superclasses (if applicable).
- Property #KEY_DOCGEN_WORD_STYLES_PREFIX_TOC / #KEY_DOCGEN_WORD_STYLES_PREFIX_TOC is a comma-separated list of single-word prefixes for TOC/heading style names. If none specified, default #DEFAULT_STYLES_PREFIX_TOC / #DEFAULT_STYLES_PREFIX_HEAD is used. When scanning existing TOC/heading styles, their names are matched also against this list. For writing, TOC/heading style names must be read from the document, but the retained style is matched if possible according to order and values in this list.
- Property #KEY_DOCGEN_WORD_STYLES_PARA / #KEY_DOCGEN_WORD_STYLES_FIG / #KEY_DOCGEN_WORD_STYLES_TABHEAD / #KEY_DOCGEN_WORD_STYLES_TABCELL / #KEY_DOCGEN_WORD_STYLES_FIGCAPT / #KEY_DOCGEN_WORD_STYLES_TABCAPT is a comma-separated list of single-word prefixes for TOC/heading style names. If none specified, default #DEFAULT_STYLES_PARA / #DEFAULT_STYLES_FIG / #DEFAULT_STYLES_TABHEAD / #DEFAULT_STYLES_TABCELL / #DEFAULT_STYLES_FIGCAPT / #DEFAULT_STYLES_TABCAPT is used. When scanning existing TOC/heading styles, their names are matched also against this list. For writing, TOC/heading style names must be read from the document, but the retained style is matched if possible according to order and values in this list.

MIBs generation properties (for IEC62351-7)

(Since 02v03) These options specify and control the generation of MIBs when property #KEY_MIBGEN_ON is set to "true". It can be enabled simultaneously with the MS Word document generation (#KEY_DOCGEN_ON = true) if both MIBs and MS Word document generation is wanted.

With properties #KEY_MIBGEN_OUT_DIR_FULL and #KEY_MIBGEN_OUT_DIR_LIGHT you specify the directories where to put the resulting full and light MIBs, respectively. If these properties are not present or empty, default directory names `DEFAULT_MIBS_OUT_DIRNAME` and `DEFAULT_MIBSLIGHT_OUT_DIRNAME` will be used, respectively. In every case, these directories will be created under the default output directory: `OUTPUT_DIR_NAME`.

CIM-profile document generation properties

Currently, there are no properties for this use case, except for the top-property #KEY_PROFILES_DOCGEN_ON; note that at present, this functionality is not implemented.

Field Summary

`public static final`	`DEFAULT_BLANK_PNG_FILENAME` Empty image, used when not storing diagrams into files. Value: **`blank.png`**
`public static final`	`DEFAULT_MIBS_OUT_DIRNAME` Default directory name for full MIBs. Value: **`mibs`**
`public static final`	`DEFAULT_MIBSLIGHT_OUT_DIRNAME` Default directory name for light MIBs. Value: **`mibslight`**

public static final	DEFAULT_OUT_XML_DOC_FILENAME Default file name for XML doc output document (generated from a model). Value: **base-small-doc.xml**
public static final	DEFAULT_OUT_XML_SPEC_FILENAME Default file name for XML spec output document (generated from a model). Value: **base-small-spec.xml**
public static final	DEFAULT_PROFILES_RELPATH Default value for
public static final	DEFAULT_PROPS_FILE_NAME Application configuration properties, to be edited by the user. Value: **config.properties**
public static final	DEFAULT_STYLES_FIG Default style name for figure, if none provided. Value: **FIGURE**
public static final	DEFAULT_STYLES_FIGCAPT Default style name for figure caption, if none provided. Value: **FIGURE-title**
public static final	DEFAULT_STYLES_PARA Default style name for text paragraph, if none provided. Value: **PARAGRAPH**
public static final	DEFAULT_STYLES_PREFIX_HEAD Default prefix (style name without number) for heading styles, if none provided. Value: **Heading**
public static final	DEFAULT_STYLES_PREFIX_TOC Default prefix (style name without number) for TOC styles, if none provided. Value: **TOC**
public static final	DEFAULT_STYLES_TABCAPT Default style name for table caption, if none provided. Value: **TABLE-title**
public static final	DEFAULT_STYLES_TABCELL Default style name for table cell, if none provided. Value: **TABLE-cell**
public static final	DEFAULT_STYLES_TABHEAD Default style name for table heading, if none provided. Value: **TABLE-col-heading**
public static final	DEFAULT_VERSION_PROP_NAME Project version property name (for use by ant script), and if not available through the jar manifest, available through . Value: **project.version**
public static final	DEFAULT_WEBACCESS_SCHEMA_FILENAME Searched on classpath. Value: **IECDomain.xsd**

public static final	DEFAULT_WORD_IN_TEMPLATE_FILENAME Default file name for MS Word input template. Value: **base-small-template.docx**
public static final	DEFAULT_WORD_OUT_DOCUMENT_FILENAME Default file name for MS Word output document (generated from input template and a model). Value: **base-small.docx**
public static final	IEC61850_PROPS_FILE_NAME IEC 61850-specific application configuration properties, to be edited by the user. Value: **config61850.properties**
public static final	INPUT_DIR_NAME Default input directory, set to be on the classpath. Value: **input**
public static final	KEY_APP_SKIP_TIMING App configuration: Skip logging ellapsed time = "true" (default = "false", "", null). Value: **app.skipTiming**
public static final	KEY_DOCGEN_IEC61850_INCLUDE_METAMODEL_INHERITANCE Write inheritance from IEC61850 UML meta-model KEY_VALIDATION_IEC61850_PACKAGE_META_MODEL = "true" (default = "false", "", null). Value: **docgen.iec61850.includeMetamodelInheritance**
public static final	KEY_DOCGEN_IEC61850_WRITE_UML_TYPES Write types with their real names, as they are in IEC61850 UML = "true" (default = "false", "", null). Value: **docgen.iec61850.writeUmlTypes**
public static final	KEY_DOCGEN_INCLUDE_INFORMATIVE Include informative UML model elements in the generated document = "true" (default = "false", "", null). Value: **docgen.includeInformative**
public static final	KEY_DOCGEN_INCLUDE_NON_PUBLIC Include non-public UML model elements in the generated document = "true" (default = "false", "", null). Value: **docgen.includeNonPublic**
public static final	KEY_DOCGEN_ON Top-level functionality: Enable document generation = "true" (default = "false", "", null). Value: **docgen.on**
public static final	KEY_DOCGEN_PRINT_HTML Enable formatted documentation of UML model elements in the generated document = "true" (default = "false", "", null). Value: **docgen.printHtml**
public static final	KEY_DOCGEN_SHOW_CUSTOM_STEREOTYPES Enable showing custom UML stereotypes on UML model elements in the generated document = "true" (default = "false", "", null). Value: **docgen.showCustomStereotypes**

public static final	KEY_DOCGEN_SHOW_NAMESPACE_PACKAGES Comma-separated list of packages for which to explicitly show namespace and URI (if known) in the auto-generated document. Value: **docgen.showNamespacePackages**
public static final	KEY_DOCGEN_SKIP_FOR_CUSTOM_STEREOTYPES Comma-separated list of custom stereotypes which if used, their elements are to be skipped when generating documents. Value: **docgen.skipForCustomStereotypes**
public static final	KEY_DOCGEN_WORD_ANALYSE_PLACEHOLDERS Enable dry run of MS Word document generation = "true" (default = "false", "", null). Value: **docgen.word.analysePlaceholders**
public static final	KEY_DOCGEN_WORD_IN_TEMPLATE File name of the (input) MS Word document template, expected to be found on the classpath. Value: **docgen.word.inTemplate**
public static final	KEY_DOCGEN_WORD_INCLUDE_INHERITANCE_PATH Include inheritance paths when printing classes in the generated document in the generated MS Word document = "true" (default = "false", "", null). Value: **docgen.word.includeInheritancePath**
public static final	KEY_DOCGEN_WORD_INTRO_TO_FIGURE_BEFORE Preserves original way of printing introduction to figures first in MS Word document generation = "true" (default = "false", "", null), as opposed to referring to the figure caption below the figure and its caption. Value: **docgen.word.introToFigureBefore**
public static final	KEY_DOCGEN_WORD_OUT_DOCUMENT File name of the (output) MS Word generated document; will be created in #OUTPUT_DIR_NAME from template #KEY_DOCGEN_WORD_IN_TEMPLATE and the UML, profile or in-memory model. Value: **docgen.word.outDocument**
public static final	KEY_DOCGEN_WORD_SAVE_REOPEN_EVERY The number of tables (and table captions) to write before closing and reopening the document. Value: **docgen.word.saveReopenEvery**
public static final	KEY_DOCGEN_WORD_STYLES_FIG Comma-separated list of single-word style names for figures, in order of preference. Value: **docgen.word.styles.fig**
public static final	KEY_DOCGEN_WORD_STYLES_FIGCAPT Comma-separated list of single-word style names for figure caption, in order of preference. Value: **docgen.word.styles.figcapt**
public static final	KEY_DOCGEN_WORD_STYLES_PARA Comma-separated list of single-word style names for text paragraphs, in order of preference. Value: **docgen.word.styles.para**
public static final	KEY_DOCGEN_WORD_STYLES_PREFIX_HEAD Comma-separated list of Heading style name prefixes (style name without number), in order of preference. Value: **docgen.word.styles.prefix.head**

public static final	KEY_DOCGEN_WORD_STYLES_PREFIX_TOC Comma-separated list of TOC style name prefixes (style name without number), in order of preference. Value: **docgen.word.styles.prefix.toc**
public static final	KEY_DOCGEN_WORD_STYLES_TABCAPT Comma-separated list of single-word style names for table caption, in order of preference. Value: **docgen.word.styles.tabcapt**
public static final	KEY_DOCGEN_WORD_STYLES_TABCELL Comma-separated list of single-word style names for table cells, in order of preference. Value: **docgen.word.styles.tabcell**
public static final	KEY_DOCGEN_WORD_STYLES_TABHEAD Comma-separated list of single-word style names for table headings, in order of preference. Value: **docgen.word.styles.tabhead**
public static final	KEY_DOCGEN_WORD_USE_DOC_FORMAT Force MS Word COM (.doc) document generation = "true" (default = "false", "", null), as opposed to the Open XML (.docx) format. Value: **docgen.word.useDocFormat**
public static final	KEY_DOCGEN_WORD_USE_HYPERLINKS Enable hyperlinking of UML model elements in the generated MS Word document = "true" (default = "false", "", null). Value: **docgen.word.useHyperlinks**
public static final	KEY_DOCGEN_XML_OUT_DOC File name of the (output) XML generated documentation (translatable) document; will be created in #OUTPUT_DIR_NAME from the UML, profile or in-memory model. Value: **docgen.xml.outDoc**
public static final	KEY_DOCGEN_XML_OUT_SPEC File name of the (output) XML generated specification document; will be created in #OUTPUT_DIR_NAME from the UML, profile or in-memory model. Value: **docgen.xml.outSpec**
public static final	KEY_DOCGEN_XML_SCOPE Comma-separated list of values corresponding to literals in OwningWg; empty value (default) takes them all. Value: **docgen.xml.scope**
public static final	KEY_MIBGEN_ON Top-level functionality: Enable MIBs generation = "true" (default = "false", "", null). Value: **mibgen.on**
public static final	KEY_MIBGEN_OUT_DIR_FULL Directory name for the (output) full MIBs; will be created in #OUTPUT_DIR_NAME from the UML or in-memory model. Value: **mibgen.outDirFull**
public static final	KEY_MIBGEN_OUT_DIR_LIGHT Directory name for the (output) light MIBs; will be created in #OUTPUT_DIR_NAME from the UML or in-memory model. Value: **mibgen.outDirLight**

public static final	KEY_MODEL_BUILDER One of enumeration literals in ModelBuilderKind (def = ModelBuilderKind.db). Value: **model.builder**
public static final	KEY_MODEL_FILENAME File name of a UML repository (model), expected to be found on the classpath. Value: **model.filename**
public static final	KEY_MODEL_NATURE_IEC61850 Comma-separated list of model packages (directly below the root) that are IEC61850, or derive from it. Value: **model.nature.iec61850**
public static final	KEY_PROFILES_CROSSCHECK_ON Top-level functionality: Enable crosscheck between the UML model and a set of profiles = "true" (default = "false", "", null). Value: **profiles.crosscheck.on**
public static final	KEY_PROFILES_DIRNAMES Comma-separated list of values corresponding to literals in OwningWg; empty value (default) takes them all. Value: **profiles.dirnames**
public static final	KEY_PROFILES_DOCGEN_ON Top-level functionality: Enable document generation for profiles = "true" (default = "false", "", null), if also #KEY_DOCGEN_ON="true". Value: **profiles.docgen.on**
public static final	KEY_PROFILES_RELPATH Relative path of directory storing profiles; default is DEFAULT_PROFILES_RELPATH. Value: **profiles.relpath**
public static final	KEY_STATISTICS_CIM_IGNORE_DOMAIN_CLASS_ATTRIBUTES (if #KEY_STATISTICS_ON="true"): Skip logging to console dependencies through usage of types from CIM Domain package in attributes = "true" (default = "false", "", null). Value: **statistics.cim.ignoreDomainClassAttributes**
public static final	KEY_STATISTICS_CIM_IGNORE_ID_OBJECT_INHERITANCE (if #KEY_STATISTICS_ON="true"): Skip logging to console dependencies through inheritance from CIM IdentifiedObject = "true" (default = "false", "", null). Value: **statistics.cim.ignoreIdObjectInheritance**
public static final	KEY_STATISTICS_ON Top-level functionality: Enable statistics = "true" (default = "false", "", null). Value: **statistics.on**
public static final	KEY_STATISTICS_TAGS_TO_IGNORE (if #KEY_STATISTICS_ON="true"): Comma-separated list of tagged values for which to skip logging to console. Value: **statistics.tagsToIgnore**
public static final	KEY_VALIDATION_ASSOCIATIONS_OFF Skip all validation rules for associations and thier ends = "true" (default = "false", "", null). Value: **validation.associations.off**
public static final	KEY_VALIDATION_ATTRIBUTES_OFF Skip all validation rules for attributes = "true" (default = "false", "", null). Value: **validation.attributes.off**

public static final	KEY_VALIDATION_CLASSES_OFF Skip all validation rules for classes = "true" (default = "false", "", null). Value: **validation.classes.off**
public static final	KEY_VALIDATION_DEPENDENCIES_OFF Skip all validation rules for (hand-drawn UML) dependencies = "true" (default = "false", "", null). Value: **validation.dependencies.off**
public static final	KEY_VALIDATION_DIAGRAMS_OFF Skip all validation rules for diagrams = "true" (default = "false", "", null). Value: **validation.diagrams.off**
public static final	KEY_VALIDATION_IEC61850_PACKAGE_FC Name of the UML package where the functional constraints of IEC61850 are defined. Value: **validation.iec61850.packageFC**
public static final	KEY_VALIDATION_IEC61850_PACKAGE_LN_MAPS Name of the UML package where the requirements specification for logical nodes (IEC61850-5) is defined. Value: **validation.iec61850.packageLnMaps**
public static final	KEY_VALIDATION_IEC61850_PACKAGE_META_MODEL Name of the UML package where the meta-model of IEC61850 is defined. Value: **validation.iec61850.packageMetaModel**
public static final	KEY_VALIDATION_IEC61850_PACKAGE_PRES_COND Name of the UML package where the presence conditions of IEC61850 are defined. Value: **validation.iec61850.packagePresCond**
public static final	KEY_VALIDATION_IEC61850_PACKAGE_TRGOP Name of the UML package where the trigger options of IEC61850 are defined. Value: **validation.iec61850.packageTrgOp**
public static final	KEY_VALIDATION_IEC61850_PACKAGE72_TOP Name of the UML package where the meta-model of IEC61850 is defined. Value: **validation.iec61850.package72Top**
public static final	KEY_VALIDATION_IEC61850_PACKAGES_BASIC Comma-separated list of package names whose all deep-child elements (both sub-packages and classes) should contain a human-readable name and may need special table formatting in IEC61850-7-2. Value: **validation.iec61850.packagesBasic**
public static final	KEY_VALIDATION_IEC61850_PACKAGES_CDC Comma-separated list of package names whose all deep-child elements (both sub-packages and classes) should contain a human-readable name and apply special formatting for headings in IEC61850-7-3. Value: **validation.iec61850.packagesCdc**
public static final	KEY_VALIDATION_IEC61850_PACKAGES_DA Comma-separated list of package names whose all deep-child elements (both sub-packages and classes) should contain a human-readable name and may need special table formatting in IEC61850-7-3. Value: **validation.iec61850.packagesDa**

public static final	KEY_VALIDATION_IEC61850_PACKAGES_DO_ABBR Name of the UML package where the abbreviations for data object names in IEC61850 are defined. Value: **validation.iec61850.packagesDoAbbr**
public static final	KEY_VALIDATION_IEC61850_PACKAGES_ENUMS_XML Comma-separated list of package names containing enumerations that must be printed as XML (in addition to tables) in IEC61850-7-3 and IEC61850-7-4. Value: **validation.iec61850.packagesEnumsXml**
public static final	KEY_VALIDATION_IEC61850_PACKAGES_LN Comma-separated list of package names whose all deep-child elements (both sub-packages and classes) should contain a human-readable name and apply special formatting for headings in IEC61850-7-4. Value: **validation.iec61850.packagesLn**
public static final	KEY_VALIDATION_IEC61850_PACKAGES72 Comma-separated list of package names required for generation of main clauses in IEC61850-7-2. Value: **validation.iec61850.packages72**
public static final	KEY_VALIDATION_IEC61850_PACKAGES73 Comma-separated list of package names required for generation of main clauses in IEC61850-7-3. Value: **validation.iec61850.packages73**
public static final	KEY_VALIDATION_IEC61850_PACKAGES74 Comma-separated list of package names required for generation of main clauses in IEC61850-7-4. Value: **validation.iec61850.packages74**
public static final	KEY_VALIDATION_LOGGING_VERBOSE Log to console also validation steps with no errors = "true" (default = "false", "", null). Value: **validation.logging.verbose**
public static final	KEY_VALIDATION_ON Top-level functionality: Enable validation = "true" (default = "false", "", null). Value: **validation.on**
public static final	KEY_VALIDATION_OPERATIONS_OFF Skip all validation rules for operations and their parameters = "true" (default = "false", "", null). Value: **validation.operations.off**
public static final	KEY_VALIDATION_PACKAGES_DATA_INDEX Comma-separated list of package names required for building data index from all the attributes on classes from the given package and below, recursively. Value: **validation.packagesDataIndex**
public static final	KEY_VALIDATION_PACKAGES_OFF Skip all validation rules for packages = "true" (default = "false", "", null). Value: **validation.packages.off**
public static final	KEY_VALIDATION_RULES_OFF Comma-separated list of individual rule class names to be skipped during validation. Value: **validation.rules.off**

public static final	KEY_VALIDATION_SCOPE Comma-separated list of values corresponding to literals in OwningWg; empty value (default) takes them all. Value: **validation.scope**
public static final	KEY_XMIEXPORT_DIALECTS Comma-separated list of values corresponding to literals in XMIDialect; empty value (default) takes them all. Value: **xmiexport.dialects**
public static final	KEY_XMIEXPORT_ON Top-level functionality: Enable XMI export = "true" (default = "false", "", null). Value: **xmiexport.on**
public static final	MODEL_PICS_RELPATH Directory path for exported diagram images.
public static final	OUTPUT_DIR_NAME Directory into which to create output files. Value: **output**
public static final	PICS_DIR_NAME Directory name for exported diagram images. Value: **pics**
public static final	PROFILES_DIR_NAME Directory under classpath below which to search for profile files. Value: **profiles**
public static final	VERSION_PROPS_FILENAME Build properties (for use by ant script), containing also the version information. Value: **build.properties**
public static final	XSD_EXT Supported profile format (file extension). Value: **xsd**

Constructor Summary

public	Config(java.lang.String propsFilename, java.lang.String modelFilename) Constructor.
public	Config(java.util.Properties props, java.lang.String modelFilename) Constructor; useful for testing.

Method Summary

static java.lang.String	deduceAppVersion()
java.lang.String	getAppVersion() Returns application version deduced from manifest (if running with a jar), or read from VERSION_PROPS_FILENAME file otherwise.
java.lang.String	getBlankPngFileAbsPath() Returns absolute path of the default image file, to be used as replacement when the "real" image is not available.

java.util.List	getDocgenShowNamespacePackages()
java.util.List	getDocgenSkipForCustomStereotypes()
java.lang.String	getDocgenWordInTemplateFileAbsPath() Returns absolute path of input Word template file, null if doc generation disabled or output files are given in another format.
java.lang.String	getDocgenWordOutDocumentFileAbsPath() Returns absolute path of output Word document file, null if doc generation disabled or output files are given in another format.
int	getDocgenWordSaveReopenEvery() In case an integer cannot be parsed, returns -1; otherwise, an absolute value of #KEY_DOCGEN_WORD_SAVE_REOPEN_EVERY.
java.util.List	getDocgenWordStylesFig() Returns non-empty list of user-prioritised names for figure styles.
java.util.List	getDocgenWordStylesFigcapt() Returns non-empty list of user-prioritised names for figure caption styles.
java.util.List	getDocgenWordStylesPara() Returns non-empty list of user-prioritised names for paragraph text styles.
java.util.List	getDocgenWordStylesPrefixHead() Returns non-empty list of user-prioritised prefixes for heading style names.
java.util.List	getDocgenWordStylesPrefixToc() Returns non-empty list of user-prioritised prefixes for TOC style names.
java.util.List	getDocgenWordStylesTabcapt() Returns non-empty list of user-prioritised names for table caption styles.
java.util.List	getDocgenWordStylesTabcell() Returns non-empty list of user-prioritised names for table cells styles.
java.util.List	getDocgenWordStylesTabhead() Returns non-empty list of user-prioritised names for table heading styles.
java.lang.String	getDocgenXmlOutDocFileAbsPath() Returns absolute path of output XML doc file, null if doc generation disabled or output files are given in another format.
java.lang.String	getDocgenXmlOutSpecFileAbsPath() Returns absolute path of output XML spec file, null if doc generation disabled or output files are given in another format.
java.util.EnumSet	getDocgenXmlScope() Returns the owners of packages that determine the scope of XML generation.
java.lang.String	getDocgenXsdInWebaccessFileAbsPath() Returns absolute path of XML web access schema file, null if doc generation disabled or output files are given in another format.
java.lang.String	getDocgenXsdOutWebaccessFileAbsPath() Returns absolute path of where to copy the XML web access schema, null if doc generation disabled or output files are given in another format.

java.util.List	getIec61850NaturePackages() Comma-separated list of names of model packages (below the root) with non-CIM nature.
java.lang.String	getMibgenOutDirFullAbsPath() Returns absolute path of output for full MIBs, null if doc generation disabled or output files are given in another format.
java.lang.String	getMibgenOutDirLightAbsPath() Returns absolute path of output for light MIBs, null if doc generation disabled or output files are given in another format.
ModelBuilderKind	getModelBuilder() Returns whether to use SQL to build model.
java.lang.String	getModelFileAbsPath() Returns absolute path of the UML model file.
java.lang.String	getPicsDirAbsPath() Returns absolute path string for directory where to export images from the model.
java.util.Map	getProfileFiles()
java.lang.String	getProfilesRelpath()
java.util.Collection	getStatisticsTagsToIgnore()
java.lang.String	getValidationIec61850Package72Top()
java.lang.String	getValidationIec61850PackageFc()
java.lang.String	getValidationIec61850PackageLnMaps()
java.lang.String	getValidationIec61850PackageMetaModel()
java.lang.String	getValidationIec61850PackagePresCond()
java.util.Collection	getValidationIec61850Packages72()
java.util.Collection	getValidationIec61850Packages73()
java.util.Collection	getValidationIec61850Packages74()
java.util.Collection	getValidationIec61850PackagesBasic()
java.util.Collection	getValidationIec61850PackagesCdc()
java.util.Collection	getValidationIec61850PackagesDa()
java.util.Collection	getValidationIec61850PackagesDoAbbr()

java.util.Collection	getValidationIec61850PackagesDocgen() Returns the union of getValidationIec61850Packages72(), getValidationIec61850Packages73() and getValidationIec61850Packages74().
java.util.Collection	getValidationIec61850PackagesEnumsXml()
java.util.Collection	getValidationIec61850PackagesExtTitle() Returns the union of getValidationIec61850PackagesLn(), getValidationIec61850PackagesCdc() and .
java.util.Collection	getValidationIec61850PackagesLn()
java.lang.String	getValidationIec61850PackageTrgOp()
java.util.Collection	getValidationPackagesDataIndex()
java.util.Collection	getValidationRulesOff()
java.util.EnumSet	getValidationScope() Returns the owners of packages that determine the scope of validation and statistics.
java.util.EnumSet	getXmiexportDialects() Returns the configured XMI dialects to be used for export.
boolean	hasUmlModel() Returns true when EA file is specified in configuration.
boolean	isAppSkipTiming() Returns whether to skip logging ellapsed time.
boolean	isDocgenIec61850IncludeMetamodelInheritance()
boolean	isDocgenIec61850WriteUmlTypes()
boolean	isDocgenIncludeInformative()
boolean	isDocgenIncludeNonPublic()
boolean	isDocgenModelOn() Returns true if only docgen from EA is enabled (but not profile docgen).
boolean	isDocgenOn()
boolean	isDocgenPrintHtml() Returns whether to respect markup (present in UML descriptions) in output document.
boolean	isDocgenShowCustomStereotypes()
boolean	isDocgenWordAnalysePlaceholders() Returns whether to only analyse placeholders in output MS Word document, without replacing them with the full content.
boolean	isDocgenWordIncludeInheritancePath() Returns whether to include inheritance path in output MS Word document.

boolean	isDocgenWordIntroToFigureBefore()
boolean	isDocgenWordUseDocFormat()
boolean	isDocgenWordUseHyperlinks() Returns whether to use hyperlinks in output MS Word document.
boolean	isMibgenOn()
boolean	isProfilesCrosscheckOn()
boolean	isProfilesDocgenOn() Returns true if both general and profile docgen are enabled.
boolean	isRemovePicsAfterExit() Returns whether to remove exported diagrams at application exit (true if MS Word output is to be generated).
boolean	isStatisticsCimIgnoreDomainClassAttributes()
boolean	isStatisticsCimIgnoreIdObjectInheritance()
boolean	isStatisticsOn()
boolean	isValidationAssociationsOn()
boolean	isValidationAttributesOn()
boolean	isValidationClassesOn()
boolean	isValidationDependenciesOn()
boolean	isValidationDiagramsOn()
boolean	isValidationLoggingVerbose()
boolean	isValidationOn()
boolean	isValidationOperationsOn()
boolean	isValidationPackagesOn()
boolean	isXmiexportOn() Returns whether export to XMI is enabled; applicable only if the source of the model is an .eap file.
java.lang.String	toString()

Methods inherited from class `java.lang.Object`

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Fields

KEY_APP_SKIP_TIMING

`public static final java.lang.String` **`KEY_APP_SKIP_TIMING`**

App configuration: Skip logging ellapsed time = "true" (default = "false", "", null).
Constant value: **`app.skipTiming`**

KEY_MODEL_FILENAME

`public static final java.lang.String` **`KEY_MODEL_FILENAME`**

File name of a UML repository (model), expected to be found on the classpath.
Constant value: **`model.filename`**

KEY_MODEL_NATURE_IEC61850

`public static final java.lang.String` **`KEY_MODEL_NATURE_IEC61850`**

Comma-separated list of model packages (directly below the root) that are IEC61850, or derive from it.
Constant value: **`model.nature.iec61850`**

KEY_MODEL_BUILDER

`public static final java.lang.String` **`KEY_MODEL_BUILDER`**

One of enumeration literals in `ModelBuilderKind` (def = `ModelBuilderKind.db`).
Constant value: **`model.builder`**

KEY_PROFILES_RELPATH

`public static final java.lang.String` **`KEY_PROFILES_RELPATH`**

Relative path of directory storing profiles; default is `DEFAULT_PROFILES_RELPATH`.
Constant value: **`profiles.relpath`**

KEY_PROFILES_DIRNAMES

`public static final java.lang.String` **`KEY_PROFILES_DIRNAMES`**

Comma-separated list of values corresponding to literals in `OwningWg`; empty value (default) takes them all. These correspond to profile subdirectories, under #KEY_PROFILES_RELPATH, below which the profile files are located. The file extension that is recognised is #XSD_EXT .
Constant value: **`profiles.dirnames`**

KEY_XMIEXPORT_ON

`public static final java.lang.String` **`KEY_XMIEXPORT_ON`**

Top-level functionality: Enable XMI export = "true" (default = "false", "", null).
Constant value: **`xmiexport.on`**

KEY_XMIEXPORT_DIALECTS

`public static final java.lang.String` **`KEY_XMIEXPORT_DIALECTS`**

(continued from last page)

Comma-separated list of values corresponding to literals in `XMIDialect`; empty value (default) takes them all.
Constant value: **`xmiexport.dialects`**

KEY_VALIDATION_ON

`public static final java.lang.String` **`KEY_VALIDATION_ON`**

Top-level functionality: Enable validation = "true" (default = "false", "", null).
Constant value: **`validation.on`**

KEY_VALIDATION_SCOPE

`public static final java.lang.String` **`KEY_VALIDATION_SCOPE`**

Comma-separated list of values corresponding to literals in `OwningWg`; empty value (default) takes them all.
Constant value: **`validation.scope`**

KEY_VALIDATION_PACKAGES_OFF

`public static final java.lang.String` **`KEY_VALIDATION_PACKAGES_OFF`**

Skip all validation rules for packages = "true" (default = "false", "", null).
Constant value: **`validation.packages.off`**

KEY_VALIDATION_CLASSES_OFF

`public static final java.lang.String` **`KEY_VALIDATION_CLASSES_OFF`**

Skip all validation rules for classes = "true" (default = "false", "", null).
Constant value: **`validation.classes.off`**

KEY_VALIDATION_ASSOCIATIONS_OFF

`public static final java.lang.String` **`KEY_VALIDATION_ASSOCIATIONS_OFF`**

Skip all validation rules for associations and thier ends = "true" (default = "false", "", null).
Constant value: **`validation.associations.off`**

KEY_VALIDATION_ATTRIBUTES_OFF

`public static final java.lang.String` **`KEY_VALIDATION_ATTRIBUTES_OFF`**

Skip all validation rules for attributes = "true" (default = "false", "", null).
Constant value: **`validation.attributes.off`**

KEY_VALIDATION_OPERATIONS_OFF

`public static final java.lang.String` **`KEY_VALIDATION_OPERATIONS_OFF`**

Skip all validation rules for operations and their parameters = "true" (default = "false", "", null).
Constant value: **`validation.operations.off`**

KEY_VALIDATION_DEPENDENCIES_OFF

`public static final java.lang.String` **`KEY_VALIDATION_DEPENDENCIES_OFF`**

Skip all validation rules for (hand-drawn UML) dependencies = "true" (default = "false", "", null).
Constant value: **`validation.dependencies.off`**

KEY_VALIDATION_DIAGRAMS_OFF

`public static final java.lang.String` **`KEY_VALIDATION_DIAGRAMS_OFF`**

Skip all validation rules for diagrams = "true" (default = "false", "", null).
Constant value: **`validation.diagrams.off`**

KEY_VALIDATION_RULES_OFF

`public static final java.lang.String` **`KEY_VALIDATION_RULES_OFF`**

Comma-separated list of individual rule class names to be skipped during validation.
Constant value: **`validation.rules.off`**

KEY_VALIDATION_LOGGING_VERBOSE

`public static final java.lang.String` **`KEY_VALIDATION_LOGGING_VERBOSE`**

Log to console also validation steps with no errors = "true" (default = "false", "", null).
Constant value: **`validation.logging.verbose`**

KEY_VALIDATION_PACKAGES_DATA_INDEX

`public static final java.lang.String` **`KEY_VALIDATION_PACKAGES_DATA_INDEX`**

Comma-separated list of package names required for building data index from all the attributes on classes from the given package and below, recursively.
Constant value: **`validation.packagesDataIndex`**

KEY_VALIDATION_IEC61850_PACKAGES72

`public static final java.lang.String` **`KEY_VALIDATION_IEC61850_PACKAGES72`**

Comma-separated list of package names required for generation of main clauses in IEC61850-7-2.
Constant value: **`validation.iec61850.packages72`**

KEY_VALIDATION_IEC61850_PACKAGES73

`public static final java.lang.String` **`KEY_VALIDATION_IEC61850_PACKAGES73`**

Comma-separated list of package names required for generation of main clauses in IEC61850-7-3.
Constant value: **`validation.iec61850.packages73`**

KEY_VALIDATION_IEC61850_PACKAGES74

`public static final java.lang.String` **`KEY_VALIDATION_IEC61850_PACKAGES74`**

Comma-separated list of package names required for generation of main clauses in IEC61850-7-4.
Constant value: **`validation.iec61850.packages74`**

KEY_VALIDATION_IEC61850_PACKAGE_META_MODEL

`public static final java.lang.String` **`KEY_VALIDATION_IEC61850_PACKAGE_META_MODEL`**

Name of the UML package where the meta-model of IEC61850 is defined.
Constant value: **`validation.iec61850.packageMetaModel`**

(continued from last page)

KEY_VALIDATION_IEC61850_PACKAGE72_TOP

`public static final java.lang.String` **`KEY_VALIDATION_IEC61850_PACKAGE72_TOP`**

Name of the UML package where the meta-model of IEC61850 is defined.
Constant value: **`validation.iec61850.package72Top`**

KEY_VALIDATION_IEC61850_PACKAGES_ENUMS_XML

`public static final java.lang.String` **`KEY_VALIDATION_IEC61850_PACKAGES_ENUMS_XML`**

Comma-separated list of package names containing enumerations that must be printed as XML (in addition to tables) in IEC61850-7-3 and IEC61850-7-4.
Constant value: **`validation.iec61850.packagesEnumsXml`**

KEY_VALIDATION_IEC61850_PACKAGES_LN

`public static final java.lang.String` **`KEY_VALIDATION_IEC61850_PACKAGES_LN`**

Comma-separated list of package names whose all deep-child elements (both sub-packages and classes) should contain a human-readable name and apply special formatting for headings in IEC61850-7-4.
Constant value: **`validation.iec61850.packagesLn`**

KEY_VALIDATION_IEC61850_PACKAGES_CDC

`public static final java.lang.String` **`KEY_VALIDATION_IEC61850_PACKAGES_CDC`**

Comma-separated list of package names whose all deep-child elements (both sub-packages and classes) should contain a human-readable name and apply special formatting for headings in IEC61850-7-3.
Constant value: **`validation.iec61850.packagesCdc`**

KEY_VALIDATION_IEC61850_PACKAGES_DA

`public static final java.lang.String` **`KEY_VALIDATION_IEC61850_PACKAGES_DA`**

Comma-separated list of package names whose all deep-child elements (both sub-packages and classes) should contain a human-readable name and may need special table formatting in IEC61850-7-3.
Constant value: **`validation.iec61850.packagesDa`**

KEY_VALIDATION_IEC61850_PACKAGES_BASIC

`public static final java.lang.String` **`KEY_VALIDATION_IEC61850_PACKAGES_BASIC`**

Comma-separated list of package names whose all deep-child elements (both sub-packages and classes) should contain a human-readable name and may need special table formatting in IEC61850-7-2.
Constant value: **`validation.iec61850.packagesBasic`**

KEY_VALIDATION_IEC61850_PACKAGE_PRES_COND

`public static final java.lang.String` **`KEY_VALIDATION_IEC61850_PACKAGE_PRES_COND`**

Name of the UML package where the presence conditions of IEC61850 are defined.
Constant value: **`validation.iec61850.packagePresCond`**

KEY_VALIDATION_IEC61850_PACKAGE_FC

`public static final java.lang.String` **`KEY_VALIDATION_IEC61850_PACKAGE_FC`**

Name of the UML package where the functional constraints of IEC61850 are defined.
Constant value: **`validation.iec61850.packageFC`**

KEY_VALIDATION_IEC61850_PACKAGE_TRGOP

`public static final java.lang.String` **`KEY_VALIDATION_IEC61850_PACKAGE_TRGOP`**

Name of the UML package where the trigger options of IEC61850 are defined.
Constant value: **`validation.iec61850.packageTrgOp`**

KEY_VALIDATION_IEC61850_PACKAGES_DO_ABBR

`public static final java.lang.String` **`KEY_VALIDATION_IEC61850_PACKAGES_DO_ABBR`**

Name of the UML package where the abbreviations for data object names in IEC61850 are defined.
Constant value: **`validation.iec61850.packagesDoAbbr`**

KEY_VALIDATION_IEC61850_PACKAGE_LN_MAPS

`public static final java.lang.String` **`KEY_VALIDATION_IEC61850_PACKAGE_LN_MAPS`**

Name of the UML package where the requirements specification for logical nodes (IEC61850-5) is defined.
Constant value: **`validation.iec61850.packageLnMaps`**

KEY_STATISTICS_ON

`public static final java.lang.String` **`KEY_STATISTICS_ON`**

Top-level functionality: Enable statistics = "true" (default = "false", "", null).
Constant value: **`statistics.on`**

KEY_STATISTICS_TAGS_TO_IGNORE

`public static final java.lang.String` **`KEY_STATISTICS_TAGS_TO_IGNORE`**

(if #KEY_STATISTICS_ON="true"): Comma-separated list of tagged values for which to skip logging to console.
Constant value: **`statistics.tagsToIgnore`**

KEY_STATISTICS_CIM_IGNORE_ID_OBJECT_INHERITANCE

`public static final java.lang.String` **`KEY_STATISTICS_CIM_IGNORE_ID_OBJECT_INHERITANCE`**

(if #KEY_STATISTICS_ON="true"): Skip logging to console dependencies through inheritance from CIM IdentifiedObject = "true" (default = "false", "", null).
Constant value: **`statistics.cim.ignoreIdObjectInheritance`**

KEY_STATISTICS_CIM_IGNORE_DOMAIN_CLASS_ATTRIBUTES

`public static final java.lang.String` **`KEY_STATISTICS_CIM_IGNORE_DOMAIN_CLASS_ATTRIBUTES`**

(if #KEY_STATISTICS_ON="true"): Skip logging to console dependencies through usage of types from CIM Domain package in attributes = "true" (default = "false", "", null).
Constant value: **`statistics.cim.ignoreDomainClassAttributes`**

KEY_PROFILES_CROSSCHECK_ON

`public static final java.lang.String` **`KEY_PROFILES_CROSSCHECK_ON`**

Top-level functionality: Enable crosscheck between the UML model and a set of profiles = "true" (default = "false", "", null). Two models are created in-memory: one from UML and one from profiles.
Constant value: **`profiles.crosscheck.on`**

KEY_DOCGEN_ON

`public static final java.lang.String` **`KEY_DOCGEN_ON`**

Top-level functionality: Enable document generation = "true" (default = "false", "", null). UML is source of the model if #KEY_PROFILES_DOCGEN_ON="false"), otherwise the source of the model are profiles.
Constant value: **`docgen.on`**

KEY_DOCGEN_WORD_IN_TEMPLATE

`public static final java.lang.String` **`KEY_DOCGEN_WORD_IN_TEMPLATE`**

File name of the (input) MS Word document template, expected to be found on the classpath. If not specified, default `DEFAULT_WORD_IN_TEMPLATE_FILENAME` is used.
Constant value: **`docgen.word.inTemplate`**

KEY_DOCGEN_WORD_OUT_DOCUMENT

`public static final java.lang.String` **`KEY_DOCGEN_WORD_OUT_DOCUMENT`**

File name of the (output) MS Word generated document; will be created in #OUTPUT_DIR_NAME from template #KEY_DOCGEN_WORD_IN_TEMPLATE and the UML, profile or in-memory model. If not specified, default `DEFAULT_WORD_OUT_DOCUMENT_FILENAME` is used.
Constant value: **`docgen.word.outDocument`**

KEY_DOCGEN_XML_SCOPE

`public static final java.lang.String` **`KEY_DOCGEN_XML_SCOPE`**

Comma-separated list of values corresponding to literals in `OwningWg`; empty value (default) takes them all.
Constant value: **`docgen.xml.scope`**

KEY_DOCGEN_XML_OUT_SPEC

`public static final java.lang.String` **`KEY_DOCGEN_XML_OUT_SPEC`**

File name of the (output) XML generated specification document; will be created in #OUTPUT_DIR_NAME from the UML, profile or in-memory model. If not specified, default `DEFAULT_OUT_XML_SPEC_FILENAME` is used.
Constant value: **`docgen.xml.outSpec`**

KEY_DOCGEN_XML_OUT_DOC

`public static final java.lang.String` **`KEY_DOCGEN_XML_OUT_DOC`**

File name of the (output) XML generated documentation (translatable) document; will be created in #OUTPUT_DIR_NAME from the UML, profile or in-memory model. If not specified, default `DEFAULT_OUT_XML_DOC_FILENAME` is used.
Constant value: **`docgen.xml.outDoc`**

KEY_MIBGEN_ON

`public static final java.lang.String` **`KEY_MIBGEN_ON`**

Top-level functionality: Enable MIBs generation = "true" (default = "false", "", null).
Constant value: **`mibgen.on`**

KEY_MIBGEN_OUT_DIR_FULL

`public static final java.lang.String` **`KEY_MIBGEN_OUT_DIR_FULL`**

(continued from last page)

Directory name for the (output) full MIBs; will be created in #OUTPUT_DIR_NAME from the UML or in-memory model. If not specified, default DEFAULT_MIBS_OUT_DIRNAME is used.
Constant value: **mibgen.outDirFull**

KEY_MIBGEN_OUT_DIR_LIGHT

`public static final java.lang.String` **KEY_MIBGEN_OUT_DIR_LIGHT**

Directory name for the (output) light MIBs; will be created in #OUTPUT_DIR_NAME from the UML or in-memory model. If not specified, default DEFAULT_MIBSLIGHT_OUT_DIRNAME is used.
Constant value: **mibgen.outDirLight**

KEY_DOCGEN_WORD_USE_DOC_FORMAT

`public static final java.lang.String` **KEY_DOCGEN_WORD_USE_DOC_FORMAT**

Force MS Word COM (.doc) document generation = "true" (default = "false", "", null), as opposed to the Open XML (.docx) format.
Constant value: **docgen.word.useDocFormat**

KEY_DOCGEN_WORD_INTRO_TO_FIGURE_BEFORE

`public static final java.lang.String` **KEY_DOCGEN_WORD_INTRO_TO_FIGURE_BEFORE**

Preserves original way of printing introduction to figures first in MS Word document generation = "true" (default = "false", "", null), as opposed to referring to the figure caption below the figure and its caption.
Constant value: **docgen.word.introToFigureBefore**

KEY_DOCGEN_WORD_SAVE_REOPEN_EVERY

`public static final java.lang.String` **KEY_DOCGEN_WORD_SAVE_REOPEN_EVERY**

The number of tables (and table captions) to write before closing and reopening the document.
Constant value: **docgen.word.saveReopenEvery**

KEY_DOCGEN_WORD_ANALYSE_PLACEHOLDERS

`public static final java.lang.String` **KEY_DOCGEN_WORD_ANALYSE_PLACEHOLDERS**

Enable dry run of MS Word document generation = "true" (default = "false", "", null).
Constant value: **docgen.word.analysePlaceholders**

KEY_DOCGEN_WORD_USE_HYPERLINKS

`public static final java.lang.String` **KEY_DOCGEN_WORD_USE_HYPERLINKS**

Enable hyperlinking of UML model elements in the generated MS Word document = "true" (default = "false", "", null).
Constant value: **docgen.word.useHyperlinks**

KEY_DOCGEN_WORD_INCLUDE_INHERITANCE_PATH

`public static final java.lang.String` **KEY_DOCGEN_WORD_INCLUDE_INHERITANCE_PATH**

Include inheritance paths when printing classes in the generated document in the generated MS Word document = "true" (default = "false", "", null).
Constant value: **docgen.word.includeInheritancePath**

(continued from last page)

KEY_DOCGEN_WORD_STYLES_PREFIX_TOC

`public static final java.lang.String` **`KEY_DOCGEN_WORD_STYLES_PREFIX_TOC`**

Comma-separated list of TOC style name prefixes (style name without number), in order of preference. When scanning existing style names, they are matched against this list. If none specified, default `DEFAULT_STYLES_PREFIX_TOC` is used.
Constant value: **`docgen.word.styles.prefix.toc`**

KEY_DOCGEN_WORD_STYLES_PREFIX_HEAD

`public static final java.lang.String` **`KEY_DOCGEN_WORD_STYLES_PREFIX_HEAD`**

Comma-separated list of Heading style name prefixes (style name without number), in order of preference. When scanning existing style names, they are matched against this list. If none specified, default `DEFAULT_STYLES_PREFIX_HEAD` is used.
Constant value: **`docgen.word.styles.prefix.head`**

KEY_DOCGEN_WORD_STYLES_PARA

`public static final java.lang.String` **`KEY_DOCGEN_WORD_STYLES_PARA`**

Comma-separated list of single-word style names for text paragraphs, in order of preference. When scanning existing style names, they are matched against this list. If none specified, default `DEFAULT_STYLES_PARA` is used.
Constant value: **`docgen.word.styles.para`**

KEY_DOCGEN_WORD_STYLES_FIG

`public static final java.lang.String` **`KEY_DOCGEN_WORD_STYLES_FIG`**

Comma-separated list of single-word style names for figures, in order of preference. When scanning existing style names, they are matched against this list. If none specified, default `DEFAULT_STYLES_FIG` is used.
Constant value: **`docgen.word.styles.fig`**

KEY_DOCGEN_WORD_STYLES_TABHEAD

`public static final java.lang.String` **`KEY_DOCGEN_WORD_STYLES_TABHEAD`**

Comma-separated list of single-word style names for table headings, in order of preference. When scanning existing style names, they are matched against this list. If none specified, default `DEFAULT_STYLES_TABHEAD` is used.
Constant value: **`docgen.word.styles.tabhead`**

KEY_DOCGEN_WORD_STYLES_TABCELL

`public static final java.lang.String` **`KEY_DOCGEN_WORD_STYLES_TABCELL`**

Comma-separated list of single-word style names for table cells, in order of preference. When scanning existing style names, they are matched against this list. If none specified, default `DEFAULT_STYLES_TABCELL` is used.
Constant value: **`docgen.word.styles.tabcell`**

KEY_DOCGEN_WORD_STYLES_FIGCAPT

`public static final java.lang.String` **`KEY_DOCGEN_WORD_STYLES_FIGCAPT`**

Comma-separated list of single-word style names for figure caption, in order of preference. When scanning existing style names, they are matched against this list. If none specified, default `DEFAULT_STYLES_TABCAPT` is used.
Constant value: **`docgen.word.styles.figcapt`**

KEY_DOCGEN_WORD_STYLES_TABCAPT

`public static final java.lang.String` **`KEY_DOCGEN_WORD_STYLES_TABCAPT`**

(continued from last page)

Comma-separated list of single-word style names for table caption, in order of preference. When scanning existing style names, they are matched against this list. If none specified, default DEFAULT_STYLES_TABCAPT is used.
Constant value: **docgen.word.styles.tabcapt**

KEY_DOCGEN_INCLUDE_INFORMATIVE

`public static final java.lang.String` **KEY_DOCGEN_INCLUDE_INFORMATIVE**

Include informative UML model elements in the generated document = "true" (default = "false", "", null).
Constant value: **docgen.includeInformative**

KEY_DOCGEN_INCLUDE_NON_PUBLIC

`public static final java.lang.String` **KEY_DOCGEN_INCLUDE_NON_PUBLIC**

Include non-public UML model elements in the generated document = "true" (default = "false", "", null).
Constant value: **docgen.includeNonPublic**

KEY_DOCGEN_PRINT_HTML

`public static final java.lang.String` **KEY_DOCGEN_PRINT_HTML**

Enable formatted documentation of UML model elements in the generated document = "true" (default = "false", "", null).
Constant value: **docgen.printHtml**

KEY_DOCGEN_SHOW_CUSTOM_STEREOTYPES

`public static final java.lang.String` **KEY_DOCGEN_SHOW_CUSTOM_STEREOTYPES**

Enable showing custom UML stereotypes on UML model elements in the generated document = "true" (default = "false", "", null).
Constant value: **docgen.showCustomStereotypes**

KEY_DOCGEN_SKIP_FOR_CUSTOM_STEREOTYPES

`public static final java.lang.String` **KEY_DOCGEN_SKIP_FOR_CUSTOM_STEREOTYPES**

Comma-separated list of custom stereotypes which if used, their elements are to be skipped when generating documents. This list is matched against the built-in, standard stereotypes according to modelling rules and any built-in stereotype from this list is removed (i.e., no overriding of built-in stereotype consideration for document generation). Empty list is valid and reflects default behaviour, i.e., everything found in the model according to other specified filtering options is printed.
Constant value: **docgen.skipForCustomStereotypes**

KEY_DOCGEN_SHOW_NAMESPACE_PACKAGES

`public static final java.lang.String` **KEY_DOCGEN_SHOW_NAMESPACE_PACKAGES**

Comma-separated list of packages for which to explicitly show namespace and URI (if known) in the auto-generated document.
Constant value: **docgen.showNamespacePackages**

KEY_DOCGEN_IEC61850_INCLUDE_METAMODEL_INHERITANCE

`public static final java.lang.String` **KEY_DOCGEN_IEC61850_INCLUDE_METAMODEL_INHERITANCE**

Write inheritance from IEC61850 UML meta-model KEY_VALIDATION_IEC61850_PACKAGE_META_MODEL = "true" (default = "false", "", null).
Constant value: **docgen.iec61850.includeMetamodelInheritance**

(continued from last page)

KEY_DOCGEN_IEC61850_WRITE_UML_TYPES

`public static final java.lang.String` **`KEY_DOCGEN_IEC61850_WRITE_UML_TYPES`**

Write types with their real names, as they are in IEC61850 UML = "true" (default = "false", "", null).
Constant value: **`docgen.iec61850.writeUmlTypes`**

KEY_PROFILES_DOCGEN_ON

`public static final java.lang.String` **`KEY_PROFILES_DOCGEN_ON`**

Top-level functionality: Enable document generation for profiles = "true" (default = "false", "", null), if also #KEY_DOCGEN_ON="true".
Constant value: **`profiles.docgen.on`**

INPUT_DIR_NAME

`public static final java.lang.String` **`INPUT_DIR_NAME`**

Default input directory, set to be on the classpath.
Constant value: **`input`**

PROFILES_DIR_NAME

`public static final java.lang.String` **`PROFILES_DIR_NAME`**

Directory under classpath below which to search for profile files.
Constant value: **`profiles`**

DEFAULT_PROFILES_RELPATH

`public static final java.lang.String` **`DEFAULT_PROFILES_RELPATH`**

Default value for

OUTPUT_DIR_NAME

`public static final java.lang.String` **`OUTPUT_DIR_NAME`**

Directory into which to create output files.
Constant value: **`output`**

PICS_DIR_NAME

`public static final java.lang.String` **`PICS_DIR_NAME`**

Directory name for exported diagram images.
Constant value: **`pics`**

MODEL_PICS_RELPATH

`public static final java.lang.String` **`MODEL_PICS_RELPATH`**

Directory path for exported diagram images.

DEFAULT_PROPS_FILE_NAME

`public static final java.lang.String` **`DEFAULT_PROPS_FILE_NAME`**

(continued from last page)

Application configuration properties, to be edited by the user.
Constant value: **config.properties**

IEC61850_PROPS_FILE_NAME

`public static final java.lang.String` **IEC61850_PROPS_FILE_NAME**

IEC 61850-specific application configuration properties, to be edited by the user.
Constant value: **config61850.properties**

VERSION_PROPS_FILENAME

`public static final java.lang.String` **VERSION_PROPS_FILENAME**

Build properties (for use by ant script), containing also the version information.
Constant value: **build.properties**

DEFAULT_WEBACCESS_SCHEMA_FILENAME

`public static final java.lang.String` **DEFAULT_WEBACCESS_SCHEMA_FILENAME**

Searched on classpath.
Constant value: **IECDomain.xsd**

DEFAULT_VERSION_PROP_NAME

`public static final java.lang.String` **DEFAULT_VERSION_PROP_NAME**

Project version property name (for use by ant script), and if not available through the jar manifest, available through .
Constant value: **project.version**

DEFAULT_WORD_IN_TEMPLATE_FILENAME

`public static final java.lang.String` **DEFAULT_WORD_IN_TEMPLATE_FILENAME**

Default file name for MS Word input template.
Constant value: **base-small-template.docx**

DEFAULT_WORD_OUT_DOCUMENT_FILENAME

`public static final java.lang.String` **DEFAULT_WORD_OUT_DOCUMENT_FILENAME**

Default file name for MS Word output document (generated from input template and a model).
Constant value: **base-small.docx**

DEFAULT_OUT_XML_SPEC_FILENAME

`public static final java.lang.String` **DEFAULT_OUT_XML_SPEC_FILENAME**

Default file name for XML spec output document (generated from a model).
Constant value: **base-small-spec.xml**

DEFAULT_OUT_XML_DOC_FILENAME

`public static final java.lang.String` **DEFAULT_OUT_XML_DOC_FILENAME**

Default file name for XML doc output document (generated from a model).
Constant value: **base-small-doc.xml**

(continued from last page)

DEFAULT_MIBS_OUT_DIRNAME

`public static final java.lang.String` **`DEFAULT_MIBS_OUT_DIRNAME`**

Default directory name for full MIBs.
Constant value: **`mibs`**

DEFAULT_MIBSLIGHT_OUT_DIRNAME

`public static final java.lang.String` **`DEFAULT_MIBSLIGHT_OUT_DIRNAME`**

Default directory name for light MIBs.
Constant value: **`mibslight`**

DEFAULT_BLANK_PNG_FILENAME

`public static final java.lang.String` **`DEFAULT_BLANK_PNG_FILENAME`**

Empty image, used when not storing diagrams into files.
Constant value: **`blank.png`**

DEFAULT_STYLES_PREFIX_TOC

`public static final java.lang.String` **`DEFAULT_STYLES_PREFIX_TOC`**

Default prefix (style name without number) for TOC styles, if none provided.
Constant value: **`TOC`**

DEFAULT_STYLES_PREFIX_HEAD

`public static final java.lang.String` **`DEFAULT_STYLES_PREFIX_HEAD`**

Default prefix (style name without number) for heading styles, if none provided.
Constant value: **`Heading`**

DEFAULT_STYLES_PARA

`public static final java.lang.String` **`DEFAULT_STYLES_PARA`**

Default style name for text paragraph, if none provided.
Constant value: **`PARAGRAPH`**

DEFAULT_STYLES_FIG

`public static final java.lang.String` **`DEFAULT_STYLES_FIG`**

Default style name for figure, if none provided.
Constant value: **`FIGURE`**

DEFAULT_STYLES_TABHEAD

`public static final java.lang.String` **`DEFAULT_STYLES_TABHEAD`**

Default style name for table heading, if none provided.
Constant value: **`TABLE-col-heading`**

DEFAULT_STYLES_TABCELL

`public static final java.lang.String` **`DEFAULT_STYLES_TABCELL`**

(continued from last page)

Default style name for table cell, if none provided.
Constant value: **`TABLE-cell`**

DEFAULT_STYLES_FIGCAPT

`public static final java.lang.String` **`DEFAULT_STYLES_FIGCAPT`**

Default style name for figure caption, if none provided.
Constant value: **`FIGURE-title`**

DEFAULT_STYLES_TABCAPT

`public static final java.lang.String` **`DEFAULT_STYLES_TABCAPT`**

Default style name for table caption, if none provided.
Constant value: **`TABLE-title`**

XSD_EXT

`public static final java.lang.String` **`XSD_EXT`**

Supported profile format (file extension).
Constant value: **`xsd`**

Constructors

Config

```
public Config(java.lang.String propsFilename,
              java.lang.String modelFilename)
```

Constructor.

Parameters:

`propsFilename` - non-empty name of properties file; if null, default will be used. If no such a file can be found on the classpath, empty properties set is created.
`modelFilename` - non-empty name of model file, that will override the value in property #KEY_MODEL_FILENAME; if null, the value in property #KEY_MODEL_FILENAME is used; if that one is also null, no model file will be read.

Throws:

`ApplicationException`

Config

```
public Config(java.util.Properties props,
              java.lang.String modelFilename)
```

Constructor; useful for testing.

Parameters:

`props` - properties initialised from code instead of from file.
`modelFilename` - non-empty name of model file, that will override the value in property #KEY_MODEL_FILENAME; if null, the value in property #KEY_MODEL_FILENAME is used; if that one is also null, no model file will be read.

Throws:

`ApplicationException`

Methods

(continued from last page)

deduceAppVersion

`public static java.lang.String deduceAppVersion()`

getAppVersion

`public java.lang.String getAppVersion()`

Returns application version deduced from manifest (if running with a jar), or read from `VERSION_PROPS_FILENAME` file otherwise.

isAppSkipTiming

`public boolean isAppSkipTiming()`

Returns whether to skip logging ellapsed time.

getModelBuilder

`public ModelBuilderKind getModelBuilder()`

Returns whether to use SQL to build model.

getModelFileAbsPath

`public java.lang.String getModelFileAbsPath()`

Returns absolute path of the UML model file.

hasUmlModel

`public boolean hasUmlModel()`

Returns true when EA file is specified in configuration.

getIec61850NaturePackages

`public java.util.List getIec61850NaturePackages()`

Comma-separated list of names of model packages (below the root) with non-CIM nature.

getBlankPngFileAbsPath

`public java.lang.String getBlankPngFileAbsPath()`

Returns absolute path of the default image file, to be used as replacement when the "real" image is not available.

isXmiexportOn

`public boolean isXmiexportOn()`

Returns whether export to XMI is enabled; applicable only if the source of the model is an .eap file.

(continued from last page)

getXmiexportDialects

`public java.util.EnumSet getXmiexportDialects()`

Returns the configured XMI dialects to be used for export.

isValidationOn

`public boolean isValidationOn()`

getValidationScope

`public java.util.EnumSet getValidationScope()`

Returns the owners of packages that determine the scope of validation and statistics. Note that despite these options, the full model needs to be built.

isValidationPackagesOn

`public boolean isValidationPackagesOn()`

isValidationClassesOn

`public boolean isValidationClassesOn()`

isValidationAssociationsOn

`public boolean isValidationAssociationsOn()`

isValidationAttributesOn

`public boolean isValidationAttributesOn()`

isValidationOperationsOn

`public boolean isValidationOperationsOn()`

isValidationDependenciesOn

`public boolean isValidationDependenciesOn()`

isValidationDiagramsOn

`public boolean isValidationDiagramsOn()`

(continued from last page)

getValidationRulesOff

```
public java.util.Collection getValidationRulesOff()
```

isValidationLoggingVerbose

```
public boolean isValidationLoggingVerbose()
```

getValidationPackagesDataIndex

```
public java.util.Collection getValidationPackagesDataIndex()
```

getValidationIec61850Packages72

```
public java.util.Collection getValidationIec61850Packages72()
```

getValidationIec61850Packages73

```
public java.util.Collection getValidationIec61850Packages73()
```

getValidationIec61850Packages74

```
public java.util.Collection getValidationIec61850Packages74()
```

getValidationIec61850PackageMetaModel

```
public java.lang.String getValidationIec61850PackageMetaModel()
```

getValidationIec61850Package72Top

```
public java.lang.String getValidationIec61850Package72Top()
```

getValidationIec61850PackagesEnumsXml

```
public java.util.Collection getValidationIec61850PackagesEnumsXml()
```

(continued from last page)

getValidationIec61850PackagesLn

`public java.util.Collection` **`getValidationIec61850PackagesLn`**`()`

getValidationIec61850PackagesCdc

`public java.util.Collection` **`getValidationIec61850PackagesCdc`**`()`

getValidationIec61850PackagesDa

`public java.util.Collection` **`getValidationIec61850PackagesDa`**`()`

getValidationIec61850PackagesBasic

`public java.util.Collection` **`getValidationIec61850PackagesBasic`**`()`

getValidationIec61850PackagesExtTitle

`public java.util.Collection` **`getValidationIec61850PackagesExtTitle`**`()`

Returns the union of `getValidationIec61850PackagesLn()`, `getValidationIec61850PackagesCdc()` and .

getValidationIec61850PackagesDocgen

`public java.util.Collection` **`getValidationIec61850PackagesDocgen`**`()`

Returns the union of `getValidationIec61850Packages72()`, `getValidationIec61850Packages73()` and `getValidationIec61850Packages74()`.

getValidationIec61850PackagePresCond

`public java.lang.String` **`getValidationIec61850PackagePresCond`**`()`

getValidationIec61850PackageFc

`public java.lang.String` **`getValidationIec61850PackageFc`**`()`

getValidationIec61850PackageTrgOp

`public java.lang.String` **`getValidationIec61850PackageTrgOp`**`()`

getValidationIec61850PackagesDoAbbr

`public java.util.Collection` **`getValidationIec61850PackagesDoAbbr`**`()`

(continued from last page)

getValidationIec61850PackageLnMaps

`public java.lang.String` **`getValidationIec61850PackageLnMaps`**`()`

isStatisticsOn

`public boolean` **`isStatisticsOn`**`()`

getStatisticsTagsToIgnore

`public java.util.Collection` **`getStatisticsTagsToIgnore`**`()`

isStatisticsCimIgnoreIdObjectInheritance

`public boolean` **`isStatisticsCimIgnoreIdObjectInheritance`**`()`

isStatisticsCimIgnoreDomainClassAttributes

`public boolean` **`isStatisticsCimIgnoreDomainClassAttributes`**`()`

isProfilesCrosscheckOn

`public boolean` **`isProfilesCrosscheckOn`**`()`

isMibgenOn

`public boolean` **`isMibgenOn`**`()`

getMibgenOutDirFullAbsPath

`public java.lang.String` **`getMibgenOutDirFullAbsPath`**`()`

Returns absolute path of output for full MIBs, null if doc generation disabled or output files are given in another format.

getMibgenOutDirLightAbsPath

`public java.lang.String` **`getMibgenOutDirLightAbsPath`**`()`

Returns absolute path of output for light MIBs, null if doc generation disabled or output files are given in another format.

(continued from last page)

isDocgenOn

`public boolean isDocgenOn()`

isDocgenWordUseDocFormat

`public boolean isDocgenWordUseDocFormat()`

isDocgenWordIntroToFigureBefore

`public boolean isDocgenWordIntroToFigureBefore()`

getDocgenWordSaveReopenEvery

`public int getDocgenWordSaveReopenEvery()`

In case an integer cannot be parsed, returns -1; otherwise, an absolute value of #KEY_DOCGEN_WORD_SAVE_REOPEN_EVERY.

isDocgenWordAnalysePlaceholders

`public boolean isDocgenWordAnalysePlaceholders()`

Returns whether to only analyse placeholders in output MS Word document, without replacing them with the full content.

isDocgenWordUseHyperlinks

`public boolean isDocgenWordUseHyperlinks()`

Returns whether to use hyperlinks in output MS Word document.

isDocgenWordIncludeInheritancePath

`public boolean isDocgenWordIncludeInheritancePath()`

Returns whether to include inheritance path in output MS Word document.

getDocgenWordStylesPrefixToc

`public java.util.List getDocgenWordStylesPrefixToc()`

Returns non-empty list of user-prioritised prefixes for TOC style names.

getDocgenWordStylesPrefixHead

`public java.util.List getDocgenWordStylesPrefixHead()`

Returns non-empty list of user-prioritised prefixes for heading style names.

getDocgenWordStylesPara

`public java.util.List getDocgenWordStylesPara()`

(continued from last page)

Returns non-empty list of user-prioritised names for paragraph text styles.

getDocgenWordStylesFig

`public java.util.List` **`getDocgenWordStylesFig`**`()`

Returns non-empty list of user-prioritised names for figure styles.

getDocgenWordStylesTabhead

`public java.util.List` **`getDocgenWordStylesTabhead`**`()`

Returns non-empty list of user-prioritised names for table heading styles.

getDocgenWordStylesTabcell

`public java.util.List` **`getDocgenWordStylesTabcell`**`()`

Returns non-empty list of user-prioritised names for table cells styles.

getDocgenWordStylesFigcapt

`public java.util.List` **`getDocgenWordStylesFigcapt`**`()`

Returns non-empty list of user-prioritised names for figure caption styles.

getDocgenWordStylesTabcapt

`public java.util.List` **`getDocgenWordStylesTabcapt`**`()`

Returns non-empty list of user-prioritised names for table caption styles.

getDocgenWordInTemplateFileAbsPath

`public java.lang.String` **`getDocgenWordInTemplateFileAbsPath`**`()`

Returns absolute path of input Word template file, null if doc generation disabled or output files are given in another format.

getDocgenWordOutDocumentFileAbsPath

`public java.lang.String` **`getDocgenWordOutDocumentFileAbsPath`**`()`

Returns absolute path of output Word document file, null if doc generation disabled or output files are given in another format.

getDocgenXsdInWebaccessFileAbsPath

`public java.lang.String` **`getDocgenXsdInWebaccessFileAbsPath`**`()`

Returns absolute path of XML web access schema file, null if doc generation disabled or output files are given in another format.

getDocgenXmlOutSpecFileAbsPath

`public java.lang.String` **`getDocgenXmlOutSpecFileAbsPath`**`()`

Returns absolute path of output XML spec file, null if doc generation disabled or output files are given in another format.

getDocgenXmlOutDocFileAbsPath

`public java.lang.String` **`getDocgenXmlOutDocFileAbsPath`**`()`

Returns absolute path of output XML doc file, null if doc generation disabled or output files are given in another format.

getDocgenXsdOutWebaccessFileAbsPath

`public java.lang.String` **`getDocgenXsdOutWebaccessFileAbsPath`**`()`

Returns absolute path of where to copy the XML web access schema, null if doc generation disabled or output files are given in another format.

isRemovePicsAfterExit

`public boolean` **`isRemovePicsAfterExit`**`()`

Returns whether to remove exported diagrams at application exit (true if MS Word output is to be generated).

getDocgenXmlScope

`public java.util.EnumSet` **`getDocgenXmlScope`**`()`

Returns the owners of packages that determine the scope of XML generation. Note that despite these options, the full model needs to be built.

isDocgenIncludeInformative

`public boolean` **`isDocgenIncludeInformative`**`()`

isDocgenIncludeNonPublic

`public boolean` **`isDocgenIncludeNonPublic`**`()`

isDocgenPrintHtml

`public boolean` **`isDocgenPrintHtml`**`()`

Returns whether to respect markup (present in UML descriptions) in output document.

isDocgenShowCustomStereotypes

`public boolean` **`isDocgenShowCustomStereotypes`**`()`

getDocgenSkipForCustomStereotypes

`public java.util.List` **`getDocgenSkipForCustomStereotypes`**`()`

(continued from last page)

getDocgenShowNamespacePackages

`public java.util.List` **`getDocgenShowNamespacePackages`**`()`

isDocgenIec61850IncludeMetamodelInheritance

`public boolean` **`isDocgenIec61850IncludeMetamodelInheritance`**`()`

isDocgenIec61850WriteUmlTypes

`public boolean` **`isDocgenIec61850WriteUmlTypes`**`()`

isProfilesDocgenOn

`public boolean` **`isProfilesDocgenOn`**`()`

Returns true if both general and profile docgen are enabled.

isDocgenModelOn

`public boolean` **`isDocgenModelOn`**`()`

Returns true if only docgen from EA is enabled (but not profile docgen).

getPicsDirAbsPath

`public java.lang.String` **`getPicsDirAbsPath`**`()`

Returns absolute path string for directory where to export images from the model.

getProfilesRelpath

`public java.lang.String` **`getProfilesRelpath`**`()`

getProfileFiles

`public java.util.Map` **`getProfileFiles`**`()`

toString

`public java.lang.String` **`toString`**`()`

org.tanjakostic.jcleancim.common
Class ModelBuilderKind

```
java.lang.Object
  |-java.lang.Enum
    |-org.tanjakostic.jcleancim.common.ModelBuilderKind
```

All Implemented Interfaces:
java.io.Serializable, java.lang.Comparable

public final class **ModelBuilderKind**
extends java.lang.Enum

Kind for model builder.

Field Summary

public static final	db Reads an Access DB, jet3.5 format as read-only.
public static final	japi Original implementation (extremely slow EA API).
public static final	sqlxml Uses EA Java API to open repository and do bulk queries instead of iterations.

Method Summary

java.lang.String	getText()
static ModelBuilderKind	valueOf(java.lang.String name)
static ModelBuilderKind[]	values()

Methods inherited from class `java.lang.Enum`

clone, compareTo, equals, finalize, getDeclaringClass, hashCode, name, ordinal, toString, valueOf

Methods inherited from class `java.lang.Object`

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface `java.lang.Comparable`

compareTo

Fields

(continued from last page)

db

`public static final org.tanjakostic.jcleancim.common.ModelBuilderKind **db**`

Reads an Access DB, jet3.5 format as read-only.

sqlxml

`public static final org.tanjakostic.jcleancim.common.ModelBuilderKind **sqlxml**`

Uses EA Java API to open repository and do bulk queries instead of iterations.

japi

`public static final org.tanjakostic.jcleancim.common.ModelBuilderKind **japi**`

Original implementation (extremely slow EA API).

Methods

values

`public static ModelBuilderKind[] **values**()`

valueOf

`public static ModelBuilderKind **valueOf**(java.lang.String name)`

getText

`public java.lang.String **getText**()`

org.tanjakostic.jcleancim.common
Class Nature

```
java.lang.Object
  +-java.lang.Enum
    +-org.tanjakostic.jcleancim.common.Nature
```

All Implemented Interfaces:
java.io.Serializable, java.lang.Comparable

---

public final class **Nature**
extends java.lang.Enum

Nature of the UML object, determining the modelling and validation rules to apply.

Implementation note: We keep this class in the common package instead of model package because OwningWg depends on it.

Field Summary

public static final	CIM Canonical CIM domain.
public static final	IEC61850 Pure IEC 61850 domain.

Method Summary

static boolean	isAnyCim(Nature nature)
static boolean	isIec61850(Nature nature)
static Nature	valueOf(java.lang.String name)
static Nature[]	values()

Methods inherited from class `java.lang.Enum`

clone, compareTo, equals, finalize, getDeclaringClass, hashCode, name, ordinal, toString, valueOf

Methods inherited from class `java.lang.Object`

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface `java.lang.Comparable`

compareTo

(continued from last page)

Fields

CIM

`public static final org.tanjakostic.jcleancim.common.Nature CIM`

Canonical CIM domain.

IEC61850

`public static final org.tanjakostic.jcleancim.common.Nature IEC61850`

Pure IEC 61850 domain.

Methods

values

`public static Nature[] values()`

valueOf

`public static Nature valueOf(java.lang.String name)`

isAnyCim

`public static boolean isAnyCim(Nature nature)`

isIec61850

`public static boolean isIec61850(Nature nature)`

org.tanjakostic.jcleancim.common
Class OwningWg

```
java.lang.Object
  +-java.lang.Enum
    +-org.tanjakostic.jcleancim.common.OwningWg
```

All Implemented Interfaces:
java.io.Serializable, java.lang.Comparable

---

public final class **OwningWg**
extends java.lang.Enum

IEC TC57 working group owning the top level package in the combined CIM/IEC61850 model.

We assume that the current UML model has three standard model packages (one that contains canonical CIM top-level packages, one with CIM profiles, and one that contains IEC61850 top-level packages) and any number of custom model packages (both CIM and non-CIM extensions).

TODO: We may want to replace this enum with a class to allow for more flexibility for non-standard packages that could be specified in the org.tanjakostic.jcleancim.common.Config#DEFAULT_PROPS_FILE_NAME file.

Implementation note: We keep this class in the common package instead of model package because `Config` depends on it.

Field Summary

public static final	JWG25
public static final	OTHER_CIM
public static final	OTHER_IEC61850
public static final	WG10
public static final	WG13
public static final	WG14
public static final	WG16
public static final	WG17
public static final	WG18
public static final	WG19

Method Summary

static OwningWg	determineAssociationOwner(OwningWg oneEndOwner, OwningWg otherEndOwner) Returns the owner of an association if both ends have been initialised, null otherwise.
java.util.EnumSet	getAllowedOtherEndOwners() Returns the allowed dependencies of this owner, as per IEC TC57 top-level package dependencies rules.
java.lang.String	getAppDomain()
Nature	getNature()
static OwningWg	getOwnerForTopPackage(java.lang.String topPackageName) Utility method: returns the owner for the given name of top package, null if there is no such a name assigned to an owner.
static java.util.Collection	getReservedTopPackageNames() Returns reserved (standard) top package names.
java.lang.String	getTopPackageName()
boolean	involvedIn(OwningWg oneEnd, OwningWg otherEnd) Returns whether one of the arguments has this owner.
static OwningWg	valueOf(java.lang.String name)
static OwningWg[]	values()

Methods inherited from class `java.lang.Enum`
`clone, compareTo, equals, finalize, getDeclaringClass, hashCode, name, ordinal, toString, valueOf`

Methods inherited from class `java.lang.Object`
`clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait`

Methods inherited from interface `java.lang.Comparable`
`compareTo`

Fields

WG13

`public static final org.tanjakostic.jcleancim.common.OwningWg` **`WG13`**

WG14

`public static final org.tanjakostic.jcleancim.common.OwningWg` **`WG14`**

WG16

```
public static final org.tanjakostic.jcleancim.common.OwningWg WG16
```

OTHER_CIM

```
public static final org.tanjakostic.jcleancim.common.OwningWg OTHER_CIM
```

WG10

```
public static final org.tanjakostic.jcleancim.common.OwningWg WG10
```

WG17

```
public static final org.tanjakostic.jcleancim.common.OwningWg WG17
```

WG18

```
public static final org.tanjakostic.jcleancim.common.OwningWg WG18
```

JWG25

```
public static final org.tanjakostic.jcleancim.common.OwningWg JWG25
```

WG19

```
public static final org.tanjakostic.jcleancim.common.OwningWg WG19
```

OTHER_IEC61850

```
public static final org.tanjakostic.jcleancim.common.OwningWg OTHER_IEC61850
```

Methods

values

```
public static OwningWg[] values()
```

(continued from last page)

valueOf

```
public static OwningWg valueOf(java.lang.String name)
```

getTopPackageName

```
public java.lang.String getTopPackageName()
```

getNature

```
public Nature getNature()
```

getAppDomain

```
public java.lang.String getAppDomain()
```

getReservedTopPackageNames

```
public static java.util.Collection getReservedTopPackageNames()
```

Returns reserved (standard) top package names.

getOwnerForTopPackage

```
public static OwningWg getOwnerForTopPackage(java.lang.String topPackageName)
```

Utility method: returns the owner for the given name of top package, null if there is no such a name assigned to an owner. This latter is the case of extensions.

getAllowedOtherEndOwners

```
public java.util.EnumSet getAllowedOtherEndOwners()
```

Returns the allowed dependencies of this owner, as per IEC TC57 top-level package dependencies rules.

determineAssociationOwner

```
public static OwningWg determineAssociationOwner(OwningWg oneEndOwner,
        OwningWg otherEndOwner)
```

(continued from last page)

Returns the owner of an association if both ends have been initialised, null otherwise.

UML generalisation (inheritance) and UML dependency (hand-drawn in the model, among elements) are relationships that have natural dependency direction, i.e., explicitly from source end to target end. However, for associations in CIM, we unfortunately cannot rely on natural dependencies, because associations are agreed to be bi-directional (we always have association end names on both sides of an association). So, in this method, it does not matter what is source and what is target - `oneEndOwner` and `otherEndOwner`. Owner is assigned for the fact that there is an association between two classes, potentially from different top-level packages (i.e., with potentially different owner).

The owner returned is in the reverse direction of IEC TC57 agreed dependencies. For instance, for an association involving classes between `WG13` and `WG14`, owner is `WG14`, because the class of `WG14` depends on the class of `WG13`, and this latter need not know about who links to it. Therefore, we calculate the actual owner according to the IEC TC57 top-level package dependencies reverse order, as follows:

`WG13 -> WG14`

`WG13 -> WG14 -> WG16`

`WG13 -> WG14 -> WG16 -> OTHER_CIM`

`WG10 -> WG17`

`WG10 -> WG18`

`WG10 -> JWG25`

`[WG13 -> WG14 | WG10 -> WG17 | WG10 -> WG18 | WG10 -> JWG25] -> WG19`

`any -> OTHER_IEC61850`.

Model validators have the job of detecting inconsistencies in dependencies by using `getAllowedOtherEndOwners()`.

Parameters:

`oneEndOwner` - owner of one end of the association.
`otherEndOwner` - owner of the other end of the association.

Returns:

calculated owner of the association.

involvedIn

```
public boolean involvedIn(OwningWg oneEnd,
          OwningWg otherEnd)
```

Returns whether one of the arguments has this owner.

org.tanjakostic.jcleancim.common
Class XMIDialect

```
java.lang.Object
  +-java.lang.Enum
    +-org.tanjakostic.jcleancim.common.XMIDialect
```

All Implemented Interfaces:
java.io.Serializable, java.lang.Comparable

public final class **XMIDialect**
extends java.lang.Enum

Supported XMI dialects for exporting.

Field Summary

public static final	cimtool
public static final	ea_xmi11
public static final	ea_xmi21

Method Summary

java.lang.String	getAsSuffix() Returns the suffix, together with .xmi extension, as used for file name.
java.lang.String	getName()
static XMIDialect	valueOf(java.lang.String name)
static XMIDialect[]	values()

Methods inherited from class `java.lang.Enum`

clone, compareTo, equals, finalize, getDeclaringClass, hashCode, name, ordinal, toString, valueOf

Methods inherited from class `java.lang.Object`

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface `java.lang.Comparable`

compareTo

(continued from last page)

Fields

ea_xmi11

```
public static final org.tanjakostic.jcleancim.common.XMIDialect ea_xmi11
```

ea_xmi21

```
public static final org.tanjakostic.jcleancim.common.XMIDialect ea_xmi21
```

cimtool

```
public static final org.tanjakostic.jcleancim.common.XMIDialect cimtool
```

Methods

values

```
public static XMIDialect[] values()
```

valueOf

```
public static XMIDialect valueOf(java.lang.String name)
```

getName

```
public java.lang.String getName()
```

getAsSuffix

```
public java.lang.String getAsSuffix()
```

Returns the suffix, together with .xmi extension, as used for file name.

Package

org.tanjakostic.jcleancim.docgen

org.tanjakostic.jcleancim.docgen
Class UnsupportedInputFormatException

```
java.lang.Object
  |-java.lang.Throwable
    |-java.lang.Exception
      |-org.tanjakostic.jcleancim.util.ApplicationException
        |-org.tanjakostic.jcleancim.docgen.UnsupportedInputFormatException
```

All Implemented Interfaces:
java.io.Serializable

---

public class **UnsupportedInputFormatException**
extends ApplicationException

Used when the format for input document is a non-supported one.

Constructor Summary

public	UnsupportedInputFormatException()
public	UnsupportedInputFormatException(java.lang.String message, java.lang.Throwable cause)
public	UnsupportedInputFormatException(java.lang.String message)
public	UnsupportedInputFormatException(java.lang.Throwable cause)

Methods inherited from class `java.lang.Throwable`
addSuppressed, fillInStackTrace, getCause, getLocalizedMessage, getMessage, getStackTrace, getSuppressed, initCause, printStackTrace, printStackTrace, printStackTrace, setStackTrace, toString

Methods inherited from class `java.lang.Object`
clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

---

Constructors

UnsupportedInputFormatException

public **UnsupportedInputFormatException**()

(continued from last page)

UnsupportedInputFormatException

```
public UnsupportedInputFormatException(java.lang.String message,
                                       java.lang.Throwable cause)
```

UnsupportedInputFormatException

```
public UnsupportedInputFormatException(java.lang.String message)
```

UnsupportedInputFormatException

```
public UnsupportedInputFormatException(java.lang.Throwable cause)
```

org.tanjakostic.jcleancim.docgen
Class UnsupportedOutputFormatException

```
java.lang.Object
  |-java.lang.Throwable
    |-java.lang.Exception
      |-org.tanjakostic.jcleancim.util.ApplicationException
        |-org.tanjakostic.jcleancim.docgen.UnsupportedOutputFormatException
```

All Implemented Interfaces:
java.io.Serializable

---

public class **UnsupportedOutputFormatException**
extends ApplicationException

Used when the format for output document is a non-supported one.

Constructor Summary

public	UnsupportedOutputFormatException()
public	UnsupportedOutputFormatException(java.lang.String message, java.lang.Throwable cause)
public	UnsupportedOutputFormatException(java.lang.String message)
public	UnsupportedOutputFormatException(java.lang.Throwable cause)

Methods inherited from class `java.lang.Throwable`
addSuppressed, fillInStackTrace, getCause, getLocalizedMessage, getMessage, getStackTrace, getSuppressed, initCause, printStackTrace, printStackTrace, printStackTrace, setStackTrace, toString

Methods inherited from class `java.lang.Object`
clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

---

Constructors

UnsupportedOutputFormatException

public **UnsupportedOutputFormatException**()

(continued from last page)

UnsupportedOutputFormatException

```
public UnsupportedOutputFormatException(java.lang.String message,
                                        java.lang.Throwable cause)
```

UnsupportedOutputFormatException

```
public UnsupportedOutputFormatException(java.lang.String message)
```

UnsupportedOutputFormatException

```
public UnsupportedOutputFormatException(java.lang.Throwable cause)
```

org.tanjakostic.jcleancim.docgen
Class WriterFactory

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.docgen.WriterFactory
```

public class **WriterFactory**
extends java.lang.Object

Factory class, allowing us to specify creation of concrete writers in one place and avoid undesired dependencies.

Method Summary

static Writer	createWriter(Config cfg, DocCollector collector) Creates writer initialised with input data.

Methods inherited from class `java.lang.Object`

`clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait`

Methods

createWriter

```
public static Writer createWriter(Config cfg,
          DocCollector collector)
  throws UnsupportedInputFormatException,
          UnsupportedOutputFormatException,
          java.io.IOException
```

Creates writer initialised with `input` data.

Throws:

`UnsupportedInputFormatException` - if the requested format (extension) of the input file(s) is not supported.
`UnsupportedOutputFormatException` - if the requested format (extension) of the output file(s) is not supported.
`IOException` - on any file system-related problem.

Package

org.tanjakostic.jcleancim.docgen.collector

This package defines interfaces and classes responsible for collecting documentation and figures from the UML model, or freely initialised through the API without any UML model. The result is the UML model content relevant for writing documentation (independent of the UML model) that is then passed to a `Writer`, to actually output some documentation.

Major interfaces and classes are:

- `DocCollector` - the collector of documentation.
- `DocgenConfig` - configuration options specific to document generation (such as what to include/exclude), whether to retain HTML documentation and such.
- `DocgenConfig` - configuration options specific to document generation (such as what to include/exclude), whether to retain HTML documentation and such.
- `FreeFormDocumentation` - collected documentation for free format printing, such as when using a template and placeholders, where you can freely choose content to generate).
- `FixedFormDocumentation` - collected documentation for fixed format printing, such as when printing relevant content for a name space.
- Interfaces are those that end with "Doc" or "Scl". Other are classes responsible for formatting and configuration data.
- `ModelFinder` - interface defining thin set of methods to do lookup into the model as required for document generation (allowed us to do document generation tests without actually having the full model loaded and built from EA file).

The implementation of interfaces are all available in the `impl` sub-package.

org.tanjakostic.jcleancim.docgen.collector
Class AGSpec

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.docgen.collector.AGSpec
```

All Implemented Interfaces:
RawData

public class **AGSpec**
extends java.lang.Object
implements RawData

Specific to IEC 61850 tables for logical nodes and common data classes, this simple data structure holds definition for a valid group.

Field Summary

public static final	DA_CATEGORY Element name when used for instance DAs. Value: **DACategory**
public static final	DA_CTL_MIRROR
public static final	DA_DESCRIPTION
public static final	DA_MEAS
public static final	DA_SDO
public static final	DA_SETTING
public static final	DA_SPAR
public static final	DA_STATUS
public static final	DA_SUBSTITUTION
public static final	DA_TRACKING
public static final	DA_UNDEFINED
public static final	DEFVAL_TAG Default value for element name when used for pretty strings. Value: **PS**
public static final	DO_CATEGORY Element name when used for instance DOs. Value: **DOCategory**

public static final	DO_CONTROL
public static final	DO_DESCRIPTION
public static final	DO_MEAS
public static final	DO_SETTING
public static final	DO_STATUS
public static final	DO_TRACKING
public static final	DO_UNDEFINED

Method Summary

java.lang.String	copyCell(RawData src, java.lang.String key)
java.lang.String	copyNonEmptyCell(RawData src, java.lang.String key)
static AGSpec	create(java.lang.String instTag, java.lang.String kindTag, java.lang.String subhead)
static AGSpec	createSpecial(java.lang.String instTag, java.lang.String kindTag, java.lang.String subhead)
java.lang.String	getCell(java.lang.String key)
java.util.Map	getCells()
static java.util.List	getForInstTag(java.lang.String instTag)
java.lang.String	getInstTag() Returns element name used as container for instance data (DO or FCDA category/group).
java.lang.String	getKindTag() Returns kind.
static java.util.Map	getPredefinedAGSpecs() Returns all the predefined table formats: key=instTag (da vs.
java.lang.String	getSubhead() Returns subhead, i.e., value to be printed (translatable string).
java.lang.String	getSubheadId() Returns subhead identification.
java.lang.String	getTag() Returns element name used as container for pretty string.
boolean	hasKey(java.lang.String key)

boolean	isSpecial() Returns whether this group is somehow special.
java.lang.String	putCell(java.lang.String key, java.lang.String value)
java.lang.String	putCellNonEmpty(java.lang.String key, java.lang.String value)

Methods inherited from class java.lang.Object
clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.docgen.collector.RawData
copyCell, copyNonEmptyCell, getCell, getCells, hasKey, putCell, putCellNonEmpty

Fields

DEFVAL_TAG

public static final java.lang.String **DEFVAL_TAG**

Default value for element name when used for pretty strings.
Constant value: **PS**

DA_CATEGORY

public static final java.lang.String **DA_CATEGORY**

Element name when used for instance DAs.
Constant value: **DACategory**

DA_UNDEFINED

public static final org.tanjakostic.jcleancim.docgen.collector.AGSpec **DA_UNDEFINED**

DA_SDO

public static final org.tanjakostic.jcleancim.docgen.collector.AGSpec **DA_SDO**

DA_STATUS

public static final org.tanjakostic.jcleancim.docgen.collector.AGSpec **DA_STATUS**

DA_MEAS

public static final org.tanjakostic.jcleancim.docgen.collector.AGSpec **DA_MEAS**

DA_CTL_MIRROR

public static final org.tanjakostic.jcleancim.docgen.collector.AGSpec **DA_CTL_MIRROR**

DA_SUBSTITUTION

public static final org.tanjakostic.jcleancim.docgen.collector.AGSpec **DA_SUBSTITUTION**

DA_SETTING

public static final org.tanjakostic.jcleancim.docgen.collector.AGSpec **DA_SETTING**

DA_TRACKING

public static final org.tanjakostic.jcleancim.docgen.collector.AGSpec **DA_TRACKING**

DA_DESCRIPTION

public static final org.tanjakostic.jcleancim.docgen.collector.AGSpec **DA_DESCRIPTION**

DA_SPAR

public static final org.tanjakostic.jcleancim.docgen.collector.AGSpec **DA_SPAR**

DO_CATEGORY

public static final java.lang.String **DO_CATEGORY**

Element name when used for instance DOs.
Constant value: **DOCategory**

DO_UNDEFINED

public static final org.tanjakostic.jcleancim.docgen.collector.AGSpec **DO_UNDEFINED**

DO_DESCRIPTION

public static final org.tanjakostic.jcleancim.docgen.collector.AGSpec **DO_DESCRIPTION**

(continued from last page)

DO_STATUS

```
public static final org.tanjakostic.jcleancim.docgen.collector.AGSpec DO_STATUS
```

DO_MEAS

```
public static final org.tanjakostic.jcleancim.docgen.collector.AGSpec DO_MEAS
```

DO_CONTROL

```
public static final org.tanjakostic.jcleancim.docgen.collector.AGSpec DO_CONTROL
```

DO_SETTING

```
public static final org.tanjakostic.jcleancim.docgen.collector.AGSpec DO_SETTING
```

DO_TRACKING

```
public static final org.tanjakostic.jcleancim.docgen.collector.AGSpec DO_TRACKING
```

Methods

create

```
public static AGSpec create(java.lang.String instTag,
         java.lang.String kindTag,
         java.lang.String subhead)
```

createSpecial

```
public static AGSpec createSpecial(java.lang.String instTag,
         java.lang.String kindTag,
         java.lang.String subhead)
```

getPredefinedAGSpecs

```
public static java.util.Map getPredefinedAGSpecs()
```

Returns all the predefined table formats: key=instTag (da vs. do).

getForInstTag

```
public static java.util.List getForInstTag(java.lang.String instTag)
```

getTag

```
public java.lang.String getTag()
```

Returns element name used as container for pretty string.

getKindTag

```
public java.lang.String getKindTag()
```

Returns kind.

getSubhead

```
public java.lang.String getSubhead()
```

Returns subhead, i.e., value to be printed (translatable string).

getSubheadId

```
public java.lang.String getSubheadId()
```

Returns subhead identification.

isSpecial

```
public boolean isSpecial()
```

Returns whether this group is somehow special.

getInstTag

```
public java.lang.String getInstTag()
```

Returns element name used as container for instance data (DO or FCDA category/group).

putCell

```
public final java.lang.String putCell(java.lang.String key,
         java.lang.String value)
```

putCellNonEmpty

```
public final java.lang.String putCellNonEmpty(java.lang.String key,
         java.lang.String value)
```

copyCell

```
public final java.lang.String copyCell(RawData src,
         java.lang.String key)
```

copyNonEmptyCell

```
public final java.lang.String copyNonEmptyCell(RawData src,
         java.lang.String key)
```

hasKey

```
public final boolean hasKey(java.lang.String key)
```

getCells

```
public final java.util.Map getCells()
```

getCell

```
public final java.lang.String getCell(java.lang.String key)
```

org.tanjakostic.jcleancim.docgen.collector
Class BookmarkRegistry

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.docgen.collector.BookmarkRegistry
```

public class **BookmarkRegistry**
extends java.lang.Object

Registry for bookmarks.

Use `getOrCreateBookmarkID(UmlObject)` for UmlObjects that you potentially want to be able to refer to from others. At present, we do only classes (they are types and super-types) and enumeration literals (they are potentially used as initial values).

Use `markAsAvailableInDocument(String)` in `ObjectDoc` subclasses that actually denote places that you would like to reference (by printing documentation for classes and enumeration literals). These are like bookmarks.

Constructor Summary

public	`BookmarkRegistry()` Constructor.

Method Summary

java.lang.String	`findID(UmlObject o)`
java.lang.String	`getOrCreateBookmarkID(UmlObject o)` Returns the existing or the newly created bookmark ID for key `o`.
boolean	`isAvailableInDocument(java.lang.String bookmarkID)` Returns whether the bookmark has been added to the document during writing; if so, it can be used to create hyperlink.
void	`markAsAvailableInDocument(java.lang.String bookmarkID)` Called from writer for ObjectDocs that have bookmark ID when actual bookmark is added to the document; these are then available in the last pass, to insert hyperlinks pointing to those bookmarks.

Methods inherited from class `java.lang.Object`

`clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait`

Constructors

BookmarkRegistry

`public` **`BookmarkRegistry()`**

Constructor.

Methods

(continued from last page)

getOrCreateBookmarkID

`public java.lang.String **getOrCreateBookmarkID**(UmlObject o)`

Returns the existing or the newly created bookmark ID for key `o`. Use this for every object whose documentation you may want to refer to.

findID

`public java.lang.String **findID**(UmlObject o)`

markAsAvailableInDocument

`public void **markAsAvailableInDocument**(java.lang.String bookmarkID)`

Called from writer for ObjectDocs that have bookmark ID when actual bookmark is added to the document; these are then available in the last pass, to insert hyperlinks pointing to those bookmarks. At present, we create bookmarks for classes and enumeration literals only.

isAvailableInDocument

`public boolean **isAvailableInDocument**(java.lang.String bookmarkID)`

Returns whether the bookmark has been added to the document during writing; if so, it can be used to create hyperlink.

org.tanjakostic.jcleancim.docgen.collector
Interface ClassDoc

All Superinterfaces:
ObjectDoc, RawData

public interface **ClassDoc**
extends ObjectDoc

Data required for documentation of classes.

Here the layout you may use:

```
 getHeadingText()
     getInheritancePath()               // if not empty and if enabled by configuration, you
can print
     getDescription()                   // (includes constraints, if enabled by
configuration)
     getDiagramDocs()           // loop and create figures
     getAttributesDoc()         // if getAttributesDoc().notEmpty() then create table
     getAssociationEndsDoc()    // if getAssociationEndsDoc().notEmpty() then create table
     getOperationsDoc()         // if getOperationsDoc().notEmpty() then create table
```

Field Summary

public static final	ANY_DA_OR_CDC_HEADING_FORMAT (deprecated, abstract, informative, custom), alias, name Value: **%s%s (%s)**
public static final	CLASS_TITLE_PREFIX_FMT_IEC61850 Value: **<<%s>>**
public static final	CONSTRAINTS_TXT_CIM (CIM) Text to append to description if constraints need to be explicitly printed. Value: **Constraints:**
public static final	CONSTRAINTS_TXT_IEC61850 (IEC61850) Text to append to description if constraints need to be explicitly printed. Value: **Conditions:**
public static final	INHERITANCE_PATH_PREFIX Prefix when introducing inheritance path. Value: **Inheritance path =**
public static final	INHERITANCE_PATH_SEP Separator when printing inheritance path. Value: **:**

public static final	LN_HEADING_FORMAT (deprecated, abstract, informative, custom), alias, nbsp, name Value: **%sLN: %s%sName: %s**
public static final	NO_ALIAS_HEADING_FORMAT (deprecated, abstract, informative, custom), name, (kindLabel) Value: **%s%s%s**
public static final	OLDNAME_FMT At present, used for some 61850 classes only (based on tagged value). Value: **(old name = %s)**
public static final	OTHER_WITH_ALIAS_HEADING_FORMAT (deprecated, abstract, informative, custom), alias, name, (stereotype) Value: **%s%s (%s%s)**
public static final	PRIM_DA_HEADING_FORMAT (deprecated, abstract, informative, custom), alias, name, kindLabel Value: **%s%s (%s %s)**

Fields inherited from interface org.tanjakostic.jcleancim.docgen.collector.ObjectDoc

DEFAULT_PREFIX_FMT

Method Summary

abstract PropertiesDoc	getAssocEndsDoc() Returns the documentation for all the association ends on the 'other' side of this class, suitable to be printed as a table.
abstract PropertiesDoc	getAttributesDoc() Returns the documentation for all the attributes of this class, suitable to be printed as a table.
abstract java.lang.String	getClassPlaceholderName() Returns the *qualified* name of the class, with the separator appropriate for placeholder.
abstract java.util.List	getDiagramDocs() Returns the documentation for all the diagrams of this class.
abstract TextDescription	getInheritancePath() Returns all superclasses formatted as inheritance path, suitable to be printed in a single paragraph.
abstract PropertiesDoc	getOperationsDoc() Returns the documentation for all the operations of this class, suitable to be printed as a table.

Methods inherited from interface org.tanjakostic.jcleancim.docgen.collector.ObjectDoc

getBmRegistry, getBookmarkID, getDescription, getDocgenCfg, getHeadingText

Methods inherited from interface org.tanjakostic.jcleancim.docgen.collector.RawData

copyCell, copyNonEmptyCell, getCell, getCells, hasKey, putCell, putCellNonEmpty

Fields

(continued from last page)

CLASS_TITLE_PREFIX_FMT_IEC61850

`public static final java.lang.String` **`CLASS_TITLE_PREFIX_FMT_IEC61850`**

Constant value: **`<<%s>>`**

NO_ALIAS_HEADING_FORMAT

`public static final java.lang.String` **`NO_ALIAS_HEADING_FORMAT`**

(deprecated, abstract, informative, custom), name, (kindLabel)
Constant value: **`%s%s%s`**

LN_HEADING_FORMAT

`public static final java.lang.String` **`LN_HEADING_FORMAT`**

(deprecated, abstract, informative, custom), alias, nbsp, name
Constant value: **`%sLN: %s%sName: %s`**

PRIM_DA_HEADING_FORMAT

`public static final java.lang.String` **`PRIM_DA_HEADING_FORMAT`**

(deprecated, abstract, informative, custom), alias, name, kindLabel
Constant value: **`%s%s (%s %s)`**

ANY_DA_OR_CDC_HEADING_FORMAT

`public static final java.lang.String` **`ANY_DA_OR_CDC_HEADING_FORMAT`**

(deprecated, abstract, informative, custom), alias, name
Constant value: **`%s%s (%s)`**

OTHER_WITH_ALIAS_HEADING_FORMAT

`public static final java.lang.String` **`OTHER_WITH_ALIAS_HEADING_FORMAT`**

(deprecated, abstract, informative, custom), alias, name, (stereotype)
Constant value: **`%s%s (%s%s)`**

INHERITANCE_PATH_PREFIX

`public static final java.lang.String` **`INHERITANCE_PATH_PREFIX`**

Prefix when introducing inheritance path.
Constant value: **`Inheritance path =`**

INHERITANCE_PATH_SEP

`public static final java.lang.String` **`INHERITANCE_PATH_SEP`**

Separator when printing inheritance path.
Constant value: **`:`**

(continued from last page)

OLDNAME_FMT

`public static final java.lang.String` **`OLDNAME_FMT`**

At present, used for some 61850 classes only (based on tagged value).
Constant value: **`(old name = %s)`**

CONSTRAINTS_TXT_IEC61850

`public static final java.lang.String` **`CONSTRAINTS_TXT_IEC61850`**

(IEC61850) Text to append to description if constraints need to be explicitly printed.
Constant value: **`Conditions:`**

CONSTRAINTS_TXT_CIM

`public static final java.lang.String` **`CONSTRAINTS_TXT_CIM`**

(CIM) Text to append to description if constraints need to be explicitly printed.
Constant value: **`Constraints:`**

Methods

getClassPlaceholderName

`public abstract java.lang.String` **`getClassPlaceholderName`**`()`

Returns the *qualified* name of the class, with the separator appropriate for placeholder. This gives the writer the name of the class present as placeholder in the template, and to avoid search through the model.

getInheritancePath

`public abstract TextDescription` **`getInheritancePath`**`()`

Returns all superclasses formatted as inheritance path, suitable to be printed in a single paragraph. Consider printing only if the returned value is not empty.

getAttributesDoc

`public abstract PropertiesDoc` **`getAttributesDoc`**`()`

Returns the documentation for all the attributes of this class, suitable to be printed as a table. Use `PropertiesDoc.notEmpty()` to see whether there is anything to print.

getAssocEndsDoc

`public abstract PropertiesDoc` **`getAssocEndsDoc`**`()`

Returns the documentation for all the association ends on the 'other' side of this class, suitable to be printed as a table. Use `PropertiesDoc.notEmpty()` to see whether there is anything to print.

getOperationsDoc

`public abstract PropertiesDoc` **`getOperationsDoc`**`()`

Returns the documentation for all the operations of this class, suitable to be printed as a table. Use `PropertiesDoc.notEmpty()` to see whether there is anything to print.

(continued from last page)

getDiagramDocs

`public abstract java.util.List` **`getDiagramDocs`**`()`

Returns the documentation for all the diagrams of this class.

org.tanjakostic.jcleancim.docgen.collector
Interface ClassScl

public interface **ClassScl**
extends

Method Summary

abstract java.util.List	getData()
abstract java.lang.String	getEnd()
abstract java.lang.String	getStart()
abstract java.lang.String	toXml(boolean prettyPrint)

Methods

getStart

```
public abstract java.lang.String getStart()
```

getData

```
public abstract java.util.List getData()
```

getEnd

```
public abstract java.lang.String getEnd()
```

toXml

```
public abstract java.lang.String toXml(boolean prettyPrint)
```

org.tanjakostic.jcleancim.docgen.collector
Class ColumnSpec

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.docgen.collector.ColumnSpec
```

All Implemented Interfaces:
RawData

public class **ColumnSpec**
extends java.lang.Object
implements RawData

Column specification.

Field Summary

public static final	DEFVAL_TAG Value: **Col**

Constructor Summary

public	ColumnSpec(java.lang.String tag, java.lang.String attrTag, java.lang.String label, java.lang.String docID, int relativeWidth, boolean formatted) Constructor; allows to specify any tag name.

Method Summary

java.lang.String	copyCell(RawData src, java.lang.String key)
java.lang.String	copyNonEmptyCell(RawData src, java.lang.String key)
static ColumnSpec	createFmted(int relWidth, java.lang.String attrTag, java.lang.String docID, java.lang.String label)
static ColumnSpec	createUnfmted(int relWidth, java.lang.String attrTag, java.lang.String docID, java.lang.String label)
java.lang.String	getAttrTag() name of XML attribute printed in this column
java.lang.String	getCell(java.lang.String key)
java.util.Map	getCells()
java.lang.String	getDocID() docID for label (XML only)
java.lang.String	getLabel() Word label = XML doc (translatable)

int	getRelWidth() relative columns width; useful for Word only (not printed in XML)
java.lang.String	getTag() XML element name
boolean	hasKey(java.lang.String key)
boolean	isFormatted() if true, the instance data content may be formatted (and is translatable)
java.lang.String	putCell(java.lang.String key, java.lang.String value)
java.lang.String	putCellNonEmpty(java.lang.String key, java.lang.String value)

Methods inherited from class `java.lang.Object`
clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface `org.tanjakostic.jcleancim.docgen.collector.RawData`
copyCell, copyNonEmptyCell, getCell, getCells, hasKey, putCell, putCellNonEmpty

Fields

DEFVAL_TAG

```
public static final java.lang.String DEFVAL_TAG
```

Constant value: **`Col`**

Constructors

ColumnSpec

```
public ColumnSpec(java.lang.String tag,
                  java.lang.String attrTag,
                  java.lang.String label,
                  java.lang.String docID,
                  int relativeWidth,
                  boolean formatted)
```

Constructor; allows to specify any `tag` name.

Parameters:

`tag`
`attrTag`
`label`
`docID`
`relativeWidth`
`formatted`

Methods

(continued from last page)

createFmted

```
public static ColumnSpec createFmted(int relWidth,
         java.lang.String attrTag,
         java.lang.String docID,
         java.lang.String label)
```

createUnfmted

```
public static ColumnSpec createUnfmted(int relWidth,
         java.lang.String attrTag,
         java.lang.String docID,
         java.lang.String label)
```

getLabel

```
public final java.lang.String getLabel()
```

Word label = XML doc (translatable)

getDocID

```
public java.lang.String getDocID()
```

docID for label (XML only)

getTag

```
public final java.lang.String getTag()
```

XML element name

getAttrTag

```
public java.lang.String getAttrTag()
```

name of XML attribute printed in this column

getRelWidth

```
public final int getRelWidth()
```

relative columns width; useful for Word only (not printed in XML)

isFormatted

```
public boolean isFormatted()
```

if true, the instance data content may be formatted (and is translatable)

putCell

```
public final java.lang.String putCell(java.lang.String key,
         java.lang.String value)
```

(continued from last page)

putCellNonEmpty

```
public final java.lang.String putCellNonEmpty(java.lang.String key,
         java.lang.String value)
```

copyCell

```
public final java.lang.String copyCell(RawData src,
         java.lang.String key)
```

copyNonEmptyCell

```
public final java.lang.String copyNonEmptyCell(RawData src,
         java.lang.String key)
```

hasKey

```
public final boolean hasKey(java.lang.String key)
```

getCells

```
public final java.util.Map getCells()
```

getCell

```
public final java.lang.String getCell(java.lang.String key)
```

org.tanjakostic.jcleancim.docgen.collector
Interface DocCollector

All Known Implementing Classes:
DocCollectorImpl

public interface **DocCollector**
extends

Collects documentation content for the model packages available in `UmlModel`, according to configuration, without generating any document. Results are available with `getFreeFormDocumentation()` and `getFixedFormDocumentation()`.

Method Summary

abstract boolean	`addSkippedInformativePackage(java.lang.String qName)` Must be called by every newly created `PackageDoc` that is to be skipped.
abstract void	`addToFlattened(ClassDoc classDoc)` Must be called for every newly created `ClassDoc` that need not be skipped.
abstract void	`addToFlattened(PackageDoc packageDoc)` Must be called for every newly created `PackageDoc` that need not be skipped.
abstract void	`addToScoped(PackageDoc packageDoc)` Must be called by every newly created `PackageDoc` that need not be skipped, if `owner` is in scope and it needs to be included in the name space: adds the `packageDoc` under appropriate nature and name space, no-op otherwise.
abstract void	`collect(UmlModel model)` Collects recursively documentation from UML model packages, their sub-packages, etc.
abstract BookmarkRegistry	`getBmRegistry()` Returns bookmark registry populated from the UML model.
abstract DocgenConfig	`getDocgenCfg()` Returns configuration according to which the documentation is collected for generation.
abstract FixedFormDocumentation	`getFixedFormDocumentation()` Creates if not yet called and then returns documentation per package, indexed by nature.
abstract FreeFormDocumentation	`getFreeFormDocumentation()` Creates if not yet called and then returns documentation per package, indexed by package name for easy reference; if there were any two packages with the same name, retains only the first one collected.
abstract boolean	`isFromUml()` Returns whether this collector has been created from a UML model (as opposed to pure API calls).

Methods

(continued from last page)

collect

`public abstract void **collect**(UmlModel model)`

Collects recursively documentation from UML model packages, their sub-packages, etc.

Parameters:
`model` - UML model.

Throws:
`UnsupportedOperationException` - if this instance has not been created with an uderlying UML model.

---

getFreeFormDocumentation

`public abstract FreeFormDocumentation **getFreeFormDocumentation**()`

Creates if not yet called and then returns documentation per package, indexed by package name for easy reference; if there were any two packages with the same name, retains only the first one collected. This presentation is convenient for free selection of packages, in any order, by any model nature.

---

getFixedFormDocumentation

`public abstract FixedFormDocumentation **getFixedFormDocumentation**()`

Creates if not yet called and then returns documentation per package, indexed by nature. This presentation is convenient for nature-dependent selection of packages.

---

addToFlattened

`public abstract void **addToFlattened**(PackageDoc packageDoc)`

Must be called for every newly created `PackageDoc` that need not be skipped.

Parameters:
`packageDoc` - package documentation to retain.

---

addToFlattened

`public abstract void **addToFlattened**(ClassDoc classDoc)`

Must be called for every newly created `ClassDoc` that need not be skipped.

Parameters:
`classDoc` - class documentation to retain.

---

addToScoped

`public abstract void **addToScoped**(PackageDoc packageDoc)`

Must be called by every newly created `PackageDoc` that need not be skipped, if `owner` is in scope and it needs to be included in the name space: adds the `packageDoc` under appropriate nature and name space, no-op otherwise.

Parameters:
`packageDoc`

---

addSkippedInformativePackage

`public abstract boolean **addSkippedInformativePackage**(java.lang.String qName)`

(continued from last page)

Must be called by every newly created `PackageDoc` that is to be skipped. Returns whether `qName` has been added to the collection of skipped informative package names. Because it is intended to be used for logging only, we don't need objects (strings are enough).

Parameters:

`qName` - qualified name of the package.

getBmRegistry

```
public abstract BookmarkRegistry getBmRegistry()
```

Returns bookmark registry populated from the UML model.

getDocgenCfg

```
public abstract DocgenConfig getDocgenCfg()
```

Returns configuration according to which the documentation is collected for generation.

isFromUml

```
public abstract boolean isFromUml()
```

Returns whether this collector has been created from a UML model (as opposed to pure API calls).

org.tanjakostic.jcleancim.docgen.collector
Class DocgenConfig

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.docgen.collector.DocgenConfig
```

public class **DocgenConfig**
extends java.lang.Object

Subset of `Config` relevant for documentation generation.

Field Summary

public final	basicPckNames
public final	cdcPckNames
public final	daPckNames
public final	dataIndexPckNames
public final	doAbbrPckNames
public final	docgenPckNames
public final	enumsXmlPckNames
public final	fcPckName
public final	includeInf
public final	includeInheritancePath
public final	includeInhFromMetamodel
public final	includeNonPublic
public final	keepHtml
public final	lnMapPckName
public final	lnPckNames
public final	owners
public final	presCondPckName

public final	showCustomStereotypes
public final	showNamespacePackages
public final	skipForCustomStereotypes
public final	trgOpPckName
public final	useHyperlinks
public final	writeUmlTypes

Constructor Summary

public	DocgenConfig(boolean includeInf, boolean includeNonPublic, boolean printHtml, boolean showCustomStereotypes, java.util.Collection skipForCustomStereotypes, java.util.Collection showNamespacePackages, boolean includeInheritancePath) Constructor for CIM-like simple printing; for XML all CIM owners are included.
public	DocgenConfig(boolean includeInf, boolean includeNonPublic, boolean printHtml, boolean showCustomStereotypes, java.util.Collection skipForCustomStereotypes, java.util.Collection showNamespacePackages, boolean useHyperlinks, boolean includeInheritancePath, java.util.EnumSet owners, boolean includeInhFromMetamodel, boolean writeUmlTypes, java.util.Collection docgenPckNames, java.util.Collection dataIndexPckNames, java.lang.String lnMapPckName, java.lang.String presCondPckName, java.lang.String fcPckName, java.lang.String trgOpPckName, java.util.Collection doAbbrPckNames, java.util.Collection enumsXmlPckNames, java.util.Collection lnPckNames, java.util.Collection cdcPckNames, java.util.Collection daPckNames, java.util.Collection basicPckNames) Constructor for IEC61850 printing.
public	DocgenConfig(Config cfg) Constructor.

Methods inherited from class `java.lang.Object`

`clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait`

Fields

includeInf

`public final boolean` **`includeInf`**

includeNonPublic

`public final boolean` **`includeNonPublic`**

keepHtml

public final boolean **keepHtml**

showCustomStereotypes

public final boolean **showCustomStereotypes**

skipForCustomStereotypes

public final java.util.Collection **skipForCustomStereotypes**

showNamespacePackages

public final java.util.Collection **showNamespacePackages**

useHyperlinks

public final boolean **useHyperlinks**

includeInheritancePath

public final boolean **includeInheritancePath**

owners

public final java.util.EnumSet **owners**

includeInhFromMetamodel

public final boolean **includeInhFromMetamodel**

writeUmlTypes

public final boolean **writeUmlTypes**

docgenPckNames

public final java.util.Collection **docgenPckNames**

(continued from last page)

dataIndexPckNames

public final java.util.Collection **dataIndexPckNames**

lnMapPckName

public final java.lang.String **lnMapPckName**

presCondPckName

public final java.lang.String **presCondPckName**

fcPckName

public final java.lang.String **fcPckName**

trgOpPckName

public final java.lang.String **trgOpPckName**

doAbbrPckNames

public final java.util.Collection **doAbbrPckNames**

enumsXmlPckNames

public final java.util.Collection **enumsXmlPckNames**

lnPckNames

public final java.util.Collection **lnPckNames**

cdcPckNames

public final java.util.Collection **cdcPckNames**

(continued from last page)

daPckNames

```
public final java.util.Collection daPckNames
```

basicPckNames

```
public final java.util.Collection basicPckNames
```

Constructors

DocgenConfig

```
public DocgenConfig(boolean includeInf,
                    boolean includeNonPublic,
                    boolean printHtml,
                    boolean showCustomStereotypes,
                    java.util.Collection skipForCustomStereotypes,
                    java.util.Collection showNamespacePackages,
                    boolean includeInheritancePath)
```

Constructor for CIM-like simple printing; for XML all CIM owners are included.

Parameters:

`includeInf`
`includeNonPublic`
`printHtml`
`includeInheritancePath`

DocgenConfig

```
public DocgenConfig(boolean includeInf,
                    boolean includeNonPublic,
                    boolean printHtml,
                    boolean showCustomStereotypes,
                    java.util.Collection skipForCustomStereotypes,
                    java.util.Collection showNamespacePackages,
                    boolean useHyperlinks,
                    boolean includeInheritancePath,
                    java.util.EnumSet owners,
                    boolean includeInhFromMetamodel,
                    boolean writeUmlTypes,
                    java.util.Collection docgenPckNames,
                    java.util.Collection dataIndexPckNames,
                    java.lang.String lnMapPckName,
                    java.lang.String presCondPckName,
                    java.lang.String fcPckName,
                    java.lang.String trgOpPckName,
                    java.util.Collection doAbbrPckNames,
                    java.util.Collection enumsXmlPckNames,
                    java.util.Collection lnPckNames,
                    java.util.Collection cdcPckNames,
                    java.util.Collection daPckNames,
                    java.util.Collection basicPckNames)
```

Constructor for IEC61850 printing.

DocgenConfig

```
public DocgenConfig(Config cfg)
```

(continued from last page)

Constructor.

Parameters:
`cfg`

org.tanjakostic.jcleancim.docgen.collector
Interface EntryDoc

All Superinterfaces:
RawData

All Known Implementing Classes:
EntryDocImpl

public interface **EntryDoc**
extends RawData

Record (table row) representation of an object.

Nested Class Summary

class	EntryDoc.Kind EntryDoc.Kind

Field Summary

public static final	SEPARATOR Separator of "cells" for poor-man printing in e.g. Value: |

Method Summary

abstract AGSpec	getAttrGroupSpec() Returns non-null instance if this is EntryDoc.Kind.groupSubhead, null otherwise.
abstract java.lang.String	getBookmarkID() Returns (potentially null) bookmark ID for referenceable items, e.g., enum literals.
abstract FormatInfo	getFormatInfo() Returns format information about the formattable cell at index j, null if this entry has no formattable cell, or if the formattable cell is actually not formatted.
abstract EntryDoc.Kind	getKind() Returns the kind of this entry; useful for formatting.
abstract java.lang.String[]	getValues() Returns array of values for columns (i.e., row content).
abstract java.lang.String	toCsv() Returns a single string of comma-separated items in a row, and Util.NL between the rows.

Methods inherited from interface org.tanjakostic.jcleancim.docgen.collector.RawData

copyCell, copyNonEmptyCell, getCell, getCells, hasKey, putCell, putCellNonEmpty

Fields

(continued from last page)

SEPARATOR

`public static final java.lang.String` **`SEPARATOR`**

Separator of "cells" for poor-man printing in e.g. toString() method.
Constant value: **|**

Methods

getValues

`public abstract java.lang.String[]` **`getValues`**`()`

Returns array of values for columns (i.e., row content).

getKind

`public abstract EntryDoc.Kind` **`getKind`**`()`

Returns the kind of this entry; useful for formatting.

getAttrGroupSpec

`public abstract AGSpec` **`getAttrGroupSpec`**`()`

Returns non-null instance if this is `EntryDoc.Kind.groupSubhead`, null otherwise.

getFormatInfo

`public abstract FormatInfo` **`getFormatInfo`**`()`

Returns format information about the formattable cell at index `j`, null if this entry has no formattable cell, or if the formattable cell is actually not formatted.

getBookmarkID

`public abstract java.lang.String` **`getBookmarkID`**`()`

Returns (potentially null) bookmark ID for referenceable items, e.g., enum literals.

toCsv

`public abstract java.lang.String` **`toCsv`**`()`

Returns a single string of comma-separated items in a row, and `Util.NL` between the rows. The last character is *not* `Util.NL`, but the last value in the last cell.

org.tanjakostic.jcleancim.docgen.collector
Class EntryDoc.Kind

```
java.lang.Object
  +-java.lang.Enum
    +-org.tanjakostic.jcleancim.docgen.collector.EntryDoc.Kind
```

All Implemented Interfaces:
java.io.Serializable, java.lang.Comparable

---

public static final class **EntryDoc.Kind**
extends java.lang.Enum

Describes the kind of an entry, to facilitate document generation formatting.

Field Summary

public static final	columnLabels An entry of this kind represents the row that contains labels for columns of the table.
public static final	data An entry of this kind represents the row with data for properties of a class, or an arbitrary set of values.
public static final	groupSubhead For IEC61850, we need to sometimes print fancy tables, with the attributes categorised into groups.
public static final	tableName For IEC61850, we need to sometimes print fancy tables, with the very first heading row containing the name of the table.

Method Summary

static EntryDoc.Kind	valueOf(java.lang.String name)
static EntryDoc.Kind[]	values()

Methods inherited from class `java.lang.Enum`

clone, compareTo, equals, finalize, getDeclaringClass, hashCode, name, ordinal, toString, valueOf

Methods inherited from class `java.lang.Object`

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface `java.lang.Comparable`

compareTo

Fields

tableName

`public static final org.tanjakostic.jcleancim.docgen.collector.EntryDoc.Kind` **`tableName`**

For IEC61850, we need to sometimes print fancy tables, with the very first heading row containing the name of the table. An entry of this kind represents the row that is the table name, so the first value contains that name and the remaining values are empty strings.

When printing a row from this entry, the cells of the table need to be merged, the shadding needs to be applied, the style of the row needs to be set to column heading style, and the row needs to be set as heading row.

columnLabels

`public static final org.tanjakostic.jcleancim.docgen.collector.EntryDoc.Kind` **`columnLabels`**

An entry of this kind represents the row that contains labels for columns of the table.

When printing a row from this entry, the shadding needs to be applied, the style of the row needs to be set to column heading style, and the row needs to be set as heading row.

groupSubhead

`public static final org.tanjakostic.jcleancim.docgen.collector.EntryDoc.Kind` **`groupSubhead`**

For IEC61850, we need to sometimes print fancy tables, with the attributes categorised into groups. Each such group has a name and is followed by entries that describe attributes. An entry of this kind represents the row that is a sub-head of the group, so the first value contains that name and the remaining values are empty strings.

When printing a row from this entry, the cells of the table need to be merged, the shadding needs to be applied, and the style of the row needs to be set to column heading style (but not as heading row).

data

`public static final org.tanjakostic.jcleancim.docgen.collector.EntryDoc.Kind` **`data`**

An entry of this kind represents the row with data for properties of a class, or an arbitrary set of values. One of the values (typically: description) may contain formatting. If the configuration requires to respect formatting, it needs to be processed appropriately before actual printing.

Methods

values

`public static EntryDoc.Kind[]` **`values`**`()`

valueOf

`public static EntryDoc.Kind` **`valueOf`**`(java.lang.String name)`

org.tanjakostic.jcleancim.docgen.collector
Interface FigureDoc

All Superinterfaces:
ObjectDoc, RawData

All Known Implementing Classes:
FigureDocImpl

---

public interface **FigureDoc**
extends ObjectDoc

Data required for documentation of figures.

Here the layout you may use for a diagram within the package or class:

```
getFigureFile()
[Figure # - ] + getCaptionText()
[Figure #: ] + getDescription()
```

or the legacy one, with the figure introduction before the figure (when org.tanjakostic.jcleancim.common.Config#KEY_DOCGEN_WORD_INTRO_TO_FIGURE_BEFORE=true):

```
[Figure # ] + getIntroText()
getFigureFile()
[Figure # - ] + getCaptionText()
getDescription()
```

A writer needs to supply what is enclosed in [] and for the rest you call methods of this interface.

Field Summary

public static final	CAPTION_TEXT_FORMAT Value: **%s**
public static final	INTRO_TEXT_FORMAT Value: **shows %s.**

Fields inherited from interface org.tanjakostic.jcleancim.docgen.collector.ObjectDoc

DEFAULT_PREFIX_FMT

Method Summary

abstract java.lang.String	getCaptionText() Returns text that describes the caption.
abstract java.io.File	getFigureFile() Returns file with the figure.

abstract java.lang.String	getIntroText() Returns text that will introduce the table.

Methods inherited from interface org.tanjakostic.jcleancim.docgen.collector.ObjectDoc
getBmRegistry, getBookmarkID, getDescription, getDocgenCfg, getHeadingText

Methods inherited from interface org.tanjakostic.jcleancim.docgen.collector.RawData
copyCell, copyNonEmptyCell, getCell, getCells, hasKey, putCell, putCellNonEmpty

Fields

INTRO_TEXT_FORMAT

`public static final java.lang.String` **`INTRO_TEXT_FORMAT`**

Constant value: **`shows %s.`**

CAPTION_TEXT_FORMAT

`public static final java.lang.String` **`CAPTION_TEXT_FORMAT`**

Constant value: **`%s`**

Methods

getIntroText

`public abstract java.lang.String` **`getIntroText`**`()`

Returns text that will introduce the table. E.g. "Figure 23 **{introText}**".

getFigureFile

`public abstract java.io.File` **`getFigureFile`**`()`

Returns file with the figure.

getCaptionText

`public abstract java.lang.String` **`getCaptionText`**`()`

Returns text that describes the caption. E.g., "Figure 23 - **{captionText}**".

org.tanjakostic.jcleancim.docgen.collector
Class FixedFormDocumentation

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.docgen.collector.FixedFormDocumentation
```

public class **FixedFormDocumentation**
extends java.lang.Object

Documentation in the fixed form, such as for printing content per model nature and per name space (as with XML output). The scope or retained packages is limitted according to configuration.

Constructor Summary

public	FixedFormDocumentation(java.util.Map nsPackageDocs) Constructor.

Method Summary

java.util.Map	getNsPackageDocs() Returns retained scoped package docs per nature and per name space.

Methods inherited from class `java.lang.Object`
clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Constructors

FixedFormDocumentation

public **FixedFormDocumentation**(java.util.Map nsPackageDocs)

Constructor.

Parameters:
nsPackageDocs

Methods

getNsPackageDocs

public java.util.Map **getNsPackageDocs**()

Returns retained scoped package docs per nature and per name space.

org.tanjakostic.jcleancim.docgen.collector
Class FormatInfo

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.docgen.collector.FormatInfo
```

public class **FormatInfo**
extends java.lang.Object

Constructor Summary

public	FormatInfo(TextDescription.TextKind kind, java.lang.Integer fmtIdx) Constructor; if you don't need formatting, just use null.

Method Summary

java.lang.Integer	getFormattedColumnIdx() Returns the index of the column that may contain formatting.
TextDescription.TextKind	getKind() Returns kind of this format.
java.lang.String	toString()

Methods inherited from class java.lang.Object
clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Constructors

FormatInfo

```
public FormatInfo(TextDescription.TextKind kind,
                  java.lang.Integer fmtIdx)
```

Constructor; if you don't need formatting, just use null.

Parameters:
kind - kind of formatting
fmtIdx - index of the formattable column

Methods

getKind

```
public TextDescription.TextKind getKind()
```

Returns kind of this format.

getFormattedColumnIdx

`public java.lang.Integer` **`getFormattedColumnIdx`**`()`

Returns the index of the column that may contain formatting.

toString

`public java.lang.String` **`toString`**`()`

org.tanjakostic.jcleancim.docgen.collector
Class FreeFormDocumentation

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.docgen.collector.FreeFormDocumentation
```

public class **FreeFormDocumentation**
extends java.lang.Object

Documentation in the free form, such as for printing content simply per package or class name (like for MS Word documentation). The scope of retained packages is limitted according to configuration.

Constructor Summary

public	FreeFormDocumentation(ModelFinder modelFinder, BookmarkRegistry bmRegistry, java.util.Map packageDocs, java.util.Map classDocs) Constructor.

Method Summary

BookmarkRegistry	getBmRegistry()
ClassDoc	getClassDoc(java.lang.String qName)
java.util.Map	getClassDocs()
ModelFinder	getModelFinder()
PackageDoc	getPackageDoc(java.lang.String name)
java.util.Map	getPackageDocs()

Methods inherited from class `java.lang.Object`

`clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait`

Constructors

FreeFormDocumentation

```
public FreeFormDocumentation(ModelFinder modelFinder,
                             BookmarkRegistry bmRegistry,
                             java.util.Map packageDocs,
                             java.util.Map classDocs)
```

Constructor.

Parameters:

`modelFinder` - model facade; if null, most of placeholders will have error.

(continued from last page)

`bmRegistry` - registry of bookmarks, i.e., contains maps between bookmark IDs (unique strings) and UML objects; can be used for generating hyperlinks.
`packageDocs` - "flattened" map of package documentation instances, with package name as key (to allow to quickly find the package name from what is read in the placeholder). If null or empty, placeholders dealing with packages will all have error and empty content.
`classDocs` - "flattened" map of class documentation instances, with *qualified* class name as key (to allow to quickly find the package name from what is read in the placeholder). If null or empty, placeholders dealing with classes will all have error and empty content.

Methods

getModelFinder

`public ModelFinder getModelFinder()`

getBmRegistry

`public BookmarkRegistry getBmRegistry()`

getPackageDocs

`public java.util.Map getPackageDocs()`

getPackageDoc

`public PackageDoc getPackageDoc(java.lang.String name)`

getClassDocs

`public java.util.Map getClassDocs()`

getClassDoc

`public ClassDoc getClassDoc(java.lang.String qName)`

org.tanjakostic.jcleancim.docgen.collector
Class GroupsSpec

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.docgen.collector.GroupsSpec
```

public class **GroupsSpec**
extends java.lang.Object

Field Summary

public static final	DACategories
public static final	DOCategories

Constructor Summary

public	GroupsSpec(java.lang.String name, java.util.Collection agSpecs)

Method Summary

java.util.Collection	getAgSpecs()
static java.util.List	getGroups()
java.lang.String	getName()
static java.util.Map	getPredefinedGroupsSpecs() Returns all the predefined categories.

Methods inherited from class `java.lang.Object`
clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Fields

DACategories

public static final org.tanjakostic.jcleancim.docgen.collector.GroupsSpec **DACategories**

DOCategories

public static final org.tanjakostic.jcleancim.docgen.collector.GroupsSpec **DOCategories**

(continued from last page)

Constructors

GroupsSpec

```
public GroupsSpec(java.lang.String name,
                  java.util.Collection agSpecs)
```

Methods

getPredefinedGroupsSpecs

```
public static java.util.Map getPredefinedGroupsSpecs()
```

Returns all the predefined categories.

getGroups

```
public static java.util.List getGroups()
```

getName

```
public java.lang.String getName()
```

getAgSpecs

```
public java.util.Collection getAgSpecs()
```

org.tanjakostic.jcleancim.docgen.collector
Class IDHelper

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.docgen.collector.IDHelper
```

public class **IDHelper**
extends java.lang.Object

Creates identifiers. Primary usage is for referencing items in documents.

Method Summary

`java.lang.String`	`createDocID(java.lang.String prefix, java.lang.String ending)` Returns a concatenation of qualified object name and `ending`arguments combined with an internal counter.
`static IDHelper`	`instance()` Accessor to a singleton instance.

Methods inherited from class `java.lang.Object`
`clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait`

Methods

instance

```
public static IDHelper instance()
```

Accessor to a singleton instance.

createDocID

```
public java.lang.String createDocID(java.lang.String prefix,
         java.lang.String ending)
```

Returns a concatenation of qualified object name and `ending`arguments combined with an internal counter.

Parameters:

`prefix` - (if not null or trimmed and empty) prefix to start the ID with; typically, a qualified name of an item for which to create a document identifier.
`ending` - (if not null or trimmed and empty) suffix to append, to enhance human readability.

org.tanjakostic.jcleancim.docgen.collector
Interface ModelFinder

All Known Implementing Classes:
ModelFinderImpl

public interface **ModelFinder**
extends

Thin set of methods, allowing us to do document generation tests without actually having the full model loaded and built from EA file.

Method Summary

abstract java.lang.String	`findAttributeValue(java.lang.String className, java.lang.String attributeName)` Returns value of first attribute `attributeName` of first class `className` when found, null otherwise.
abstract java.lang.String	`findClassName(java.lang.String packageName, java.lang.String className)` Returns the name of the first class `className` withing the first package `packageName` when found, null otherwise.
abstract java.io.File	`findDiagramFile(java.lang.String containerName, java.lang.String diagramName)` Returns file containing the first `diagramName` on the first container `containerName` when found, null otherwise.
abstract TextDescription	`findDiagramNote(java.lang.String containerName, java.lang.String diagramName)` Returns the note (description) of the first `diagramName` on the first container `containerName` when found, null otherwise.
abstract java.lang.String	`findIec61850NsName(java.lang.String className)` Returns the name space name for the IEC 61850 namespace class `className` when found, null otherwise.

Methods

findAttributeValue

```
public abstract java.lang.String findAttributeValue(java.lang.String className,
         java.lang.String attributeName)
```

Returns value of first attribute `attributeName` of first class `className` when found, null otherwise.

findDiagramFile

```
public abstract java.io.File findDiagramFile(java.lang.String containerName,
         java.lang.String diagramName)
```

Returns file containing the first `diagramName` on the first container `containerName` when found, null otherwise. Note that the diagram container could be either package or class; if there is a diagram with the same name on a package and on a class, the package diagram is returned.

findDiagramNote

```
public abstract TextDescription findDiagramNote(java.lang.String containerName,
         java.lang.String diagramName)
```

Returns the note (description) of the first `diagramName` on the first container `containerName` when found, null otherwise. Note that the diagram container could be either package or class; if there is a diagram with the same name on a package and on a class, the package diagram is returned.

findClassName

```
public abstract java.lang.String findClassName(java.lang.String packageName,
         java.lang.String className)
```

Returns the name of the first class `className` withing the first package `packageName` when found, null otherwise.

findIec61850NsName

```
public abstract java.lang.String findIec61850NsName(java.lang.String className)
```

Returns the name space name for the IEC 61850 namespace class `className` when found, null otherwise.

org.tanjakostic.jcleancim.docgen.collector
Interface ObjectDoc

All Superinterfaces:
RawData

All Subinterfaces:
ClassDoc, FigureDoc, PackageDoc, PropertiesDoc

All Known Implementing Classes:
AbstractObjectDoc

---

public interface **ObjectDoc**
extends RawData

Interface common to most kinds of documentation for the model.

Field Summary

public static final	DEFAULT_PREFIX_FMT Value: **(%s)**

Method Summary

abstract BookmarkRegistry	getBmRegistry() Returns the populated bookmark registry.
abstract java.lang.String	getBookmarkID() Returns an ID guaranteed to be unique for the model; usable for referencing such as e.g., hyperlinks (as anchor id in HTML or bookmark in Word) or references (as id in XML).
abstract TextDescription	getDescription() Returns description of this model element.
abstract DocgenConfig	getDocgenCfg() Returns document generation specific configuration.
abstract java.lang.String	getHeadingText() Returns text to be used for chapter heading for this model element; may be empty.

Methods inherited from interface org.tanjakostic.jcleancim.docgen.collector.RawData
copyCell, copyNonEmptyCell, getCell, getCells, hasKey, putCell, putCellNonEmpty

---

Fields

DEFAULT_PREFIX_FMT

public static final java.lang.String **DEFAULT_PREFIX_FMT**

Constant value: **(%s)**

Methods

getDocgenCfg

`public abstract DocgenConfig getDocgenCfg()`

Returns document generation specific configuration.

getHeadingText

`public abstract java.lang.String getHeadingText()`

Returns text to be used for chapter heading for this model element; may be empty.

getDescription

`public abstract TextDescription getDescription()`

Returns description of this model element.

getBmRegistry

`public abstract BookmarkRegistry getBmRegistry()`

Returns the populated bookmark registry.

getBookmarkID

`public abstract java.lang.String getBookmarkID()`

Returns an ID guaranteed to be unique for the model; usable for referencing such as e.g., hyperlinks (as anchor id in HTML or bookmark in Word) or references (as id in XML).

org.tanjakostic.jcleancim.docgen.collector
Interface PackageDoc

All Superinterfaces:
ObjectDoc, RawData

public interface **PackageDoc**
extends ObjectDoc

Data required for documentation of packages; documentation includes classes and sub-packages. Creating this instance for a root package results in creation of the doc data for the whole package contents.

Here the layout you may use for "regular" package, with `PlaceholderSpec.Kind.PACKAGE`:

```
 getHeadingText()
     getGenHeadingText()
         getNsUriAndPrefix()     // if not empty and if enabled by configuration, you can
print
         getDescription()
         getDiagramDocs()        // loop
     getClassDocs()              // loop
     getChildPackageDocs()       // loop
```

This kind of documentation is needed for printing the full content of the relevant packages in both CIM and IEC61850 domains.

If the package has been configured to print data index in place of `PlaceholderSpec.Kind.DATA_INDEX` (i.e., `Config.getValidationPackagesDataIndex()` is not empty), then `getDataIndexDoc()` will be non-null and can be used. This kind of documentation is needed for data index clauses in IEC61850-7-4 and IEC61850-7-3.

If the package has been configured to print enums as XML in place of `PlaceholderSpec.Kind.SCL_ENUMS` (i.e., `Config.getValidationIec61850PackagesEnumsXml()` is not empty), then `getEnumsPackageScl()` will be non-null and can be used. This kind of documentation is needed for annexes listing enums as XML in IEC61850-7-4 and IEC61850-7-3.

If the package has been configured to print the functional constraints table in place of `PlaceholderSpec.Kind.FCS` (i.e., `Config.getValidationIec61850PackageFc()` is not null), then `getFcPackageDoc()` will be non-null and can be used. This kind of documentation is needed for a subclause in IEC61850-7-2 and an annex in IEC61850-7-3.

If the package has been configured to print the trigger options table in place of `PlaceholderSpec.Kind.TRGOPS` (i.e., `Config.getValidationIec61850PackageTrgOp()` is not null), then `getTrgOpPackageDoc()` will be non-null and can be used. This kind of documentation is needed for a subclause in IEC61850-7-2.

If the package has been configured to print data object abbreviations in place of `PlaceholderSpec.Kind.ABBREVIATIONS` (i.e., `Config.getValidationIec61850PackagesDoAbbr()` is not null), then `getAbbrPackageDoc()` will be non-null and can be used. This kind of documentation is needed for Abbreviations clause in IEC61850-7-4.

If the package has been configured to print presence conditions in place of `PlaceholderSpec.Kind.PRES_CONDITIONS` (i.e., `Config.getValidationIec61850PackagePresCond()` is not null), then `getPresCondPackageDoc()` will be non-null and can be used. This kind of documentation is needed for Presence conditions clause in IEC61850-7-3.

If the package has been configured to print LN mapings in place of `PlaceholderSpec.Kind.LNMAP_PACKAGE` (i.e., `Config.getValidationIec61850PackageLnMaps()` is not null), then `getLnMapPackageDoc()` will be non-null and can be used. This kind of documentation is needed for one clause in IEC61850-7-4 (where we have to show tables of "mappings" between requirements LNs of IEC61850-5, and real LNs of IEC61850-7-4).

Field Summary

public static final	HANGING_PARA_TITLE IEC does not allow hanging paragraphs, so we include a sub-clause per package. Value: **General**
public static final	HEADING_FORMAT (deprecated, informative, custom), package label, name Value: **%s%s%s**
public static final	LNPKG_HEADING_FORMAT (IEC 61850) Format to use for logical node packages: alias (UML name). Value: **%s (%s)**
public static final	NAMESPACE_FORMAT nsuri = uri, nsprefix = prefix Value: **[%s = %s, %s = %s]**
public static final	PKG_TITLE_PREFIX_FMT_IEC61850 Value: **<<%s>>**

Fields inherited from interface org.tanjakostic.jcleancim.docgen.collector.ObjectDoc
DEFAULT_PREFIX_FMT

Method Summary

abstract PropertiesDoc	getAbbrPackageDoc()
abstract java.util.List	getChildPackageDocs() Returns documentation for all the child packages of this package.
abstract java.util.List	getClassDocs() Returns documentation for all the classes in this package.
abstract PropertiesDoc	getDataIndexDoc()
abstract PackageScl	getEnumsPackageScl()
abstract PropertiesDoc	getFcPackageDoc()
abstract java.util.List	getFigureDocs() Returns documentation for all the figures in this package.
abstract java.lang.String	getGenHeadingText() To avoid hanging paragraphs (i.e., those with some text but without title), ensure to include a "general" heading and print the doc and diagrams of the package under it.
abstract PropertiesDoc	getLnMapPackageDoc()
abstract java.lang.String	getModelName() Returns name of the model package (the one with nature) to which it belongs.
abstract NamespaceInfo	getNamespaceInfo() Returns name space information if it is defined for the package, null otherwise.

abstract Nature	getNature() Returns nature of the package.
abstract TextDescription	getNsUriAndPrefix() Returns formatted namespace URI and prefix, suitable to be printed in a single paragraph.
abstract java.lang.String	getPackageName() Returns the name of the package.
abstract PropertiesDoc	getPresCondPackageDoc()
abstract PropertiesDoc	getTrgOpPackageDoc()

Methods inherited from interface org.tanjakostic.jcleancim.docgen.collector.ObjectDoc
getBmRegistry, getBookmarkID, getDescription, getDocgenCfg, getHeadingText

Methods inherited from interface org.tanjakostic.jcleancim.docgen.collector.RawData
copyCell, copyNonEmptyCell, getCell, getCells, hasKey, putCell, putCellNonEmpty

Fields

PKG_TITLE_PREFIX_FMT_IEC61850

public static final java.lang.String **PKG_TITLE_PREFIX_FMT_IEC61850**

Constant value: **<<%s>>**

HEADING_FORMAT

public static final java.lang.String **HEADING_FORMAT**

(deprecated, informative, custom), package label, name
Constant value: **%s%s%s**

LNPKG_HEADING_FORMAT

public static final java.lang.String **LNPKG_HEADING_FORMAT**

(IEC 61850) Format to use for logical node packages: alias (UML name).
Constant value: **%s (%s)**

HANGING_PARA_TITLE

public static final java.lang.String **HANGING_PARA_TITLE**

IEC does not allow hanging paragraphs, so we include a sub-clause per package.
Constant value: **General**

NAMESPACE_FORMAT

public static final java.lang.String **NAMESPACE_FORMAT**

(continued from last page)

nsuri = uri, nsprefix = prefix
Constant value: **[%s = %s, %s = %s]**

Methods

getPackageName

`public abstract java.lang.String` **`getPackageName`**`()`

Returns the name of the package. This gives the writer the name of the package present as placeholder in the template, and to avoid search through the model.

getNamespaceInfo

`public abstract` `NamespaceInfo` **`getNamespaceInfo`**`()`

Returns name space information if it is defined for the package, null otherwise. This is the means to identify content for fixed-format documentation output (such as XML).

getModelName

`public abstract java.lang.String` **`getModelName`**`()`

Returns name of the model package (the one with nature) to which it belongs.

getNature

`public abstract` `Nature` **`getNature`**`()`

Returns nature of the package.

getGenHeadingText

`public abstract java.lang.String` **`getGenHeadingText`**`()`

To avoid hanging paragraphs (i.e., those with some text but without title), ensure to include a "general" heading and print the doc and diagrams of the package under it.

getNsUriAndPrefix

`public abstract` `TextDescription` **`getNsUriAndPrefix`**`()`

Returns formatted namespace URI and prefix, suitable to be printed in a single paragraph. Consider printing only if the returned value is not empty.

getFigureDocs

`public abstract java.util.List` **`getFigureDocs`**`()`

Returns documentation for all the figures in this package.

getClassDocs

`public abstract java.util.List` **`getClassDocs`**`()`

Returns documentation for all the classes in this package.

(continued from last page)

getChildPackageDocs

`public abstract java.util.List getChildPackageDocs()`

Returns documentation for all the child packages of this package.

getDataIndexDoc

`public abstract PropertiesDoc getDataIndexDoc()`

getLnMapPackageDoc

`public abstract PropertiesDoc getLnMapPackageDoc()`

getPresCondPackageDoc

`public abstract PropertiesDoc getPresCondPackageDoc()`

getFcPackageDoc

`public abstract PropertiesDoc getFcPackageDoc()`

getTrgOpPackageDoc

`public abstract PropertiesDoc getTrgOpPackageDoc()`

getAbbrPackageDoc

`public abstract PropertiesDoc getAbbrPackageDoc()`

getEnumsPackageScl

`public abstract PackageScl getEnumsPackageScl()`

org.tanjakostic.jcleancim.docgen.collector
Interface PackageScl

public interface **PackageScl**
extends

Field Summary

public static final	SCL_ENUM_HEADING_DEFAULT simple, without explicit package name (like for a single dedicated annex) Value: **SCL enumerations**
public static final	SCL_ENUM_HEADING_FORMAT_WITH_PCK_NAME with explicit package name (in case you chain a couple of packages) Value: **SCL enumerations (from %s)**

Method Summary

abstract java.util.List	getClassScls() Class as XML strings.
abstract java.lang.String	getHeadingText() Returns text that can be used as heading of a chapter.
abstract boolean	notEmpty() Use this one to see whether to create XML for this package at all.
abstract java.lang.String	toXml(boolean prettyPrint)

Fields

SCL_ENUM_HEADING_DEFAULT

public static final java.lang.String **SCL_ENUM_HEADING_DEFAULT**

simple, without explicit package name (like for a single dedicated annex)
Constant value: **SCL enumerations**

SCL_ENUM_HEADING_FORMAT_WITH_PCK_NAME

public static final java.lang.String **SCL_ENUM_HEADING_FORMAT_WITH_PCK_NAME**

with explicit package name (in case you chain a couple of packages)
Constant value: **SCL enumerations (from %s)**

Methods

notEmpty

public abstract boolean **notEmpty**()

(continued from last page)

Use this one to see whether to create XML for this package at all.

getHeadingText

`public abstract java.lang.String` **`getHeadingText`**`()`

Returns text that can be used as heading of a chapter.

getClassScls

`public abstract java.util.List` **`getClassScls`**`()`

Class as XML strings.

toXml

`public abstract java.lang.String` **`toXml`**`(boolean prettyPrint)`

org.tanjakostic.jcleancim.docgen.collector
Class PlaceholderSpec

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.docgen.collector.PlaceholderSpec
```

public class **PlaceholderSpec**
extends java.lang.Object

When using `FreeFormDocumentation`, templates for doc generation have to use labels to indicate where to insert the documentation of what element of the UML model into the output document. Currently needed and recognised labels to be used in the input templates are in the following format:

```
   startUmlDiagram.{packageName}.{diagramName}.endUml
   startUmlDiagNote.{packageName}.{diagramName}.endUml
   startUmlAttribute.{className}.{attributeName}.endUml
   startUmlIec61850NsName.{className}.endUml             (IEC 61850-7-*, for name space name)
   startUmlFile..endUml

   startUmlPresenceConditions.{packageName}.endUml (IEC 61850-7-3, for presence conditions
table)
   startUmlFCs.{packageName}.endUml                (IEC 61850-7-3 and IEC 61850-7-2, for FC
table)
   startUmlTrgOps.{packageName}.endUml             (IEC 61850-7-2, for TrgOp table)
   startUmlAbbreviations.{packageName}.endUml      (IEC 61850-7-4, for DO abbreviations
table)
   startUmlSclEnums.{packageName}.endUml           (IEC 61850-7-4, 7-3, for enums as XML)

   startUmlPackage.{packageName}.endUml
   startUmlClass.{packageName}.{className}.endUml
   startUmlDataIndex.{packageName}.endUml          (for IEC 61850-7-4,3, for data semantics
tables)
   startUmlLNMapPackage.{packageName}.endUml       (for IEC 61850-7-4, for function/LN map
tables)
```

The tokens enlosed in curly braces are the names of UML elements designating what needs to be inserted in place of the whole above string.

This format avoids us the need to define bookmarks in the input document (tedious and error-prone) and makes it simple to sequentially search the input document and insert the text and diagrams as they come.

Usage

Instance of kind `PlaceholderSpec.Kind.UNSUPPORTED` always returns non-null error that you may want to use to replace the placeholder to indicate failure. Instances of kind `PlaceholderSpec.Kind.FILE` are the simplest as they need not parse anything, so no errors and no saved tokens.

The instances of other kinds do need to parse the placeholder and will have tokens set as follows:

- first token (`PlaceholderSpec.Kind.PRES_CONDITIONS`, `PlaceholderSpec.Kind.ABBREVIATIONS`, `PlaceholderSpec.Kind.SCL_ENUMS` , `PlaceholderSpec.Kind.PACKAGE`, `PlaceholderSpec.Kind.LNMAP_PACKAGE`, `PlaceholderSpec.Kind.DATA_INDEX`, `PlaceholderSpec.Kind.FCS`, `PlaceholderSpec.Kind.TRGOPS`) or

- both tokens (`PlaceholderSpec.Kind.ATTRIBUTE`, `PlaceholderSpec.Kind.DIAGRAM`, `PlaceholderSpec.Kind.DIAG_NOTE`, `PlaceholderSpec.Kind.CLASS`).

In case the parsed token(s) is null or empty, the instance will contain a non-null error string. So, you will want to check first for error (null means no errors) before passing tokens for search, and in case of error, you may want to replace the placeholder to indicate failure.

Important: To have correct headings and paragraph formats, ensure you use the following placeholders in a heading paragraph:

- `PlaceholderSpec.Kind.PACKAGE`,
- `PlaceholderSpec.Kind.CLASS`,
- `PlaceholderSpec.Kind.PRES_CONDITIONS`,
- `PlaceholderSpec.Kind.FCS`,
- `PlaceholderSpec.Kind.TRGOPS`,
- `PlaceholderSpec.Kind.SCL_ENUMS`,
- `PlaceholderSpec.Kind.LNMAP_PACKAGE`, and,
- `PlaceholderSpec.Kind.DATA_INDEX`.

Nested Class Summary

class	`PlaceholderSpec.Kind` PlaceholderSpec.Kind

Field Summary

protected static final	`END_UML` Value: **`endUml`**
public static final	`HL_MS_PATTERN`
public static final	`MS_PATTERN`
protected static final	`PH_LETTERS` Value: **`[ACDFILTPS]`**
public static final	`SEPARATOR` Value: **`.`**
protected static final	`START_UML` Value: **`startUml`**
protected static final	`WILDCARD` Value: **`*`**

Constructor Summary

public	`PlaceholderSpec(java.lang.String text)`

Method Summary

static java.lang.String	`constructAbbrPackagePlaceholderText(java.lang.String pckName)`

static java.lang.String	constructAttributePlaceholderText(java.lang.String className, java.lang.String attrName)
static java.lang.String	constructClassPlaceholderText(java.lang.String pckName, java.lang.String className)
static java.lang.String	constructDataIndexPlaceholderText(java.lang.String pckName)
static java.lang.String	constructDiagNotePlaceholderText(java.lang.String containerName, java.lang.String diagName)
static java.lang.String	constructDiagramPlaceholderText(java.lang.String containerName, java.lang.String diagName)
static java.lang.String	constructEnumPackagePlaceholderText(java.lang.String pckName)
static java.lang.String	constructFcsPackagePlaceholderText(java.lang.String pckName)
static java.lang.String	constructFilePlaceholderText()
static java.lang.String	constructIec61850NsNamePlaceholderText(java.lang.String className)
static java.lang.String	constructInternalHyperlinkPlaceholderText(java.lang.String umlObjectName, java.lang.String bookmarkID)
static java.lang.String	constructLNMapPackagePlaceholderText(java.lang.String pckName)
static java.lang.String	constructPackagePlaceholderText(java.lang.String pckName)
static java.lang.String	constructPresConditionsPackagePlaceholderText(java.lang.String pckName)
static java.lang.String	constructTrgOpsPackagePlaceholderText(java.lang.String pckName)
java.lang.String	getErrorText() Returns null if there are no parsing errors, the error message otherwise.
java.lang.String	getFirstToken()
PlaceholderSpec.Kind	getKind()
java.lang.String	getSecondToken()
static java.util.List	getSupportedFormats()
java.lang.String	getText() Returns the placeholder text.
java.lang.String	toString()

void	`updateModelErrorText()` Use this setter when finder could not find valid tokens in the model (e.g., format of the placeholder is ok, but the names do not match elements in the model).

Methods inherited from class `java.lang.Object`
`clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait`

Fields

START_UML

`protected static final java.lang.String` **`START_UML`**

Constant value: **`startUml`**

END_UML

`protected static final java.lang.String` **`END_UML`**

Constant value: **`endUml`**

SEPARATOR

`public static final java.lang.String` **`SEPARATOR`**

Constant value: **`.`**

WILDCARD

`protected static final java.lang.String` **`WILDCARD`**

Constant value: **`*`**

PH_LETTERS

`protected static final java.lang.String` **`PH_LETTERS`**

Constant value: **`[ACDFILTPS]`**

MS_PATTERN

`public static final java.lang.String` **`MS_PATTERN`**

HL_MS_PATTERN

`public static final java.lang.String` **`HL_MS_PATTERN`**

(continued from last page)

Constructors

PlaceholderSpec

```
public PlaceholderSpec(java.lang.String text)
```

Methods

constructFilePlaceholderText

```
public static java.lang.String constructFilePlaceholderText()
```

constructDiagramPlaceholderText

```
public static java.lang.String constructDiagramPlaceholderText(java.lang.String
containerName,
         java.lang.String diagName)
```

constructDiagNotePlaceholderText

```
public static java.lang.String constructDiagNotePlaceholderText(java.lang.String
containerName,
         java.lang.String diagName)
```

constructAttributePlaceholderText

```
public static java.lang.String constructAttributePlaceholderText(java.lang.String
className,
         java.lang.String attrName)
```

constructIec61850NsNamePlaceholderText

```
public static java.lang.String constructIec61850NsNamePlaceholderText(java.lang.String
className)
```

constructPresConditionsPackagePlaceholderText

```
public static java.lang.String
constructPresConditionsPackagePlaceholderText(java.lang.String pckName)
```

(continued from last page)

constructFcsPackagePlaceholderText

```
public static java.lang.String constructFcsPackagePlaceholderText(java.lang.String
pckName)
```

constructTrgOpsPackagePlaceholderText

```
public static java.lang.String constructTrgOpsPackagePlaceholderText(java.lang.String
pckName)
```

constructAbbrPackagePlaceholderText

```
public static java.lang.String constructAbbrPackagePlaceholderText(java.lang.String
pckName)
```

constructEnumPackagePlaceholderText

```
public static java.lang.String constructEnumPackagePlaceholderText(java.lang.String
pckName)
```

constructPackagePlaceholderText

```
public static java.lang.String constructPackagePlaceholderText(java.lang.String
pckName)
```

constructClassPlaceholderText

```
public static java.lang.String constructClassPlaceholderText(java.lang.String pckName,
         java.lang.String className)
```

constructLNMapPackagePlaceholderText

```
public static java.lang.String constructLNMapPackagePlaceholderText(java.lang.String
pckName)
```

constructDataIndexPlaceholderText

```
public static java.lang.String constructDataIndexPlaceholderText(java.lang.String
pckName)
```

(continued from last page)

constructInternalHyperlinkPlaceholderText

```
public static java.lang.String
constructInternalHyperlinkPlaceholderText(java.lang.String umlObjectName,
         java.lang.String bookmarkID)
```

getSupportedFormats

```
public static java.util.List getSupportedFormats()
```

getText

```
public java.lang.String getText()
```

Returns the placeholder text.

getKind

```
public PlaceholderSpec.Kind getKind()
```

getFirstToken

```
public java.lang.String getFirstToken()
```

getSecondToken

```
public java.lang.String getSecondToken()
```

getErrorText

```
public java.lang.String getErrorText()
```

Returns null if there are no parsing errors, the error message otherwise.

updateModelErrorText

```
public void updateModelErrorText()
```

Use this setter when finder could not find valid tokens in the model (e.g., format of the placeholder is ok, but the names do not match elements in the model).

toString

```
public java.lang.String toString()
```

org.tanjakostic.jcleancim.docgen.collector
Class PlaceholderSpec.Kind

```
java.lang.Object
  +-java.lang.Enum
    +-org.tanjakostic.jcleancim.docgen.collector.PlaceholderSpec.Kind
```

All Implemented Interfaces:
java.io.Serializable, java.lang.Comparable

---

public static final class **PlaceholderSpec.Kind**
extends java.lang.Enum

Field Summary

public static final	ABBREVIATIONS
public static final	ATTRIBUTE
public static final	CLASS
public static final	DATA_INDEX
public static final	DIAG_NOTE Note: We intentionally do not use 'DiagramNote' to avoid overlap with 'Diagram', for text search.
public static final	DIAGRAM
public static final	FCS
public static final	FILE
public static final	HYPERLINK This one is internal, never used by client.
public static final	IEC61850_NSNAME
public static final	LNMAP_PACKAGE
public static final	PACKAGE
public static final	PRES_CONDITIONS
public static final	SCL_ENUMS
public static final	TRGOPS

public static final	UNSUPPORTED Anything not matched by others.

Method Summary

static boolean	isForHeading(PlaceholderSpec.Kind kind) Returns whether kind is expected to be in the heading paragraph.
static PlaceholderSpec.Kind	valueOf(java.lang.String name)
static PlaceholderSpec.Kind[]	values()

Methods inherited from class `java.lang.Enum`

clone, compareTo, equals, finalize, getDeclaringClass, hashCode, name, ordinal, toString, valueOf

Methods inherited from class `java.lang.Object`

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface `java.lang.Comparable`

compareTo

Fields

FILE

public static final org.tanjakostic.jcleancim.docgen.collector.PlaceholderSpec.Kind **FILE**

DIAGRAM

public static final org.tanjakostic.jcleancim.docgen.collector.PlaceholderSpec.Kind **DIAGRAM**

DIAG_NOTE

public static final org.tanjakostic.jcleancim.docgen.collector.PlaceholderSpec.Kind **DIAG_NOTE**

Note: We intentionally do not use 'DiagramNote' to avoid overlap with 'Diagram', for text search.

ATTRIBUTE

public static final org.tanjakostic.jcleancim.docgen.collector.PlaceholderSpec.Kind **ATTRIBUTE**

(continued from last page)

IEC61850_NSNAME

public static final org.tanjakostic.jcleancim.docgen.collector.PlaceholderSpec.Kind **IEC61850_NSNAME**

PRES_CONDITIONS

public static final org.tanjakostic.jcleancim.docgen.collector.PlaceholderSpec.Kind **PRES_CONDITIONS**

FCS

public static final org.tanjakostic.jcleancim.docgen.collector.PlaceholderSpec.Kind **FCS**

TRGOPS

public static final org.tanjakostic.jcleancim.docgen.collector.PlaceholderSpec.Kind **TRGOPS**

ABBREVIATIONS

public static final org.tanjakostic.jcleancim.docgen.collector.PlaceholderSpec.Kind **ABBREVIATIONS**

SCL_ENUMS

public static final org.tanjakostic.jcleancim.docgen.collector.PlaceholderSpec.Kind **SCL_ENUMS**

PACKAGE

public static final org.tanjakostic.jcleancim.docgen.collector.PlaceholderSpec.Kind **PACKAGE**

CLASS

public static final org.tanjakostic.jcleancim.docgen.collector.PlaceholderSpec.Kind **CLASS**

(continued from last page)

DATA_INDEX

```
public static final org.tanjakostic.jcleancim.docgen.collector.PlaceholderSpec.Kind
DATA_INDEX
```

LNMAP_PACKAGE

```
public static final org.tanjakostic.jcleancim.docgen.collector.PlaceholderSpec.Kind
LNMAP_PACKAGE
```

HYPERLINK

```
public static final org.tanjakostic.jcleancim.docgen.collector.PlaceholderSpec.Kind
HYPERLINK
```

This one is internal, never used by client.

UNSUPPORTED

```
public static final org.tanjakostic.jcleancim.docgen.collector.PlaceholderSpec.Kind
UNSUPPORTED
```

Anything not matched by others.

Methods

values

```
public static PlaceholderSpec.Kind[] values()
```

valueOf

```
public static PlaceholderSpec.Kind valueOf(java.lang.String name)
```

isForHeading

```
public static boolean isForHeading(PlaceholderSpec.Kind kind)
```

Returns whether `kind` is expected to be in the heading paragraph.

org.tanjakostic.jcleancim.docgen.collector
Interface PropertiesDoc

All Superinterfaces:
ObjectDoc, RawData

All Known Implementing Classes:
AbstractPropertiesDoc

---

public interface **PropertiesDoc**
extends ObjectDoc

Set of methods to allow table generation for constituents.

Here the layout you may want to use:

```
   [Table # ] + getIntroText()
   [Table # - ] + getCaptionText()
   // create table of size getColumnCount() x getRowCount()
   if getTableName() non null // merge cells (the first head row) and print
   getColumnNames()           // print regular table head
   getCellValues()            // loop [i,j]. In row i: if isRowGroupSubhead() then merge
cells
```

For IEC 61850 tables, we need to support a "table title", and thus 2 heading rows, with both table title/name and actual column names. That row is present if `getTableName()` returns non-null value.

The method `getEntryDocs()` provides the list of individual entries, in case you need formatting other than table.

Field Summary

public static final	`INHERITED_FROM` Text to use to indicate inherited members. Value: **`inherited from:`**

Fields inherited from interface `org.tanjakostic.jcleancim.docgen.collector.ObjectDoc`
`DEFAULT_PREFIX_FMT`

Method Summary

abstract java.lang.String[]	`getBookmarkIDs()` Returns array of bookmark IDs for every row `i`, with potential null entries.
abstract java.lang.String	`getCaptionText()` Returns text that describes the caption (e.g., "Table 23 - **[captionText]**").
abstract java.lang.String[][]	`getCellValues()` Returns the full table values, including heading rows (table name and column names), any potential sub-head and the actual rows with values.

abstract int	getColumnCount() Returns number of columns for the table; must be same as TableSpec.colCount() from the columns spec returned by getTableSpec().
abstract java.util.List	getDataEntryDocs() Returns (unmodifiable list of) data entries only.
abstract java.util.List	getEntryDocs() Returns (unmodifiable list of) all entries.
abstract TextDescription.TextKind[]	getFormats() Returns array of applied formattings for the formattable cells in every row i.
abstract int	getHeadingEntriesCount() Returns the number of header entries (EntryDoc.Kind.tableName or EntryDoc.Kind.columnLabels) in this instance.
abstract java.lang.String	getIntroText() Returns text that will introduce the table (e.g.
abstract int	getRowCount() Returns number of rows for the table.
abstract EntryDoc.Kind[]	getRowKinds() Returns kinds for rows from getCellValues().
abstract java.lang.String	getTableName() Returns the name of the table; this will be the first heading row, when non-null.
abstract TableSpec	getTableSpec() Returns columns specification.
abstract boolean	notEmpty() Returns whether there are any non-heading entries (EntryDoc.Kind.groupSubhead or EntryDoc.Kind.data) in this instance.

Methods inherited from interface org.tanjakostic.jcleancim.docgen.collector.ObjectDoc
getBmRegistry, getBookmarkID, getDescription, getDocgenCfg, getHeadingText

Methods inherited from interface org.tanjakostic.jcleancim.docgen.collector.RawData
copyCell, copyNonEmptyCell, getCell, getCells, hasKey, putCell, putCellNonEmpty

Fields

INHERITED_FROM

public static final java.lang.String **INHERITED_FROM**

Text to use to indicate inherited members.
Constant value: **inherited from:**

Methods

(continued from last page)

notEmpty

`public abstract boolean **notEmpty**()`

Returns whether there are any non-heading entries (`EntryDoc.Kind.groupSubhead` or `EntryDoc.Kind.data`) in this instance. Use this one to determine whether to print table at all.

getIntroText

`public abstract java.lang.String **getIntroText**()`

Returns text that will introduce the table (e.g. "Table 23 **[introText]**").

getCaptionText

`public abstract java.lang.String **getCaptionText**()`

Returns text that describes the caption (e.g., "Table 23 - **[captionText]**").

getHeadingEntriesCount

`public abstract int **getHeadingEntriesCount**()`

Returns the number of header entries (`EntryDoc.Kind.tableName` or `EntryDoc.Kind.columnLabels`) in this instance.

getEntryDocs

`public abstract java.util.List **getEntryDocs**()`

Returns (unmodifiable list of) all entries.

getDataEntryDocs

`public abstract java.util.List **getDataEntryDocs**()`

Returns (unmodifiable list of) data entries only.

getTableName

`public abstract java.lang.String **getTableName**()`

Returns the name of the table; this will be the first heading row, when non-null.

getTableSpec

`public abstract TableSpec **getTableSpec**()`

Returns columns specification. If `getTableName()` is null, labels from the columns spec (or the corresponding entry) will be the first and only heading row, otherwise the second heading row.

getRowCount

`public abstract int **getRowCount**()`

Returns number of rows for the table.

(continued from last page)

getColumnCount

`public abstract int` **`getColumnCount`**`()`

Returns number of columns for the table; must be same as `TableSpec.colCount()` from the columns spec returned by `getTableSpec()`.

getCellValues

`public abstract java.lang.String[][]` **`getCellValues`**`()`

Returns the full table values, including heading rows (table name and column names), any potential sub-head and the actual rows with values. This is suitable to have uniform access for matrix-like kind of access.

getRowKinds

`public abstract EntryDoc.Kind[]` **`getRowKinds`**`()`

Returns kinds for rows from `getCellValues()`.

getFormats

`public abstract TextDescription.TextKind[]` **`getFormats`**`()`

Returns array of applied formattings for the formattable cells in every row `i`. If the formattable cell in a row contains no formatting at all, returns null at index `i`. If no row contains formatting, returns null.

The formattable column index `j` is always the same for this table, and can be obtained from `TableSpec.getFmtIdx()`. This helps optimise writing content in a format other than raw text.

getBookmarkIDs

`public abstract java.lang.String[]` **`getBookmarkIDs`**`()`

Returns array of bookmark IDs for every row `i`, with potential null entries.

org.tanjakostic.jcleancim.docgen.collector
Interface RawData

All Subinterfaces:
EntryDoc, ObjectDoc, ClassDoc, FigureDoc, PackageDoc, PropertiesDoc

All Known Implementing Classes:
AGSpec, ColumnSpec, RawDataImpl

---

public interface **RawData**
extends

Simple wrapper for a map of key/value pairs, used to store raw, non-formatted and non- modified data from the UML object for applications that do not work with UML objects and that may need to provide formatting for printing different than the default. It is useful to communicate disparate type information required for document generation in some formats.

Method Summary

abstract java.lang.String	copyCell(RawData src, java.lang.String key) Copies value for key existing in src, into this provider; no-op if src does not contain the key.
abstract java.lang.String	copyNonEmptyCell(RawData src, java.lang.String key) Copies non-empty value for key existing in src, into this provider; no-op if src does not contain the key, or if it contains the key but the value for that key is empty.
abstract java.lang.String	getCell(java.lang.String key) Returns value for key, null if key does not exist or if key is null.
abstract java.util.Map	getCells() Returns potentially empty map of key/value pairs.
abstract boolean	hasKey(java.lang.String key) If true, key is present.
abstract java.lang.String	putCell(java.lang.String key, java.lang.String value) Adds the value for key.
abstract java.lang.String	putCellNonEmpty(java.lang.String key, java.lang.String value) Adds the value for key if value is not empty.

Methods

putCell

```
public abstract java.lang.String putCell(java.lang.String key,
          java.lang.String value)
```

Adds the value for key.

Parameters:
key - non-null, non-empty key.
value - non-null, potentially empty value for the key.

(continued from last page)

Returns:
null on success, replaced value if it existed for the given key.

putCellNonEmpty

```
public abstract java.lang.String putCellNonEmpty(java.lang.String key,
         java.lang.String value)
```

Adds the `value` for `key` if `value` is not empty.

Parameters:
`key` - non-null, non-empty key.
`value` - non-null, potentially empty value for the key.

Returns:
FIXME

copyCell

```
public abstract java.lang.String copyCell(RawData src,
         java.lang.String key)
```

Copies value for `key` existing in `src`, into this provider; no-op if `src` does not contain the `key`.

copyNonEmptyCell

```
public abstract java.lang.String copyNonEmptyCell(RawData src,
         java.lang.String key)
```

Copies non-empty value for `key` existing in `src`, into this provider; no-op if `src` does not contain the `key`, or if it contains the `key` but the value for that key is empty.

hasKey

```
public abstract boolean hasKey(java.lang.String key)
```

If true, `key` is present.

getCells

```
public abstract java.util.Map getCells()
```

Returns potentially empty map of key/value pairs.

getCell

```
public abstract java.lang.String getCell(java.lang.String key)
```

Returns value for key, null if key does not exist or if `key` is null.

org.tanjakostic.jcleancim.docgen.collector
Class RawDataImpl

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.docgen.collector.RawDataImpl
```

All Implemented Interfaces:
RawData

public class **RawDataImpl**
extends java.lang.Object
implements RawData

Default implementation.

Constructor Summary

public	RawDataImpl()

Method Summary

java.lang.String	copyCell(RawData src, java.lang.String key)
java.lang.String	copyNonEmptyCell(RawData src, java.lang.String key)
java.lang.String	getCell(java.lang.String key)
java.util.Map	getCells()
boolean	hasKey(java.lang.String key)
java.lang.String	putCell(java.lang.String key, java.lang.String value)
java.lang.String	putCellNonEmpty(java.lang.String key, java.lang.String value)

Methods inherited from class `java.lang.Object`
clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface `org.tanjakostic.jcleancim.docgen.collector.RawData`
copyCell, copyNonEmptyCell, getCell, getCells, hasKey, putCell, putCellNonEmpty

Constructors

(continued from last page)

RawDataImpl

```
public RawDataImpl()
```

Methods

putCell

```
public final java.lang.String putCell(java.lang.String key,
         java.lang.String value)
```

copyCell

```
public java.lang.String copyCell(RawData src,
         java.lang.String key)
```

copyNonEmptyCell

```
public java.lang.String copyNonEmptyCell(RawData src,
         java.lang.String key)
```

putCellNonEmpty

```
public java.lang.String putCellNonEmpty(java.lang.String key,
         java.lang.String value)
```

hasKey

```
public boolean hasKey(java.lang.String key)
```

getCells

```
public java.util.Map getCells()
```

getCell

```
public java.lang.String getCell(java.lang.String key)
```

org.tanjakostic.jcleancim.docgen.collector
Class TableSpec

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.docgen.collector.TableSpec
```

public class **TableSpec**
extends java.lang.Object

Information required to describe a table and its columns for generating documentation.

Field Summary

public static final	ABBREVS
public static final	ASSOC_ENDS
public static final	ATTR_INDEX
public static final	ATTRS
public static final	CDA_ATTRS
public static final	CDC_ATTRS
public static final	CTA_ATTRS
public static final	CUSTOM_ASSOC_ENDS
public static final	CUSTOM_LITERALS
public static final	CUSTOM_OPERATIONS
public static final	FCS
public static final	FUNCTIONS
public static final	KEY_trgOp Value: **trgOp**
public static final	LITERALS
public static final	LN_ATTRS
public static final	ODA_ATTRS
public static final	OPERATIONS

public static final	PRES_CONDS
public static final	TRG_OPS

Constructor Summary

public	TableSpec(java.lang.String name, Nature nature, java.util.List colSpecs)

Method Summary

int	colCount() Returns the number of columns described with this instance.
java.util.List	getColSpecs() Returns (unmodifiable) list of its column specs.
java.lang.Integer	getFmtIdx() Returns the index of the column that may be formatted, null if no column needs formatting.
java.lang.String[]	getLabels() Returns (cloned) labels of columns for the table; if the table does not have a name, this may be used as the first and only heading row, otherwise this will be the second heading row.
java.lang.String	getName() Returns name of this table type.
Nature	getNature() Returns name of this table type.
static java.util.Map	getPredefinedTableSpecs() Returns all the predefined table formats.
int[]	getRelativeWidths() Returns (cloned) widths of columns in percentage of the full table width.
static java.util.List	getTableSpecs(Nature nature)

Methods inherited from class `java.lang.Object`

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Fields

KEY_trgOp

public static final java.lang.String **KEY_trgOp**

Constant value: **trgOp**

(continued from last page)

FCS

public static final org.tanjakostic.jcleancim.docgen.collector.TableSpec **FCS**

TRG_OPS

public static final org.tanjakostic.jcleancim.docgen.collector.TableSpec **TRG_OPS**

PRES_CONDS

public static final org.tanjakostic.jcleancim.docgen.collector.TableSpec **PRES_CONDS**

ABBREVS

public static final org.tanjakostic.jcleancim.docgen.collector.TableSpec **ABBREVS**

FUNCTIONS

public static final org.tanjakostic.jcleancim.docgen.collector.TableSpec **FUNCTIONS**

CTA_ATTRS

public static final org.tanjakostic.jcleancim.docgen.collector.TableSpec **CTA_ATTRS**

CDA_ATTRS

public static final org.tanjakostic.jcleancim.docgen.collector.TableSpec **CDA_ATTRS**

CDC_ATTRS

public static final org.tanjakostic.jcleancim.docgen.collector.TableSpec **CDC_ATTRS**

LN_ATTRS

public static final org.tanjakostic.jcleancim.docgen.collector.TableSpec **LN_ATTRS**

ATTR_INDEX

public static final org.tanjakostic.jcleancim.docgen.collector.TableSpec **ATTR_INDEX**

(continued from last page)

CUSTOM_LITERALS

public static final org.tanjakostic.jcleancim.docgen.collector.TableSpec **CUSTOM_LITERALS**

ODA_ATTRS

public static final org.tanjakostic.jcleancim.docgen.collector.TableSpec **ODA_ATTRS**

CUSTOM_OPERATIONS

public static final org.tanjakostic.jcleancim.docgen.collector.TableSpec **CUSTOM_OPERATIONS**

CUSTOM_ASSOC_ENDS

public static final org.tanjakostic.jcleancim.docgen.collector.TableSpec **CUSTOM_ASSOC_ENDS**

LITERALS

public static final org.tanjakostic.jcleancim.docgen.collector.TableSpec **LITERALS**

ATTRS

public static final org.tanjakostic.jcleancim.docgen.collector.TableSpec **ATTRS**

ASSOC_ENDS

public static final org.tanjakostic.jcleancim.docgen.collector.TableSpec **ASSOC_ENDS**

OPERATIONS

public static final org.tanjakostic.jcleancim.docgen.collector.TableSpec **OPERATIONS**

Constructors

(continued from last page)

TableSpec

```
public TableSpec(java.lang.String name,
                 Nature nature,
                 java.util.List colSpecs)
```

Methods

getPredefinedTableSpecs

```
public static java.util.Map getPredefinedTableSpecs()
```

Returns all the predefined table formats.

getTableSpecs

```
public static java.util.List getTableSpecs(Nature nature)
```

getName

```
public java.lang.String getName()
```

Returns name of this table type.

getNature

```
public Nature getNature()
```

Returns name of this table type.

getFmtIdx

```
public java.lang.Integer getFmtIdx()
```

Returns the index of the column that may be formatted, null if no column needs formatting.

Note that this is just the specification about the data, but the actual formatting needs to be enabled by the application (according to e.g. configuration), and then processed as desired.

colCount

```
public int colCount()
```

Returns the number of columns described with this instance.

getLabels

```
public java.lang.String[] getLabels()
```

Returns (cloned) labels of columns for the table; if the table does not have a name, this may be used as the first and only heading row, otherwise this will be the second heading row.

(continued from last page)

getRelativeWidths

```
public int[] getRelativeWidths()
```

Returns (cloned) widths of columns in percentage of the full table width. It is up to the implementation to ensure that the sum of values does not exceed 100.

getColSpecs

```
public java.util.List getColSpecs()
```

Returns (unmodifiable) list of its column specs.

org.tanjakostic.jcleancim.docgen.collector
Class WAX

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.docgen.collector.WAX
```

public class **WAX**
extends java.lang.Object

String constants used for XML doc generation.

Field Summary

public static final	A_abstract Value: **abstract**
public static final	A_alias Value: **alias**
public static final	A_aliasID Value: **aliasID**
public static final	A_bookmarkID Value: **bookmarkID**
public static final	A_caption Value: **caption**
public static final	A_captionID Value: **captionID**
public static final	A_cdcId Value: **cdcId**
public static final	A_cond Value: **cond**
public static final	A_deducedTypeText Value: **deducedTypeText**
public static final	A_defaultValue Value: **defaultValue**
public static final	A_deprecated Value: **deprecated**

public static final	A_desc Value: **desc**
public static final	A_descID Value: **descID**
public static final	A_dsCond Value: **dsCond**
public static final	A_dsPresCond Value: **dsPresCond**
public static final	A_dsPresCondArgs Value: **dsPresCondArgs**
public static final	A_dsPresCondArgsID Value: **dsPresCondArgsID**
public static final	A_fc Value: **fc**
public static final	A_iecRef Value: **iecRef**
public static final	A_ieeeRef Value: **ieeeRef**
public static final	A_img Value: **img**
public static final	A_informative Value: **informative**
public static final	A_inheritedFrom Value: **inheritedFrom**
public static final	A_introduction Value: **introduction**
public static final	A_introductionID Value: **introductionID**
public static final	A_kind Value: **kind**
public static final	A_literalVal Value: **literalVal**

public static final	A_lns Value: **lns**
public static final	A_maxValue Value: **maxValue**
public static final	A_minValue Value: **minValue**
public static final	A_mult Value: **mult**
public static final	A_myMult Value: **myMult**
public static final	A_name Value: **name**
public static final	A_presCond Value: **presCond**
public static final	A_presCondArgs Value: **presCondArgs**
public static final	A_presCondArgsID Value: **presCondArgsID**
public static final	A_rsName Value: **rsName**
public static final	A_signature Value: **signature**
public static final	A_subtitle Value: **subtitle**
public static final	A_subtitleID Value: **subtitleID**
public static final	A_superClass Value: **superClass**
public static final	A_text Value: **text**
public static final	A_textID Value: **textID**

public static final	A_title Value: **title**
public static final	A_titleID Value: **titleID**
public static final	A_transient Value: **transient**
public static final	A_type Value: **type**
public static final	A_typeKind Value: **typeKind**
public static final	A_underlyingControlType Value: **underlyingControlType**
public static final	A_underlyingType Value: **underlyingType**
public static final	CAT_daCtlMirror Value: **daCtlMirror**
public static final	CAT_daDesc Value: **daDesc**
public static final	CAT_daMeas Value: **daMeas**
public static final	CAT_daNull Value: **daNull**
public static final	CAT_daSetting Value: **daSetting**
public static final	CAT_daStatus Value: **daStatus**
public static final	CAT_daSubst Value: **daSubst**
public static final	CAT_daTracking Value: **daTracking**
public static final	CAT_doControl Value: **doControl**

public static final	CAT_doDesc Value: **doDesc**
public static final	CAT_doMeas Value: **doMeas**
public static final	CAT_doNull Value: **doNull**
public static final	CAT_doSetting Value: **doSetting**
public static final	CAT_doStatus Value: **doStatus**
public static final	CAT_doTracking Value: **doTracking**
public static final	CAT_SDOs Value: **SDOs**
public static final	CAT_ServiceParameters Value: **ServiceParameters**
public static final	E_AbbreviationsTable Value: **AbbreviationsTable**
public static final	E_AssociationEndsTable Value: **AssociationEndsTable**
public static final	E_AttributesTable Value: **AttributesTable**
public static final	E_CDCAttributesTable Value: **CDCAttributesTable**
public static final	E_ConstructedDAsTable Value: **ConstructedDAsTable**
public static final	E_CoreTypeAttributesTable Value: **CoreTypeAttributesTable**
public static final	E_DACategory Value: **DACategory**
public static final	E_DataIndexTable Value: **DataIndexTable**

public static final	E_Doc Value: **Doc**
public static final	E_DOCategory Value: **DOCategory**
public static final	E_FunctionalConstraintsTable Value: **FunctionalConstraintsTable**
public static final	E_FunctionsTable Value: **FunctionsTable**
public static final	E_IEC61850Domain Value: **IEC61850Domain**
public static final	E_IECDomainDoc Value: **IECDomainDoc**
public static final	E_IECDomainSpec Value: **IECDomainSpec**
public static final	E_LiteralsTable Value: **LiteralsTable**
public static final	E_LNAttributesTable Value: **LNAttributesTable**
public static final	E_OperationsTable Value: **OperationsTable**
public static final	E_OtherAttributesTable Value: **OtherAttributesTable**
public static final	E_PresenceConditionsTable Value: **PresenceConditionsTable**
public static final	E_PS Value: **PS**
public static final	E_TriggerOptionsTable Value: **TriggerOptionsTable**
public static final	LOC_instTag Value: **instTag**
public static final	LOC_isSpecial Value: **isSpecial**

public static final	LOC_tag Value: **tag**
public static final	V_typeKind_BASIC Value: **BASIC**
public static final	V_typeKind_CODED_ENUM Value: **CODED ENUM**
public static final	V_typeKind_CONSTRUCTED Value: **CONSTRUCTED**
public static final	V_typeKind_ENUM Value: **ENUMERATED**
public static final	V_typeKind_PACKED_LIST Value: **PACKED LIST**

Methods inherited from class `java.lang.Object`
`clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait`

Fields

E_IECDomainSpec

`public static final java.lang.String` **`E_IECDomainSpec`**

Constant value: **`IECDomainSpec`**

E_IECDomainDoc

`public static final java.lang.String` **`E_IECDomainDoc`**

Constant value: **`IECDomainDoc`**

E_Doc

`public static final java.lang.String` **`E_Doc`**

Constant value: **`Doc`**

E_IEC61850Domain

`public static final java.lang.String` **`E_IEC61850Domain`**

Constant value: **`IEC61850Domain`**

A_name

`public static final java.lang.String` **`A_name`**

Constant value: **`name`**

A_informative

`public static final java.lang.String` **`A_informative`**

Constant value: **`informative`**

A_deprecated

`public static final java.lang.String` **`A_deprecated`**

Constant value: **`deprecated`**

A_inheritedFrom

`public static final java.lang.String` **`A_inheritedFrom`**

Constant value: **`inheritedFrom`**

A_abstract

`public static final java.lang.String` **`A_abstract`**

Constant value: **`abstract`**

A_superClass

`public static final java.lang.String` **`A_superClass`**

Constant value: **`superClass`**

A_alias

`public static final java.lang.String` **`A_alias`**

Constant value: **`alias`**

A_aliasID

`public static final java.lang.String` **`A_aliasID`**

Constant value: **`aliasID`**

(continued from last page)

A_desc

`public static final java.lang.String` **`A_desc`**

Constant value: **`desc`**

A_descID

`public static final java.lang.String` **`A_descID`**

Constant value: **`descID`**

A_title

`public static final java.lang.String` **`A_title`**

Constant value: **`title`**

A_titleID

`public static final java.lang.String` **`A_titleID`**

Constant value: **`titleID`**

A_subtitle

`public static final java.lang.String` **`A_subtitle`**

Constant value: **`subtitle`**

A_subtitleID

`public static final java.lang.String` **`A_subtitleID`**

Constant value: **`subtitleID`**

A_introduction

`public static final java.lang.String` **`A_introduction`**

Constant value: **`introduction`**

A_introductionID

`public static final java.lang.String` **`A_introductionID`**

Constant value: **`introductionID`**

A_caption

`public static final java.lang.String` **`A_caption`**

(continued from last page)

Constant value: **`caption`**

A_captionID

`public static final java.lang.String` **`A_captionID`**

Constant value: **`captionID`**

A_img

`public static final java.lang.String` **`A_img`**

Constant value: **`img`**

A_literalVal

`public static final java.lang.String` **`A_literalVal`**

Constant value: **`literalVal`**

A_type

`public static final java.lang.String` **`A_type`**

Constant value: **`type`**

A_mult

`public static final java.lang.String` **`A_mult`**

Constant value: **`mult`**

A_myMult

`public static final java.lang.String` **`A_myMult`**

Constant value: **`myMult`**

A_presCond

`public static final java.lang.String` **`A_presCond`**

Constant value: **`presCond`**

A_presCondArgs

`public static final java.lang.String` **`A_presCondArgs`**

Constant value: **`presCondArgs`**

A_presCondArgsID

`public static final java.lang.String` **`A_presCondArgsID`**

Constant value: **`presCondArgsID`**

A_cond

`public static final java.lang.String` **`A_cond`**

Constant value: **`cond`**

A_dsPresCond

`public static final java.lang.String` **`A_dsPresCond`**

Constant value: **`dsPresCond`**

A_dsPresCondArgs

`public static final java.lang.String` **`A_dsPresCondArgs`**

Constant value: **`dsPresCondArgs`**

A_dsPresCondArgsID

`public static final java.lang.String` **`A_dsPresCondArgsID`**

Constant value: **`dsPresCondArgsID`**

A_dsCond

`public static final java.lang.String` **`A_dsCond`**

Constant value: **`dsCond`**

A_fc

`public static final java.lang.String` **`A_fc`**

Constant value: **`fc`**

A_transient

`public static final java.lang.String` **`A_transient`**

Constant value: **`transient`**

(continued from last page)

A_signature

`public static final java.lang.String` **`A_signature`**

Constant value: **`signature`**

A_underlyingType

`public static final java.lang.String` **`A_underlyingType`**

Constant value: **`underlyingType`**

A_underlyingControlType

`public static final java.lang.String` **`A_underlyingControlType`**

Constant value: **`underlyingControlType`**

A_kind

`public static final java.lang.String` **`A_kind`**

Constant value: **`kind`**

A_text

`public static final java.lang.String` **`A_text`**

Constant value: **`text`**

A_textID

`public static final java.lang.String` **`A_textID`**

Constant value: **`textID`**

A_deducedTypeText

`public static final java.lang.String` **`A_deducedTypeText`**

Constant value: **`deducedTypeText`**

A_typeKind

`public static final java.lang.String` **`A_typeKind`**

Constant value: **`typeKind`**

A_defaultValue

`public static final java.lang.String` **`A_defaultValue`**

(continued from last page)

Constant value: **defaultValue**

A_minValue

`public static final java.lang.String` **A_minValue**

Constant value: **minValue**

A_maxValue

`public static final java.lang.String` **A_maxValue**

Constant value: **maxValue**

A_bookmarkID

`public static final java.lang.String` **A_bookmarkID**

Constant value: **bookmarkID**

V_typeKind_BASIC

`public static final java.lang.String` **V_typeKind_BASIC**

Constant value: **BASIC**

V_typeKind_ENUM

`public static final java.lang.String` **V_typeKind_ENUM**

Constant value: **ENUMERATED**

V_typeKind_CODED_ENUM

`public static final java.lang.String` **V_typeKind_CODED_ENUM**

Constant value: **CODED ENUM**

V_typeKind_PACKED_LIST

`public static final java.lang.String` **V_typeKind_PACKED_LIST**

Constant value: **PACKED LIST**

V_typeKind_CONSTRUCTED

`public static final java.lang.String` **V_typeKind_CONSTRUCTED**

Constant value: **CONSTRUCTED**

A_rsName

`public static final java.lang.String` **`A_rsName`**

Constant value: **`rsName`**

A_ieeeRef

`public static final java.lang.String` **`A_ieeeRef`**

Constant value: **`ieeeRef`**

A_iecRef

`public static final java.lang.String` **`A_iecRef`**

Constant value: **`iecRef`**

A_cdcId

`public static final java.lang.String` **`A_cdcId`**

Constant value: **`cdcId`**

A_lns

`public static final java.lang.String` **`A_lns`**

Constant value: **`lns`**

E_FunctionalConstraintsTable

`public static final java.lang.String` **`E_FunctionalConstraintsTable`**

Constant value: **`FunctionalConstraintsTable`**

E_TriggerOptionsTable

`public static final java.lang.String` **`E_TriggerOptionsTable`**

Constant value: **`TriggerOptionsTable`**

E_PresenceConditionsTable

`public static final java.lang.String` **`E_PresenceConditionsTable`**

Constant value: **`PresenceConditionsTable`**

(continued from last page)

E_AbbreviationsTable

`public static final java.lang.String` **`E_AbbreviationsTable`**

Constant value: **`AbbreviationsTable`**

E_FunctionsTable

`public static final java.lang.String` **`E_FunctionsTable`**

Constant value: **`FunctionsTable`**

E_CoreTypeAttributesTable

`public static final java.lang.String` **`E_CoreTypeAttributesTable`**

Constant value: **`CoreTypeAttributesTable`**

E_ConstructedDAsTable

`public static final java.lang.String` **`E_ConstructedDAsTable`**

Constant value: **`ConstructedDAsTable`**

E_CDCAttributesTable

`public static final java.lang.String` **`E_CDCAttributesTable`**

Constant value: **`CDCAttributesTable`**

E_LNAttributesTable

`public static final java.lang.String` **`E_LNAttributesTable`**

Constant value: **`LNAttributesTable`**

E_DataIndexTable

`public static final java.lang.String` **`E_DataIndexTable`**

Constant value: **`DataIndexTable`**

E_OtherAttributesTable

`public static final java.lang.String` **`E_OtherAttributesTable`**

Constant value: **`OtherAttributesTable`**

E_LiteralsTable

`public static final java.lang.String` **`E_LiteralsTable`**

(continued from last page)

Constant value: **`LiteralsTable`**

E_OperationsTable

`public static final java.lang.String` **`E_OperationsTable`**

Constant value: **`OperationsTable`**

E_AssociationEndsTable

`public static final java.lang.String` **`E_AssociationEndsTable`**

Constant value: **`AssociationEndsTable`**

E_AttributesTable

`public static final java.lang.String` **`E_AttributesTable`**

Constant value: **`AttributesTable`**

E_PS

`public static final java.lang.String` **`E_PS`**

Constant value: **`PS`**

E_DOCategory

`public static final java.lang.String` **`E_DOCategory`**

Constant value: **`DOCategory`**

CAT_daNull

`public static final java.lang.String` **`CAT_daNull`**

Constant value: **`daNull`**

CAT_SDOs

`public static final java.lang.String` **`CAT_SDOs`**

Constant value: **`SDOs`**

CAT_daStatus

`public static final java.lang.String` **`CAT_daStatus`**

Constant value: **`daStatus`**

CAT_daMeas

`public static final java.lang.String` **`CAT_daMeas`**

Constant value: **`daMeas`**

CAT_daCtlMirror

`public static final java.lang.String` **`CAT_daCtlMirror`**

Constant value: **`daCtlMirror`**

CAT_daSubst

`public static final java.lang.String` **`CAT_daSubst`**

Constant value: **`daSubst`**

CAT_daSetting

`public static final java.lang.String` **`CAT_daSetting`**

Constant value: **`daSetting`**

CAT_daTracking

`public static final java.lang.String` **`CAT_daTracking`**

Constant value: **`daTracking`**

CAT_daDesc

`public static final java.lang.String` **`CAT_daDesc`**

Constant value: **`daDesc`**

CAT_ServiceParameters

`public static final java.lang.String` **`CAT_ServiceParameters`**

Constant value: **`ServiceParameters`**

E_DACategory

`public static final java.lang.String` **`E_DACategory`**

Constant value: **`DACategory`**

(continued from last page)

CAT_doNull

`public static final java.lang.String` **`CAT_doNull`**

Constant value: **`doNull`**

CAT_doDesc

`public static final java.lang.String` **`CAT_doDesc`**

Constant value: **`doDesc`**

CAT_doStatus

`public static final java.lang.String` **`CAT_doStatus`**

Constant value: **`doStatus`**

CAT_doMeas

`public static final java.lang.String` **`CAT_doMeas`**

Constant value: **`doMeas`**

CAT_doControl

`public static final java.lang.String` **`CAT_doControl`**

Constant value: **`doControl`**

CAT_doSetting

`public static final java.lang.String` **`CAT_doSetting`**

Constant value: **`doSetting`**

CAT_doTracking

`public static final java.lang.String` **`CAT_doTracking`**

Constant value: **`doTracking`**

LOC_tag

`public static final java.lang.String` **`LOC_tag`**

Constant value: **`tag`**

LOC_instTag

`public static final java.lang.String` **`LOC_instTag`**

(continued from last page)

Constant value: **`instTag`**

LOC_isSpecial

`public static final java.lang.String` **`LOC_isSpecial`**

Constant value: **`isSpecial`**

Package

org.tanjakostic.jcleancim.docgen.collector.impl

org.tanjakostic.jcleancim.docgen.collector.impl
Class AbstractObjectDoc

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.docgen.collector.impl.AbstractObjectDoc
```

All Implemented Interfaces:
ObjectDoc

Direct Known Subclasses:
AbstractPropertiesDoc, FigureDocImpl

---

public abstract class **AbstractObjectDoc**
extends java.lang.Object
implements ObjectDoc

Common implementation for any kind of object documentation.

Fields inherited from interface `org.tanjakostic.jcleancim.docgen.collector.ObjectDoc`
`DEFAULT_PREFIX_FMT`

Constructor Summary

protected	`AbstractObjectDoc(DocgenConfig docgenCfg, UmlObject o, java.lang.String what, TextDescription description, TextDescription htmlDescription, boolean ignoreDesc, java.lang.String headingText, java.lang.String bookmarkID, BookmarkRegistry bmRegistry)` "Centralised" constructor, allowing for instantiation both with and without a UML object, and with and without descriptions, as follows:

Method Summary

java.lang.String	`copyCell(RawData src, java.lang.String key)`
java.lang.String	`copyNonEmptyCell(RawData src, java.lang.String key)`
static java.lang.String	`createDocId(UmlObject obj, java.lang.String ending)`
static java.lang.String	`deduceBookmark(BookmarkRegistry bmRegistry, UmlObject obj)`
static java.lang.String	`deduceQualifiersPrefix(UmlObject o, java.util.Map builtIns, boolean withCustomStereotypes, java.lang.String fmt)` Invokes `deduceQualifiersPrefix(UmlObject, Map, List, boolean, UmlObject, String)` without `moreTokens` and `parent`.
static java.lang.String	`deduceQualifiersPrefix(UmlObject o, java.util.Map builtIns, boolean withCustomStereotypes, UmlObject parent, java.lang.String fmt)` Invokes `deduceQualifiersPrefix(UmlObject, Map, List, boolean, UmlObject, String)` without `moreTokens`.

static java.lang.String	deduceQualifiersPrefix(UmlObject o, java.util.Map builtIns, java.util.List moreTokens, boolean withCustomStereotypes, java.lang.String fmt) Invokes {link deduceQualifiersPrefix(UmlObject, Map, List, boolean, UmlObject, String) without parent.
static java.lang.String	deduceQualifiersPrefix(UmlObject o, java.util.Map builtIns, java.util.List moreTokens, boolean withCustomStereotypes, UmlObject parent, java.lang.String fmt) Returns deduced string formatted according to fmt, from deprecated and informative status, potentially more qualifiers, and potentially custom stereotypes.
void	filterClasses(UmlPackage p, java.util.Collection retainedNatives)
BookmarkRegistry	getBmRegistry()
java.lang.String	getBookmarkID()
java.lang.String	getCell(java.lang.String key)
java.util.Map	getCells()
TextDescription	getDescription() This default implementation returns what has been explicitly initialised in the call to the constructor.
DocgenConfig	getDocgenCfg()
java.lang.String	getHeadingText()
boolean	hasKey(java.lang.String key)
static void	log(org.apache.log4j.Logger logger, java.lang.String message)
java.lang.String	prepareForHyperlink(UmlObject targetObj) If hyperlink option is enabled, creates a hyperlink placeholder for targetObj whose text will be written instead of name, to be replaced by a hyperlink in additional pass.
java.lang.String	prepareForHyperlinkAdjustedName(UmlObject targetObj, java.lang.String nameToDisplay) Same as prepareForHyperlink(UmlObject) except that it creates the hyperlink placeholder with
java.lang.String	putCell(java.lang.String key, java.lang.String value)
java.lang.String	putCellNonEmpty(java.lang.String key, java.lang.String value)
boolean	toSkip(UmlObject o) Returns whether to skip object o, according to configuration.
java.lang.String	toString()
boolean	useHtml(TextDescription htmlDescription) Returns true if printing HTML is enabled and htmlDescription is not empty.

Methods inherited from class `java.lang.Object`

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.docgen.collector.ObjectDoc
getBmRegistry, getBookmarkID, getDescription, getDocgenCfg, getHeadingText

Methods inherited from interface org.tanjakostic.jcleancim.docgen.collector.RawData
copyCell, copyNonEmptyCell, getCell, getCells, hasKey, putCell, putCellNonEmpty

Constructors

AbstractObjectDoc

```
protected AbstractObjectDoc(DocgenConfig docgenCfg,
                            UmlObject o,
                            java.lang.String what,
                            TextDescription description,
                            TextDescription htmlDescription,
                            boolean ignoreDesc,
                            java.lang.String headingText,
                            java.lang.String bookmarkID,
                            BookmarkRegistry bmRegistry)
```

"Centralised" constructor, allowing for instantiation both with and without a UML object, and with and without descriptions, as follows:

If `object` is non-null (and `ignoreDesc=false`), its text and HTML description fields will be used, and raw data will be added for object name, alias, description (as HTML) and heading text; if `ignoreDesc=true`, no description raw data will be added.

If `object` is null, no raw data will be created at all; if `ignoreDesc=false`, the explicit `description` and `htmlDescription` will be used instead.

Instance description (returned by `getDescription()`) retained will be in HTML format only if configuration enables HTML printing and `htmlDescription` is not empty. Otherwise, the text description is retained. If `ignoreDesc=true`, returned values are just empty text or HTML description.

Parameters:

- `docgenCfg` - non-null document generation specific configuration.
- `o` - (possibly null) UML object.
- `what` - (possibly null) describes kind of properties of `object`; used to ensure unique aliases, descriptions, etc. when an object may have multiple groups of properties (in particular, class with its attributes, associations and operations).
- `description` - (possibly null) text format description.
- `htmlDescription` - (possibly null) HTML format description.
- `ignoreDesc` - whether to ignore description altogether.
- `headingText` - (possibly null) heading text, to be used as chapter title.
- `bookmarkID` - (possibly null) bookmark ID.
- `bmRegistry` - non-null (but potentially empty) bookmark registry.

Methods

log

```
protected static void log(org.apache.log4j.Logger logger,
          java.lang.String message)
```

deduceQualifiersPrefix

```
protected static java.lang.String deduceQualifiersPrefix(UmlObject o,
         java.util.Map builtIns,
         java.util.List moreTokens,
         boolean withCustomStereotypes,
         UmlObject parent,
         java.lang.String fmt)
```

Returns deduced string formatted according to `fmt`, from deprecated and informative status, potentially more qualifiers, and potentially custom stereotypes. You will typically use this with a format `fmt` that has some enclosure for a string (e.g., (), "") and ends with a white space, to be usable as prefix.

FIXME: This became a terrible method, but I had to move forward... Needs refactoring !

Parameters:

`o` - UML object
`builtIns` - map of built-in stereotypes applicable to `o` per model nature
`moreTokens` - (potentially null or empty) list of additional tokens, to append immediately after
`withCustomStereotypes` - whether to include custom (non-built-in) stereotypes
`parent` - when non-null, using custom stereotypes of parent instead of `o`; note that you must take care of appropriately passing in the `builtIns`
`fmt` - String format for single string

deduceQualifiersPrefix

```
protected static java.lang.String deduceQualifiersPrefix(UmlObject o,
         java.util.Map builtIns,
         boolean withCustomStereotypes,
         java.lang.String fmt)
```

Invokes `deduceQualifiersPrefix(UmlObject, Map, List, boolean, UmlObject, String)` without `moreTokens` and `parent`.

deduceQualifiersPrefix

```
protected static java.lang.String deduceQualifiersPrefix(UmlObject o,
         java.util.Map builtIns,
         boolean withCustomStereotypes,
         UmlObject parent,
         java.lang.String fmt)
```

Invokes `deduceQualifiersPrefix(UmlObject, Map, List, boolean, UmlObject, String)` without `moreTokens`.

deduceQualifiersPrefix

```
protected static java.lang.String deduceQualifiersPrefix(UmlObject o,
         java.util.Map builtIns,
         java.util.List moreTokens,
         boolean withCustomStereotypes,
         java.lang.String fmt)
```

Invokes {link `deduceQualifiersPrefix(UmlObject, Map, List, boolean, UmlObject, String)` without `parent`.

createDocId

```
protected final static java.lang.String createDocId(UmlObject obj,
         java.lang.String ending)
```

deduceBookmark

```
protected static java.lang.String deduceBookmark(BookmarkRegistry bmRegistry,
        UmlObject obj)
```

prepareForHyperlink

```
protected final java.lang.String prepareForHyperlink(UmlObject targetObj)
```

If hyperlink option is enabled, creates a hyperlink placeholder for `targetObj` whose text will be written instead of name, to be replaced by a hyperlink in additional pass. Otherwise, returns the name of `targetObj`.

prepareForHyperlinkAdjustedName

```
protected final java.lang.String prepareForHyperlinkAdjustedName(UmlObject targetObj,
        java.lang.String nameToDisplay)
```

Same as `prepareForHyperlink(UmlObject)` except that it creates the hyperlink placeholder with

useHtml

```
protected final boolean useHtml(TextDescription htmlDescription)
```

Returns true if printing HTML is enabled and `htmlDescription` is not empty.

toSkip

```
protected final boolean toSkip(UmlObject o)
```

Returns whether to skip object `o`, according to configuration.

filterClasses

```
protected final void filterClasses(UmlPackage p,
        java.util.Collection retainedNatives)
```

getDocgenCfg

```
public final DocgenConfig getDocgenCfg()
```

getHeadingText

```
public final java.lang.String getHeadingText()
```

getDescription

```
public TextDescription getDescription()
```

This default implementation returns what has been explicitly initialised in the call to the constructor. Override in case you need some special processing.

getBmRegistry

```
public BookmarkRegistry getBmRegistry()
```

getBookmarkID

```
public final java.lang.String getBookmarkID()
```

toString

```
public java.lang.String toString()
```

putCell

```
public final java.lang.String putCell(java.lang.String key,
         java.lang.String value)
```

copyCell

```
public final java.lang.String copyCell(RawData src,
         java.lang.String key)
```

copyNonEmptyCell

```
public final java.lang.String copyNonEmptyCell(RawData src,
         java.lang.String key)
```

putCellNonEmpty

```
public final java.lang.String putCellNonEmpty(java.lang.String key,
         java.lang.String value)
```

hasKey

```
public final boolean hasKey(java.lang.String key)
```

getCells

```
public final java.util.Map getCells()
```

getCell

```
public final java.lang.String getCell(java.lang.String key)
```

org.tanjakostic.jcleancim.docgen.collector.impl
Class AbstractPropertiesDoc

```
java.lang.Object
  |-org.tanjakostic.jcleancim.docgen.collector.impl.AbstractObjectDoc
    |-org.tanjakostic.jcleancim.docgen.collector.impl.AbstractPropertiesDoc
```

All Implemented Interfaces:
PropertiesDoc, ObjectDoc

Direct Known Subclasses:
Attributes61850Doc

---

public abstract class **AbstractPropertiesDoc**
extends AbstractObjectDoc
implements ObjectDoc, PropertiesDoc

Common implementation for properties documentation (table).

Nested Class Summary

class	AbstractPropertiesDoc.CellText AbstractPropertiesDoc.CellText

Fields inherited from interface org.tanjakostic.jcleancim.docgen.collector.ObjectDoc
DEFAULT_PREFIX_FMT

Fields inherited from interface org.tanjakostic.jcleancim.docgen.collector.PropertiesDoc
INHERITED_FROM

Fields inherited from interface org.tanjakostic.jcleancim.docgen.collector.ObjectDoc
DEFAULT_PREFIX_FMT

Constructor Summary

protected	AbstractPropertiesDoc(DocgenConfig docgenCfg, UmlObject object, java.lang.String what, java.lang.String introText, java.lang.String captionText, TableSpec colSpec, java.lang.String tableName, BookmarkRegistry bmRegistry) Creates an instance with a EntryDoc.Kind.tableName entry (if tableName is not null) and EntryDoc.Kind.columnLabels entry (from colSpec) in the list of entries; use when you want to add EntryDoc.Kind.data (and optionally, EntryDoc.Kind.groupSubhead) entries one by one, after some complex processing.

protected	AbstractPropertiesDoc(DocgenConfig docgenCfg, UmlObject object, java.lang.String what, TextDescription description, TextDescription htmlDescription, boolean ignoreDesc, java.lang.String headingText, java.lang.String introText, java.lang.String captionText, TableSpec colSpec, java.lang.String tableName, BookmarkRegistry bmRegistry) Same as AbstractPropertiesDoc(DocgenConfig, UmlObject, String, String, String, TableSpec, String, BookmarkRegistry) , but with explicit text and html documentation parameters; this is to support chained construction for the needs of testing where we don't have UML objects but want to print descriptions.

Method Summary

boolean	addEntry(EntryDoc entryDoc) Returns whether entryDoc has been successfully added.
AbstractPropertiesDoc.CellText	deduceCellText(java.lang.String prefix, TextDescription raw, TextDescription html, UmlObject o)
AbstractPropertiesDoc.CellText	deduceCellText(java.lang.String prefix, UmlObject o) Selects the description to retain, according to configuration.
void	filterAssociationEnds(UmlClass c, java.util.Collection retainedNatives, java.util.Collection retainedInheriteds)
void	filterAttributes(UmlClass c, java.util.Collection retainedNatives, java.util.Collection retainedInheriteds)
void	filterOperations(UmlClass c, java.util.Collection retainedNatives, java.util.Collection retainedInheriteds)
java.lang.String[]	getBookmarkIDs()
java.lang.String	getCaptionText()
java.lang.String[][]	getCellValues()
int	getColumnCount()
java.util.List	getDataEntryDocs()
static java.lang.String	getDeprecatedTextAsPrefix(UmlObject o) Returns empty string if property is not deprecated, otherwise formatted text ending with white space.
java.util.List	getEntryDocs()
TextDescription.TextKind[]	getFormats()
int	getHeadingEntriesCount()
static java.lang.String	getInitValAsPrefix(UmlAttribute attr) Returns the formatted initial value detail (const/default/range) to be used as prefix.
static java.lang.String	getInitValAsSuffix(UmlAttribute attr) Returns the formatted initial value detail (const/initial/range) to be used as suffix.

java.lang.String	getIntroText()
int	getRowCount()
EntryDoc.Kind[]	getRowKinds()
java.lang.String	getTableName()
TableSpec	getTableSpec()
void	initRawData(EntryDoc entry, UmlObject property) Initialises raw data for a native property (tag, name, alias+ID, desc+ID, informative, deprecated, inheritedFrom).
void	initRawData(EntryDoc entry, UmlObject property, java.lang.String baseTypeName) Initialises raw data for an inherited property (tag, name, alias+ID, desc+ID, informative, deprecated, inheritedFrom).
boolean	notEmpty()
boolean	toSkipInherited(UmlObject o) Returns whether inherited object o needs to be skipped, according to configuration.
java.lang.String	toString()

Methods inherited from class org.tanjakostic.jcleancim.docgen.collector.impl.AbstractObjectDoc

copyCell, copyNonEmptyCell, createDocId, deduceBookmark, deduceQualifiersPrefix, deduceQualifiersPrefix, deduceQualifiersPrefix, deduceQualifiersPrefix, filterClasses, getBmRegistry, getBookmarkID, getCell, getCells, getDescription, getDocgenCfg, getHeadingText, hasKey, log, prepareForHyperlink, prepareForHyperlinkAdjustedName, putCell, putCellNonEmpty, toSkip, toString, useHtml

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.docgen.collector.ObjectDoc

getBmRegistry, getBookmarkID, getDescription, getDocgenCfg, getHeadingText

Methods inherited from interface org.tanjakostic.jcleancim.docgen.collector.RawData

copyCell, copyNonEmptyCell, getCell, getCells, hasKey, putCell, putCellNonEmpty

Methods inherited from interface org.tanjakostic.jcleancim.docgen.collector.PropertiesDoc

getBookmarkIDs, getCaptionText, getCellValues, getColumnCount, getDataEntryDocs, getEntryDocs, getFormats, getHeadingEntriesCount, getIntroText, getRowCount, getRowKinds, getTableName, getTableSpec, notEmpty

Methods inherited from interface org.tanjakostic.jcleancim.docgen.collector.ObjectDoc

getBmRegistry, getBookmarkID, getDescription, getDocgenCfg, getHeadingText

Methods inherited from interface org.tanjakostic.jcleancim.docgen.collector.RawData
copyCell, copyNonEmptyCell, getCell, getCells, hasKey, putCell, putCellNonEmpty

Constructors

AbstractPropertiesDoc

```
protected AbstractPropertiesDoc(DocgenConfig docgenCfg,
                                UmlObject object,
                                java.lang.String what,
                                java.lang.String introText,
                                java.lang.String captionText,
                                TableSpec colSpec,
                                java.lang.String tableName,
                                BookmarkRegistry bmRegistry)
```

Creates an instance with a `EntryDoc.Kind.tableName` entry (if `tableName` is not null) and `EntryDoc.Kind.columnLabels` entry (from `colSpec`) in the list of entries; use when you want to add `EntryDoc.Kind.data` (and optionally, `EntryDoc.Kind.groupSubhead`) entries one by one, after some complex processing.

Parameters:

`docgenCfg`
`object`
`what`
`introText`
`captionText`
`colSpec`
`tableName`
`bmRegistry`

AbstractPropertiesDoc

```
protected AbstractPropertiesDoc(DocgenConfig docgenCfg,
                                UmlObject object,
                                java.lang.String what,
                                TextDescription description,
                                TextDescription htmlDescription,
                                boolean ignoreDesc,
                                java.lang.String headingText,
                                java.lang.String introText,
                                java.lang.String captionText,
                                TableSpec colSpec,
                                java.lang.String tableName,
                                BookmarkRegistry bmRegistry)
```

Same as `AbstractPropertiesDoc(DocgenConfig, UmlObject, String, String, String, TableSpec, String, BookmarkRegistry)` , but with explicit text and html documentation parameters; this is to support chained construction for the needs of testing where we don't have UML objects but want to print descriptions.

Parameters:

`docgenCfg`
`object`
`what`
`description`
`htmlDescription`

(continued from last page)

ignoreDesc
headingText
introText
captionText
colSpec
tableName
bmRegistry

Methods

initRawData

```
protected void initRawData(EntryDoc entry,
         UmlObject property)
```

Initialises raw data for a native property (tag, name, alias+ID, desc+ID, informative, deprecated, inheritedFrom).

Parameters:
entry
property

initRawData

```
protected void initRawData(EntryDoc entry,
         UmlObject property,
         java.lang.String baseTypeName)
```

Initialises raw data for an inherited property (tag, name, alias+ID, desc+ID, informative, deprecated, inheritedFrom).

Parameters:
entry
property
baseTypeName - name of base type from which this property gets inherited.

deduceCellText

```
protected final AbstractPropertiesDoc.CellText deduceCellText(java.lang.String prefix,
         UmlObject o)
```

Selects the description to retain, according to configuration.

deduceCellText

```
protected final AbstractPropertiesDoc.CellText deduceCellText(java.lang.String prefix,
         TextDescription raw,
         TextDescription html,
         UmlObject o)
```

addEntry

```
protected final boolean addEntry(EntryDoc entryDoc)
```

Returns whether entryDoc has been successfully added.

(continued from last page)

filterAttributes

```
protected final void filterAttributes(UmlClass c,
         java.util.Collection retainedNatives,
         java.util.Collection retainedInheriteds)
```

filterAssociationEnds

```
protected final void filterAssociationEnds(UmlClass c,
         java.util.Collection retainedNatives,
         java.util.Collection retainedInheriteds)
```

filterOperations

```
protected final void filterOperations(UmlClass c,
         java.util.Collection retainedNatives,
         java.util.Collection retainedInheriteds)
```

toSkipInherited

```
protected final boolean toSkipInherited(UmlObject o)
```

Returns whether inherited object o needs to be skipped, according to configuration.

toString

```
public java.lang.String toString()
```

getDeprecatedTextAsPrefix

```
protected final static java.lang.String getDeprecatedTextAsPrefix(UmlObject o)
```

Returns empty string if property is not deprecated, otherwise formatted text ending with white space.

getInitValAsSuffix

```
protected static java.lang.String getInitValAsSuffix(UmlAttribute attr)
```

Returns the formatted initial value detail (const/initial/range) to be used as suffix.

getInitValAsPrefix

```
protected final static java.lang.String getInitValAsPrefix(UmlAttribute attr)
```

Returns the formatted initial value detail (const/default/range) to be used as prefix.

notEmpty

```
public final boolean notEmpty()
```

getIntroText

public final java.lang.String **getIntroText**()

getCaptionText

public final java.lang.String **getCaptionText**()

getHeadingEntriesCount

public final int **getHeadingEntriesCount**()

getEntryDocs

public final java.util.List **getEntryDocs**()

getDataEntryDocs

public java.util.List **getDataEntryDocs**()

getTableName

public final java.lang.String **getTableName**()

getTableSpec

public final TableSpec **getTableSpec**()

getRowCount

public final int **getRowCount**()

getColumnCount

public final int **getColumnCount**()

getCellValues

public final java.lang.String[][] **getCellValues**()

(continued from last page)

getRowKinds

```
public final EntryDoc.Kind[] getRowKinds()
```

getFormats

```
public TextDescription.TextKind[] getFormats()
```

getBookmarkIDs

```
public java.lang.String[] getBookmarkIDs()
```

org.tanjakostic.jcleancim.docgen.collector.impl
Class AbstractPropertiesDoc.CellText

```
java.lang.Object
  |-org.tanjakostic.jcleancim.model.TextDescription
    |-org.tanjakostic.jcleancim.docgen.collector.impl.AbstractPropertiesDoc.CellText
```

protected static class **AbstractPropertiesDoc.CellText**
extends TextDescription

Field Summary

public final	formatInfo

Fields inherited from class org.tanjakostic.jcleancim.model.TextDescription

DEFAULT_KIND, DEFAULT_TEXT, EMPTY_HTML, EMPTY_TXT, kind, text

Constructor Summary

protected	CellText(TextDescription text) Creates an instance with null formatting info (all formatting will be ignored).
protected	CellText(java.lang.Integer fmtIdx, TextDescription textDesc) Constructor.

Methods inherited from class org.tanjakostic.jcleancim.model.TextDescription

appendParagraph, isEmpty, prepend, prepend, toString

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Fields

formatInfo

public final org.tanjakostic.jcleancim.docgen.collector.FormatInfo **formatInfo**

Constructors

CellText

protected **CellText**(TextDescription text)

(continued from last page)

Creates an instance with null formatting info (all formatting will be ignored).

CellText

```
protected CellText(java.lang.Integer fmtIdx,
                   TextDescription textDesc)
```

Constructor.

org.tanjakostic.jcleancim.docgen.collector.impl
Class Attributes61850Doc

```
java.lang.Object
  +-org.tanjakostic.jcleancim.docgen.collector.impl.AbstractObjectDoc
    +-org.tanjakostic.jcleancim.docgen.collector.impl.AbstractPropertiesDoc
      +-org.tanjakostic.jcleancim.docgen.collector.impl.Attributes61850Doc
```

All Implemented Interfaces:
ObjectDoc, PropertiesDoc

---

public abstract class **Attributes61850Doc**
extends AbstractPropertiesDoc

Holds the utility method to filter groups which may stay with no members after applying configuration filters (e.g., a class actually inherits from another class, but another class is informative, and we don't want informative stuff printed), and some common formatting.

Field Summary

public static final	CAPTION_FMT Value: **Attributes of %s**
public static final	INTRO_FMT Value: **shows all attributes of %s.**
protected static final	TRANSIENT String to print in 61860-7-4 when data object is transient. Value: **T**

Fields inherited from interface org.tanjakostic.jcleancim.docgen.collector.ObjectDoc
DEFAULT_PREFIX_FMT

Fields inherited from interface org.tanjakostic.jcleancim.docgen.collector.PropertiesDoc
INHERITED_FROM

Fields inherited from interface org.tanjakostic.jcleancim.docgen.collector.ObjectDoc
DEFAULT_PREFIX_FMT

Constructor Summary

protected	Attributes61850Doc(DocgenConfig docgenCfg, UmlObject object, java.lang.String what, java.lang.String headingText, java.lang.String introText, java.lang.String captionText, TableSpec colSpec, java.lang.String tableName, BookmarkRegistry bmRegistry) Constructor.

Method Summary

void	deduceTypeText(UmlClass mmType, RawData outRawData, boolean all) Fills outRawData with value for the key org.tanjakostic.jcleancim.docgen.collector.WAX#A_deducedTypeText and, if enabled (all=true) and where they exist, values for keys org.tanjakostic.jcleancim.docgen.collector.WAX#A_type and org.tanjakostic.jcleancim.docgen.collector.WAX#A_typeKind.
void	deduceTypeTextForDataIndex(UmlClass mmType, RawData outRawData) Similar to deduceTypeText(UmlClass, RawData, boolean, boolean), with third argument set to false, and taking care of the transient CDC attributes when writing data index for LNs.
static void	fillPresenceConditionAndArgs(RawData entry, PresenceCondition pc, boolean isInherited, java.lang.String context) Fills appropriately raw data WAX.A_presCond, WAX.A_presCondArgs, WAX.A_presCondArgsID and WAX.A_cond.
static void	fillPresenceConditionAndArgs(RawData entry, PresenceCondition pc, boolean isInherited, java.lang.String context, boolean isDerivedStats) Fills appropriately raw data WAX.A_presCond, WAX.A_presCondArgs, WAX.A_presCondArgsID and WAX.A_cond if isDerivedStats=false, otherwise WAX.A_dsPresCond, WAX.A_dsPresCondArgs, WAX.A_dsPresCondArgsID and WAX.A_dsCond.
java.util.Collection	filterGroups(java.util.Collection groups) Returns non-empty groups, retained from groups after applying the filters set in configuration.
java.lang.String	prepareForHyperlink(PresenceCondition pc)

Methods inherited from class org.tanjakostic.jcleancim.docgen.collector.impl.AbstractPropertiesDoc

addEntry, deduceCellText, deduceCellText, filterAssociationEnds, filterAttributes, filterOperations, getBookmarkIDs, getCaptionText, getCellValues, getColumnCount, getDataEntryDocs, getDeprecatedTextAsPrefix, getEntryDocs, getFormats, getHeadingEntriesCount, getInitValAsPrefix, getInitValAsSuffix, getIntroText, getRowCount, getRowKinds, getTableName, getTableSpec, initRawData, initRawData, notEmpty, toSkipInherited, toString

Methods inherited from class org.tanjakostic.jcleancim.docgen.collector.impl.AbstractObjectDoc

copyCell, copyNonEmptyCell, createDocId, deduceBookmark, deduceQualifiersPrefix, deduceQualifiersPrefix, deduceQualifiersPrefix, deduceQualifiersPrefix, filterClasses, getBmRegistry, getBookmarkID, getCell, getCells, getDescription, getDocgenCfg, getHeadingText, hasKey, log, prepareForHyperlink, prepareForHyperlinkAdjustedName, putCell, putCellNonEmpty, toSkip, toString, useHtml

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.docgen.collector.ObjectDoc

getBmRegistry, getBookmarkID, getDescription, getDocgenCfg, getHeadingText

Methods inherited from interface org.tanjakostic.jcleancim.docgen.collector.RawData

copyCell, copyNonEmptyCell, getCell, getCells, hasKey, putCell, putCellNonEmpty

Methods inherited from interface org.tanjakostic.jcleancim.docgen.collector.PropertiesDoc
getBookmarkIDs, getCaptionText, getCellValues, getColumnCount, getDataEntryDocs, getEntryDocs, getFormats, getHeadingEntriesCount, getIntroText, getRowCount, getRowKinds, getTableName, getTableSpec, notEmpty

Methods inherited from interface org.tanjakostic.jcleancim.docgen.collector.ObjectDoc
getBmRegistry, getBookmarkID, getDescription, getDocgenCfg, getHeadingText

Methods inherited from interface org.tanjakostic.jcleancim.docgen.collector.RawData
copyCell, copyNonEmptyCell, getCell, getCells, hasKey, putCell, putCellNonEmpty

Fields

TRANSIENT

```
protected static final java.lang.String TRANSIENT
```

String to print in 61860-7-4 when data object is transient.
Constant value: **T**

INTRO_FMT

```
public static final java.lang.String INTRO_FMT
```

Constant value: **shows all attributes of %s.**

CAPTION_FMT

```
public static final java.lang.String CAPTION_FMT
```

Constant value: **Attributes of %s**

Constructors

Attributes61850Doc

```
protected Attributes61850Doc(DocgenConfig docgenCfg,
                             UmlObject object,
                             java.lang.String what,
                             java.lang.String headingText,
                             java.lang.String introText,
                             java.lang.String captionText,
                             TableSpec colSpec,
                             java.lang.String tableName,
                             BookmarkRegistry bmRegistry)
```

Constructor.

Methods

(continued from last page)

filterGroups

```
protected final java.util.Collection filterGroups(java.util.Collection groups)
```

Returns non-empty groups, retained from `groups` after applying the filters set in configuration.

deduceTypeText

```
protected final void deduceTypeText(UmlClass mmType,
          RawData outRawData,
          boolean all)
```

Fills `outRawData` with value for the key org.tanjakostic.jcleancim.docgen.collector.WAX#A_deducedTypeText and, if enabled (`all=true`) and where they exist, values for keys org.tanjakostic.jcleancim.docgen.collector.WAX#A_type and org.tanjakostic.jcleancim.docgen.collector.WAX#A_typeKind.

Parameters:

`mmType`

`outRawData` - in/out argument, filled here.

`all` - if true, will set all potential fields; otherwise, will set only org.tanjakostic.jcleancim.docgen.collector.WAX#A_deducedTypeText.

deduceTypeTextForDataIndex

```
protected void deduceTypeTextForDataIndex(UmlClass mmType,
          RawData outRawData)
```

Similar to `deduceTypeText(UmlClass, RawData, boolean, boolean)`, with third argument set to `false`, and taking care of the transient CDC attributes when writing data index for LNs.

prepareForHyperlink

```
protected java.lang.String prepareForHyperlink(PresenceCondition pc)
```

fillPresenceConditionAndArgs

```
protected static void fillPresenceConditionAndArgs(RawData entry,
          PresenceCondition pc,
          boolean isInherited,
          java.lang.String context)
```

Fills appropriately raw data `WAX.A_presCond`, `WAX.A_presCondArgs`, `WAX.A_presCondArgsID` and `WAX.A_cond`.

fillPresenceConditionAndArgs

```
protected static void fillPresenceConditionAndArgs(RawData entry,
          PresenceCondition pc,
          boolean isInherited,
          java.lang.String context,
          boolean isDerivedStats)
```

Fills appropriately raw data `WAX.A_presCond`, `WAX.A_presCondArgs`, `WAX.A_presCondArgsID` and `WAX.A_cond` if `isDerivedStats=false`, otherwise `WAX.A_dsPresCond`, `WAX.A_dsPresCondArgs`, `WAX.A_dsPresCondArgsID` and `WAX.A_dsCond`.

org.tanjakostic.jcleancim.docgen.collector.impl
Class DocCollectorImpl

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.docgen.collector.impl.DocCollectorImpl
```

All Implemented Interfaces:
DocCollector

public class **DocCollectorImpl**
extends java.lang.Object
implements DocCollector

Default implementation of `DocCollector`,.

Constructor Summary

public	`DocCollectorImpl(UmlModel model)` Constructs the collector from the UML model.
public	`DocCollectorImpl(Config cfg, ModelFinder modelFinder)` Constructs the instance to manually (through API) add documentation for package and other UML objects.

Method Summary

boolean	`addSkippedInformativePackage(java.lang.String qName)`
void	`addToFlattened(ClassDoc classDoc)`
void	`addToFlattened(PackageDoc packageDoc)`
void	`addToScoped(PackageDoc packageDoc)`
void	`collect(UmlModel model)`
BookmarkRegistry	`getBmRegistry()`
DocgenConfig	`getDocgenCfg()`
FixedFormDocumentation	`getFixedFormDocumentation()`
FreeFormDocumentation	`getFreeFormDocumentation()`
boolean	`isFromUml()`
java.lang.String	`toString()`

Methods inherited from class `java.lang.Object`

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.docgen.collector.DocCollector

addSkippedInformativePackage, addToFlattened, addToFlattened, addToScoped, collect, getBmRegistry, getDocgenCfg, getFixedFormDocumentation, getFreeFormDocumentation, isFromUml

Constructors

DocCollectorImpl

public **DocCollectorImpl**(UmlModel model)

Constructs the collector from the UML model. After construction, call collect(UmlModel) to obtain the input for document generation.

Parameters:
model

DocCollectorImpl

public **DocCollectorImpl**(Config cfg,
ModelFinder modelFinder)

Constructs the instance to manually (through API) add documentation for package and other UML objects.

Parameters:
cfg
modelFinder

Methods

collect

public void **collect**(UmlModel model)

getFreeFormDocumentation

public FreeFormDocumentation **getFreeFormDocumentation**()

getFixedFormDocumentation

public FixedFormDocumentation **getFixedFormDocumentation**()

addToFlattened

public void **addToFlattened**(PackageDoc packageDoc)

(continued from last page)

addToFlattened

```
public void addToFlattened(ClassDoc classDoc)
```

addToScoped

```
public void addToScoped(PackageDoc packageDoc)
```

addSkippedInformativePackage

```
public boolean addSkippedInformativePackage(java.lang.String qName)
```

getDocgenCfg

```
public DocgenConfig getDocgenCfg()
```

getBmRegistry

```
public BookmarkRegistry getBmRegistry()
```

isFromUml

```
public boolean isFromUml()
```

toString

```
public java.lang.String toString()
```

org.tanjakostic.jcleancim.docgen.collector.impl
Class EntryDocImpl

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.docgen.collector.impl.EntryDocImpl
```

All Implemented Interfaces:
EntryDoc

public class **EntryDocImpl**
extends java.lang.Object
implements EntryDoc

Single property entry (row with values).

Fields inherited from interface `org.tanjakostic.jcleancim.docgen.collector.EntryDoc`
`SEPARATOR`

Method Summary

`java.lang.String`	`copyCell(RawData src, java.lang.String key)`
`java.lang.String`	`copyNonEmptyCell(RawData src, java.lang.String key)`
`static EntryDoc`	`createColumnLabels(java.lang.String[] values)` Creates `EntryDoc.Kind.columnLabels` entry.
`static EntryDocImpl`	`createData(java.lang.String bookmarkID, FormatInfo formatInfo, java.lang.String[] values)` Creates regular entry with data.
`static EntryDoc`	`createGroupSubhead(AGSpec agSpec, int columnCount)` Creates `EntryDoc.Kind.groupSubhead` entry.
`static EntryDoc`	`createTableName(java.lang.String name, int columnCount)` Creates `EntryDoc.Kind.tableName` entry.
`static EntryDoc`	`createUnformattedData(java.lang.String bookmarkID, java.lang.String[] values)` Creates regular entry with data without any formatting (also, no new line character).
`AGSpec`	`getAttrGroupSpec()`
`java.lang.String`	`getBookmarkID()`
`java.lang.String`	`getCell(java.lang.String key)`
`java.util.Map`	`getCells()`
`FormatInfo`	`getFormatInfo()`

EntryDoc.Kind	getKind()
java.lang.String[]	getValues()
boolean	hasKey(java.lang.String key)
java.lang.String	putCell(java.lang.String key, java.lang.String value)
java.lang.String	putCellNonEmpty(java.lang.String key, java.lang.String value)
java.lang.String	toCsv()
java.lang.String	toString()

Methods inherited from class java.lang.Object
clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.docgen.collector.EntryDoc
getAttrGroupSpec, getBookmarkID, getFormatInfo, getKind, getValues, toCsv

Methods inherited from interface org.tanjakostic.jcleancim.docgen.collector.RawData
copyCell, copyNonEmptyCell, getCell, getCells, hasKey, putCell, putCellNonEmpty

Methods

createTableName

```
public static EntryDoc createTableName(java.lang.String name,
          int columnCount)
  throws java.lang.IllegalArgumentException
```

Creates EntryDoc.Kind.tableName entry.

Parameters:

name - non-null, (trimmed) non-empty name for the table.

columnCount - (positive) number of columns in the table; used to fill values with empty content except for the first item.

Throws:

IllegalArgumentException - if name is empty, or if columnCount is not positive.

createColumnLabels

```
public static EntryDoc createColumnLabels(java.lang.String[] values)
  throws java.lang.IllegalArgumentException
```

Creates EntryDoc.Kind.columnLabels entry.

Parameters:

values - non-empty array of non-null values.

(continued from last page)

Throws:
`IllegalArgumentException` - if `values` is empty.

createGroupSubhead

```
public static EntryDoc createGroupSubhead(AGSpec agSpec,
          int columnCount)
  throws java.lang.IllegalArgumentException
```

Creates `EntryDoc.Kind.groupSubhead` entry.

Parameters:
`agSpec` - non-null spec for the group subhead.
`columnCount` - (positive) number of columns in the table; used to fill values with empty content except for the first item.

Throws:
`IllegalArgumentException` - if `name` is empty, or if `columnCount` is not positive.

createUnformattedData

```
public static EntryDoc createUnformattedData(java.lang.String bookmarkID,
          java.lang.String[] values)
  throws java.lang.IllegalArgumentException
```

Creates regular entry with data without any formatting (also, no new line character).

Parameters:
`bookmarkID` - (potentially null) bookmark ID.
`values` - non-empty array of non-null values.

Throws:
`IllegalArgumentException` - if `values` is empty.

createData

```
public static EntryDocImpl createData(java.lang.String bookmarkID,
          FormatInfo formatInfo,
          java.lang.String[] values)
  throws java.lang.IllegalArgumentException
```

Creates regular entry with data.

Parameters:
`bookmarkID` - (potentially null) bookmark ID.
`formatInfo` - when non-null, index of the `values` whose content needs to preserve formatting when printed (this corresponds to a formatted column, e.g. for description of items).
`values` - non-empty array of non-null values.

Throws:
`IllegalArgumentException` - if `values` is empty.

toString

```
public java.lang.String toString()
```

(continued from last page)

getValues

```
public final java.lang.String[] getValues()
```

getKind

```
public EntryDoc.Kind getKind()
```

getAttrGroupSpec

```
public AGSpec getAttrGroupSpec()
```

getFormatInfo

```
public final FormatInfo getFormatInfo()
```

getBookmarkID

```
public java.lang.String getBookmarkID()
```

toCsv

```
public final java.lang.String toCsv()
```

putCell

```
public java.lang.String putCell(java.lang.String key,
         java.lang.String value)
```

copyCell

```
public java.lang.String copyCell(RawData src,
         java.lang.String key)
```

copyNonEmptyCell

```
public java.lang.String copyNonEmptyCell(RawData src,
         java.lang.String key)
```

(continued from last page)

putCellNonEmpty

```
public java.lang.String putCellNonEmpty(java.lang.String key,
         java.lang.String value)
```

hasKey

```
public boolean hasKey(java.lang.String key)
```

getCells

```
public java.util.Map getCells()
```

getCell

```
public java.lang.String getCell(java.lang.String key)
```

org.tanjakostic.jcleancim.docgen.collector.impl
Class FigureDocImpl

```
java.lang.Object
  |-org.tanjakostic.jcleancim.docgen.collector.impl.AbstractObjectDoc
    |-org.tanjakostic.jcleancim.docgen.collector.impl.FigureDocImpl
```

All Implemented Interfaces:
FigureDoc, ObjectDoc

public class **FigureDocImpl**
extends AbstractObjectDoc
implements ObjectDoc, FigureDoc

Data required for documentation of diagrams. For the layout, see FigureDoc.

Fields inherited from interface org.tanjakostic.jcleancim.docgen.collector.ObjectDoc
DEFAULT_PREFIX_FMT

Fields inherited from interface org.tanjakostic.jcleancim.docgen.collector.FigureDoc
CAPTION_TEXT_FORMAT, INTRO_TEXT_FORMAT

Fields inherited from interface org.tanjakostic.jcleancim.docgen.collector.ObjectDoc
DEFAULT_PREFIX_FMT

Constructor Summary

public	FigureDocImpl(DocgenConfig docgenCfg, UmlDiagram d, BookmarkRegistry bmRegistry) Constructor.
public	FigureDocImpl(DocgenConfig docgenCfg, UmlDiagram d, TextDescription description, TextDescription htmlDescription, java.io.File pic, java.lang.String showsWhat, java.lang.String caption, BookmarkRegistry bmRegistry) Constructor.

Method Summary

java.lang.String	getCaptionText()
java.io.File	getFigureFile()
java.lang.String	getIntroText()

Methods inherited from class org.tanjakostic.jcleancim.docgen.collector.impl.AbstractObjectDoc

copyCell, copyNonEmptyCell, createDocId, deduceBookmark, deduceQualifiersPrefix, deduceQualifiersPrefix, deduceQualifiersPrefix, deduceQualifiersPrefix, filterClasses, getBmRegistry, getBookmarkID, getCell, getCells, getDescription, getDocgenCfg, getHeadingText, hasKey, log, prepareForHyperlink, prepareForHyperlinkAdjustedName, putCell, putCellNonEmpty, toSkip, toString, useHtml

Methods inherited from class `java.lang.Object`

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface `org.tanjakostic.jcleancim.docgen.collector.ObjectDoc`

getBmRegistry, getBookmarkID, getDescription, getDocgenCfg, getHeadingText

Methods inherited from interface `org.tanjakostic.jcleancim.docgen.collector.RawData`

copyCell, copyNonEmptyCell, getCell, getCells, hasKey, putCell, putCellNonEmpty

Methods inherited from interface `org.tanjakostic.jcleancim.docgen.collector.FigureDoc`

getCaptionText, getFigureFile, getIntroText

Methods inherited from interface `org.tanjakostic.jcleancim.docgen.collector.ObjectDoc`

getBmRegistry, getBookmarkID, getDescription, getDocgenCfg, getHeadingText

Methods inherited from interface `org.tanjakostic.jcleancim.docgen.collector.RawData`

copyCell, copyNonEmptyCell, getCell, getCells, hasKey, putCell, putCellNonEmpty

Constructors

FigureDocImpl

```
public FigureDocImpl(DocgenConfig docgenCfg,
                     UmlDiagram d,
                     BookmarkRegistry bmRegistry)
```

Constructor.

Parameters:

docgenCfg
d
bmRegistry

FigureDocImpl

```
public FigureDocImpl(DocgenConfig docgenCfg,
                     UmlDiagram d,
                     TextDescription description,
                     TextDescription htmlDescription,
                     java.io.File pic,
                     java.lang.String showsWhat,
                     java.lang.String caption,
                     BookmarkRegistry bmRegistry)
```

(continued from last page)

Constructor.

Parameters:

`docgenCfg` - non-null docgen configuration.
`d` - (potentially null) UML diagram.
`description`
`htmlDescription`
`pic` - file containing picture.
`showsWhat` - non-null text to enclose in org.tanjakostic.jcleancim.docgen.collector.FigureDoc#INTRO_TEXT_FORMAT.
`caption` - non-null caption text.
`bmRegistry`

Methods

getIntroText

`public java.lang.String` **`getIntroText`**`()`

---

getCaptionText

`public java.lang.String` **`getCaptionText`**`()`

---

getFigureFile

`public java.io.File` **`getFigureFile`**`()`

org.tanjakostic.jcleancim.docgen.collector.impl
Class ModelFinderImpl

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.docgen.collector.impl.ModelFinderImpl
```

All Implemented Interfaces:
ModelFinder

public class **ModelFinderImpl**
extends java.lang.Object
implements ModelFinder

Provides methods for a writer to retreive UML model elements that are referenced by placeholders in the input template document.

This implementation of `ModelFinder` relies on the full in-memory model, and is convenient for document generation.

Constructor Summary

public	`ModelFinderImpl(UmlModel model)` Constructor.

Method Summary

java.lang.String	`findAttributeValue(java.lang.String className, java.lang.String attributeName)`
java.lang.String	`findClassName(java.lang.String packageName, java.lang.String className)`
java.io.File	`findDiagramFile(java.lang.String containerName, java.lang.String diagramName)`
TextDescription	`findDiagramNote(java.lang.String containerName, java.lang.String diagramName)`
java.lang.String	`findIec61850NsName(java.lang.String className)`

Methods inherited from class `java.lang.Object`

`clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait`

Methods inherited from interface `org.tanjakostic.jcleancim.docgen.collector.ModelFinder`

`findAttributeValue, findClassName, findDiagramFile, findDiagramNote, findIec61850NsName`

Constructors

(continued from last page)

ModelFinderImpl

```
public ModelFinderImpl(UmlModel model)
```

Constructor.

Methods

findAttributeValue

```
public java.lang.String findAttributeValue(java.lang.String className,
         java.lang.String attributeName)
```

findDiagramFile

```
public java.io.File findDiagramFile(java.lang.String containerName,
         java.lang.String diagramName)
```

findDiagramNote

```
public TextDescription findDiagramNote(java.lang.String containerName,
         java.lang.String diagramName)
```

findClassName

```
public java.lang.String findClassName(java.lang.String packageName,
         java.lang.String className)
```

findIec61850NsName

```
public java.lang.String findIec61850NsName(java.lang.String className)
```

Package

org.tanjakostic.jcleancim.docgen.collector.impl.ag

org.tanjakostic.jcleancim.docgen.collector.impl.ag
Class AttributeGroup

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.docgen.collector.impl.ag.AttributeGroup
```

public class **AttributeGroup**
extends java.lang.Object

Helper class, used to group attributes of a class. It is handy for doc generation of fancy IEC61850 tables for LNs and CDCs.

The group lists first native then inherited attributes, and may have a name. In case of a CIM class, it will return a single attribute group with null name, while the class representing IEC61850 LN or CDC will return multiple attribute groups, in the order suitable for creating the doc.

FIXME: consolidate with AGSpec !

Constructor Summary

public	AttributeGroup(java.util.Collection nativeAttributes, java.util.Collection inheritedAttributes)
public	AttributeGroup(AGSpec agSpec, java.util.Collection nativeAttributes, java.util.Collection inheritedAttributes)

Method Summary

AGSpec	getAgSpec()
java.util.Collection	getInheritedAttributes()
java.util.Collection	getNativeAttributes()
static java.util.Collection	initCdcGroups(UmlClass c)
static java.util.Collection	initDaGroups(UmlClass c)
static java.util.Collection	initLnGroups(UmlClass c)

Methods inherited from class `java.lang.Object`

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Constructors

(continued from last page)

AttributeGroup

```
public AttributeGroup(java.util.Collection nativeAttributes,
                      java.util.Collection inheritedAttributes)
```

AttributeGroup

```
public AttributeGroup(AGSpec agSpec,
                      java.util.Collection nativeAttributes,
                      java.util.Collection inheritedAttributes)
```

Methods

initDaGroups

```
public static java.util.Collection initDaGroups(UmlClass c)
```

initCdcGroups

```
public static java.util.Collection initCdcGroups(UmlClass c)
```

initLnGroups

```
public static java.util.Collection initLnGroups(UmlClass c)
```

getAgSpec

```
public AGSpec getAgSpec()
```

getNativeAttributes

```
public java.util.Collection getNativeAttributes()
```

getInheritedAttributes

```
public java.util.Collection getInheritedAttributes()
```

Package
org.tanjakostic.jcleancim.docgen.writer

Classes and interfaces responsible for document generation out of the UML model.

Interfaces are currently implemented for MS Word document generation only; we are working on implementing serialisation in an XML format for documentation. Implementations are residing in subpackages `word` and `xml`.

Main classes and interfaces are:

- `Writer` - interface to implement in order to write the collected documentation.
- `WriterInput` - arguments to initialise any writer.
- `Placeholder` - contains points in template document that need to be replaced with the actual documentation content, or errors for invalid formats or inexistant model elements.

TODO:

- Add warning for those placeholders that should be specified in a heading in case they are found in text.

org.tanjakostic.jcleancim.docgen.writer
Class AbstractRange

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.docgen.writer.AbstractRange
```

All Implemented Interfaces:
Range

---

public abstract class **AbstractRange**
extends java.lang.Object
implements Range

Common implementation.
Parameters:
O - technology-specific type to access range object.

Constructor Summary

public	AbstractRange()

Method Summary

void	setStartEnd(int idxStart, int idxEnd)
java.lang.String	toString()

Methods inherited from class `java.lang.Object`

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface `org.tanjakostic.jcleancim.docgen.writer.Range`

getEnd, getObject, getStart, getText, setEnd, setStart, setStartEnd, setText

---

Constructors

AbstractRange

public **AbstractRange**()

Methods

(continued from last page)

setStartEnd

```
public final void setStartEnd(int idxStart,
         int idxEnd)
```

toString

```
public java.lang.String toString()
```

org.tanjakostic.jcleancim.docgen.writer
Class AbstractWriter

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.docgen.writer.AbstractWriter
```

All Implemented Interfaces:
Writer

Direct Known Subclasses:
WAXWriter, AbstractWordWriter

---

public abstract class **AbstractWriter**
extends java.lang.Object
implements Writer

Common implementation for all writers.

Fields inherited from interface `org.tanjakostic.jcleancim.docgen.writer.Writer`
`TOOL_CUSTOM_DOC_PROP`, `UML_CUSTOM_DOC_PROP`

Constructor Summary

protected	`AbstractWriter(WriterInput input)` Constructor.

Method Summary

java.util.Map	`getDocumentMetadata()`

Methods inherited from class `java.lang.Object`
`clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait`

Methods inherited from interface `org.tanjakostic.jcleancim.docgen.writer.Writer`
`getDocumentMetadata, getInput, getInputFileNames, getOutputFileNames, getSupportedFormats, write`

Constructors

AbstractWriter

protected **AbstractWriter**(WriterInput input)

Constructor.

Methods

(continued from last page)

getDocumentMetadata

```
public final java.util.Map getDocumentMetadata()
```

org.tanjakostic.jcleancim.docgen.writer
Class Caption

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.docgen.writer.Caption
```

public class **Caption**
extends java.lang.Object

Caption for figure or table.
Parameters:
`O` - technology-specific type to access range object.

Constructor Summary

public	`Caption(CaptionKind kind, Range range)` Constructor.

Method Summary

`CaptionKind`	`getKind()`
`Range`	`getRange()`
`Style`	`getStyle()`
`java.lang.String`	`toString()`

Methods inherited from class `java.lang.Object`

`clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait`

Constructors

Caption

```
public Caption(CaptionKind kind,
               Range range)
```

Constructor.

Parameters:
`kind`

Methods

(continued from last page)

getKind

```
public final CaptionKind getKind()
```

getRange

```
public Range getRange()
```

getStyle

```
public Style getStyle()
```

toString

```
public java.lang.String toString()
```

org.tanjakostic.jcleancim.docgen.writer
Class CaptionKind

```
java.lang.Object
  +-java.lang.Enum
    +-org.tanjakostic.jcleancim.docgen.writer.CaptionKind
```

All Implemented Interfaces:
java.io.Serializable, java.lang.Comparable

public final class **CaptionKind**
extends java.lang.Enum

We support 2 kinds of caption labels: for figures and for tables (and ignore those for equations, as we don't print any numbered equations).

Field Summary

public static final	Figure
public static final	Table

Method Summary

java.lang.String	getLabel()
Style	getStyle()
boolean	looksLikeCaption(java.lang.String styleName, java.lang.String text)
static CaptionKind	valueOf(java.lang.String name)
static CaptionKind[]	values()

Methods inherited from class `java.lang.Enum`

clone, compareTo, equals, finalize, getDeclaringClass, hashCode, name, ordinal, toString, valueOf

Methods inherited from class `java.lang.Object`

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface `java.lang.Comparable`

compareTo

(continued from last page)

Fields

Figure

```
public static final org.tanjakostic.jcleancim.docgen.writer.CaptionKind Figure
```

Table

```
public static final org.tanjakostic.jcleancim.docgen.writer.CaptionKind Table
```

Methods

values

```
public static CaptionKind[] values()
```

valueOf

```
public static CaptionKind valueOf(java.lang.String name)
```

getStyle

```
public Style getStyle()
```

looksLikeCaption

```
public boolean looksLikeCaption(java.lang.String styleName,
         java.lang.String text)
```

getLabel

```
public java.lang.String getLabel()
```

org.tanjakostic.jcleancim.docgen.writer
Class Cursor

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.docgen.writer.Cursor
```

public class **Cursor**
extends java.lang.Object

Simple association of placeholder and range.
Parameters:
O - technology-specific type to access range object.

Constructor Summary

public	Cursor(Placeholder placeholder, Range range)

Method Summary

Placeholder	getPlaceholder()
Range	getRange()
java.lang.String	toString()

Methods inherited from class `java.lang.Object`

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Constructors

Cursor

```
public Cursor(Placeholder placeholder,
              Range range)
```

Methods

getPlaceholder

```
public final Placeholder getPlaceholder()
```

(continued from last page)

getRange

```
public Range getRange()
```

toString

```
public java.lang.String toString()
```

org.tanjakostic.jcleancim.docgen.writer
Class CursorList

```
java.lang.Object
  +-java.util.AbstractCollection
    +-java.util.AbstractList
      +-org.tanjakostic.jcleancim.docgen.writer.CursorList
```

All Implemented Interfaces:
java.util.Collection, java.util.List

---

public class **CursorList**
extends java.util.AbstractList

List of cursors.

Fields inherited from class `java.util.AbstractList`
`modCount`

Constructor Summary

public	CursorList()

Method Summary

boolean	add(Cursor cursor)
int	captionAdded(CaptionKind kind, Cursor cursor) Returns the number to be used in caption for the `cursor`; must be called.
void	clear()
Cursor	get(int index)
java.util.List	getReplacementFailures()
Cursor	set(int index, Cursor cursor)
int	size()
java.util.List	snapshotIndexes()
java.lang.String	toString()
CursorList	updateRanges(CursorList cursors, Range range)

Methods inherited from class `java.util.AbstractList`

add, add, addAll, clear, equals, get, hashCode, indexOf, iterator, lastIndexOf, listIterator, listIterator, remove, removeRange, set, subList

Methods inherited from class `java.util.AbstractCollection`

add, addAll, clear, contains, containsAll, isEmpty, iterator, remove, removeAll, retainAll, size, toArray, toArray, toString

Methods inherited from class `java.lang.Object`

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface `java.util.Collection`

add, addAll, clear, contains, containsAll, equals, hashCode, isEmpty, iterator, parallelStream, remove, removeAll, removeIf, retainAll, size, spliterator, stream, toArray, toArray

Methods inherited from interface `java.lang.Iterable`

forEach, iterator, spliterator

Methods inherited from interface `java.util.List`

add, add, addAll, addAll, clear, contains, containsAll, equals, get, hashCode, indexOf, isEmpty, iterator, lastIndexOf, listIterator, listIterator, remove, remove, removeAll, replaceAll, retainAll, set, size, sort, spliterator, subList, toArray, toArray

Methods inherited from interface `java.util.Collection`

add, addAll, clear, contains, containsAll, equals, hashCode, isEmpty, iterator, parallelStream, remove, removeAll, removeIf, retainAll, size, spliterator, stream, toArray, toArray

Methods inherited from interface `java.lang.Iterable`

forEach, iterator, spliterator

Constructors

CursorList

```
public CursorList()
```

Methods

updateRanges

```
public CursorList updateRanges(CursorList cursors,
        Range range)
```

(continued from last page)

snapshotIndexes

```
public java.util.List snapshotIndexes()
```

getReplacementFailures

```
public java.util.List getReplacementFailures()
```

captionAdded

```
public int captionAdded(CaptionKind kind,
        Cursor cursor)
```

Returns the number to be used in caption for the `cursor`; must be called.

size

```
public int size()
```

get

```
public Cursor get(int index)
```

add

```
public boolean add(Cursor cursor)
```

set

```
public Cursor set(int index,
        Cursor cursor)
```

clear

```
public void clear()
```

toString

```
public java.lang.String toString()
```

org.tanjakostic.jcleancim.docgen.writer
Class ExistingCaptionLabel

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.docgen.writer.ExistingCaptionLabel
```

public class **ExistingCaptionLabel**
extends java.lang.Object

Holds basic information describing caption labels, initialised from an open application / document.

Nested Class Summary

class	ExistingCaptionLabel.Kind ExistingCaptionLabel.Kind

Field Summary

public final	id String representation of ID describing type.
public final	isBuiltIn If true, this is a built-in caption label.
public final	kind
public final	name Label text.

Constructor Summary

public	ExistingCaptionLabel(java.lang.String name, java.lang.String id, boolean isBuiltIn, ExistingCaptionLabel.Kind kind) Constructor.

Method Summary

boolean	isUsableFor(CaptionKind cKind)
java.lang.String	toString()

Methods inherited from class `java.lang.Object`
clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Fields

(continued from last page)

name

```
public final java.lang.String name
```

Label text.

id

```
public final java.lang.String id
```

String representation of ID describing type. Could be also a number (like in e.g. .doc document); implementation has to cast to appropriate type used for ID.

isBuiltIn

```
public final boolean isBuiltIn
```

If true, this is a built-in caption label.

kind

```
public final org.tanjakostic.jcleancim.docgen.writer.ExistingCaptionLabel.Kind kind
```

Constructors

ExistingCaptionLabel

```
public ExistingCaptionLabel(java.lang.String name,
                            java.lang.String id,
                            boolean isBuiltIn,
                            ExistingCaptionLabel.Kind kind)
```

Constructor.

Parameters:

`name`

`id` - although String, could be also number (like in e.g. .docx document); implementation has to cast to appropriate type used for ID

`isBuiltIn`

`kind`

Methods

isUsableFor

```
public boolean isUsableFor(CaptionKind cKind)
```

toString

```
public java.lang.String toString()
```

org.tanjakostic.jcleancim.docgen.writer
Class ExistingCaptionLabel.Kind

```
java.lang.Object
  |-java.lang.Enum
    |-org.tanjakostic.jcleancim.docgen.writer.ExistingCaptionLabel.Kind
```

All Implemented Interfaces:
java.io.Serializable, java.lang.Comparable

---

public static final class **ExistingCaptionLabel.Kind**
extends java.lang.Enum

Field Summary

public static final	FIG
public static final	OTHER
public static final	TAB

Method Summary

static ExistingCaptionLabel.Kind	valueOf(java.lang.String name)
static ExistingCaptionLabel.Kind[]	values()

Methods inherited from class `java.lang.Enum`

clone, compareTo, equals, finalize, getDeclaringClass, hashCode, name, ordinal, toString, valueOf

Methods inherited from class `java.lang.Object`

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface `java.lang.Comparable`

compareTo

Fields

(continued from last page)

FIG

```
public static final org.tanjakostic.jcleancim.docgen.writer.ExistingCaptionLabel.Kind
FIG
```

TAB

```
public static final org.tanjakostic.jcleancim.docgen.writer.ExistingCaptionLabel.Kind
TAB
```

OTHER

```
public static final org.tanjakostic.jcleancim.docgen.writer.ExistingCaptionLabel.Kind
OTHER
```

Methods

values

```
public static ExistingCaptionLabel.Kind[] values()
```

valueOf

```
public static ExistingCaptionLabel.Kind valueOf(java.lang.String name)
```

org.tanjakostic.jcleancim.docgen.writer
Class ExistingStyle

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.docgen.writer.ExistingStyle
```

public class **ExistingStyle**
extends java.lang.Object

Holds basic information describing styles, initialised from an open application / document.

Implementation note: This is the "buffer" between implementation-specifics (such as Word styles) and the style enumeration convenient for fluent use in the code.

Nested Class Summary

class	ExistingStyle.Kind ExistingStyle.Kind

Field Summary

public final	id String representation of ID describing type.
public final	isBuiltIn If true, this is a built-in style.
public final	name Style name.
public final	outline Number between 1 and 9, for TOC and Heading styles.

Constructor Summary

public	ExistingStyle(java.lang.String name, java.lang.String id, boolean isBuiltIn, int outline, ExistingStyle.Kind kind) Constructor.

Method Summary

ExistingStyle.Kind	getKind()
boolean	isUsableFor(Style s) Returns true for exact style (e.g.
void	setCustomKindFrom(Style s)
void	setKind(ExistingStyle.Kind kind)
java.lang.String	toString()

Methods inherited from class `java.lang.Object`
`clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait`

Fields

name

`public final java.lang.String` **`name`**

Style name.

id

`public final java.lang.String` **`id`**

String representation of ID describing type. Could be also a number (like in e.g. .doc document); implementation has to cast to appropriate type used for ID.

isBuiltIn

`public final boolean` **`isBuiltIn`**

If true, this is a built-in style.

outline

`public final int` **`outline`**

Number between 1 and 9, for TOC and Heading styles.

Constructors

ExistingStyle

```
public ExistingStyle(java.lang.String name,
                     java.lang.String id,
                     boolean isBuiltIn,
                     int outline,
                     ExistingStyle.Kind kind)
```

Constructor.

Parameters:

`name`

`id` - although String, could be also number (like in e.g. .docx document); implementation has to cast to appropriate type used for ID

`isBuiltIn`

`outline` - outline of the numbered style; relevant for TOC and heading styles, ignored for others

`kind` - potentially null; you may want to set this later on, when you can determine the kind (such as for custom, i.e., non-built-in styles)

Methods

(continued from last page)

setCustomKindFrom

```
public void setCustomKindFrom(Style s)
```

isUsableFor

```
public boolean isUsableFor(Style s)
```

Returns true for exact style (e.g. para) and for default (e.g., norm).

getKind

```
public ExistingStyle.Kind getKind()
```

setKind

```
public void setKind(ExistingStyle.Kind kind)
```

toString

```
public java.lang.String toString()
```

org.tanjakostic.jcleancim.docgen.writer
Class ExistingStyle.Kind

```
java.lang.Object
  +-java.lang.Enum
    +-org.tanjakostic.jcleancim.docgen.writer.ExistingStyle.Kind
```

All Implemented Interfaces:
java.io.Serializable, java.lang.Comparable

---

public static final class **ExistingStyle.Kind**
extends java.lang.Enum

Field Summary

public static final	CAPT
public static final	FIG
public static final	FIGCAPT
public static final	HEAD
public static final	NORM
public static final	OTHER
public static final	PARA
public static final	TABCAPT
public static final	TABCELL
public static final	TABHEAD
public static final	TOC

Method Summary

static ExistingStyle.Kind	valueOf(java.lang.String name)
static ExistingStyle.Kind[]	values()

Methods inherited from class `java.lang.Enum`

clone, compareTo, equals, finalize, getDeclaringClass, hashCode, name, ordinal, toString, valueOf

Methods inherited from class `java.lang.Object`
clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface `java.lang.Comparable`
compareTo

Fields

PARA

public static final org.tanjakostic.jcleancim.docgen.writer.ExistingStyle.Kind **PARA**

FIG

public static final org.tanjakostic.jcleancim.docgen.writer.ExistingStyle.Kind **FIG**

TABHEAD

public static final org.tanjakostic.jcleancim.docgen.writer.ExistingStyle.Kind **TABHEAD**

TABCELL

public static final org.tanjakostic.jcleancim.docgen.writer.ExistingStyle.Kind **TABCELL**

FIGCAPT

public static final org.tanjakostic.jcleancim.docgen.writer.ExistingStyle.Kind **FIGCAPT**

TABCAPT

public static final org.tanjakostic.jcleancim.docgen.writer.ExistingStyle.Kind **TABCAPT**

HEAD

public static final org.tanjakostic.jcleancim.docgen.writer.ExistingStyle.Kind **HEAD**

TOC

```
public static final org.tanjakostic.jcleancim.docgen.writer.ExistingStyle.Kind TOC
```

NORM

```
public static final org.tanjakostic.jcleancim.docgen.writer.ExistingStyle.Kind NORM
```

CAPT

```
public static final org.tanjakostic.jcleancim.docgen.writer.ExistingStyle.Kind CAPT
```

OTHER

```
public static final org.tanjakostic.jcleancim.docgen.writer.ExistingStyle.Kind OTHER
```

Methods

values

```
public static ExistingStyle.Kind[] values()
```

valueOf

```
public static ExistingStyle.Kind valueOf(java.lang.String name)
```

org.tanjakostic.jcleancim.docgen.writer
Class Placeholder

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.docgen.writer.Placeholder
```

public class **Placeholder**
extends java.lang.Object

When using `FreeFormDocumentation`, templates for doc generation have to use labels to indicate where to insert the documentation of what element of the UML model into the output document. The format and the read-only content of the placeholder is defined in `PlaceholderSpec`, while the placeholder itself is used for writing and actual replacing of the placeholder text.

Constructor Summary

public	`Placeholder(PlaceholderSpec phSpec)`
public	`Placeholder(PlaceholderSpec phSpec, int figureCountBefore, int tableCountBefore)`

Method Summary

int	`addFigure()` Returns the index to be used to reference the added figure caption.
int	`addTable()` Returns the index to be used to reference the added table caption.
int	`getFigureCount()` Returns the last figure caption index by the current end of the range.
int	`getFigureCountBefore()` Returns number of figures with caption before this placeholder.
`PlaceholderSpec`	`getPlaceholderSpec()` Returns the placeholder specification which holds text, kind etc.
`java.lang.String`	`getReplacedText()` Returns the replacement text, as set by the user with `setReplacedText(String)`.
int	`getTableCount()` Returns the last table caption index by the current end of the range.
int	`getTableCountBefore()` Returns number of tables with caption before this placeholder.
void	`incrementFigureBefore()` Increments the number of figures before this placeholder.
void	`incrementTableBefore()` Increments the number of tables before this placeholder.

void	setReplacedText(java.lang.String replacedText) Use this setter to track progress and status of placeholder replacement.
java.lang.String	toString()

Methods inherited from class java.lang.Object
clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Constructors

Placeholder

```
public Placeholder(PlaceholderSpec phSpec)
```

Placeholder

```
public Placeholder(PlaceholderSpec phSpec,
                   int figureCountBefore,
                   int tableCountBefore)
```

Methods

getPlaceholderSpec

```
public PlaceholderSpec getPlaceholderSpec()
```

Returns the placeholder specification which holds text, kind etc. independent of writer.

setReplacedText

```
public void setReplacedText(java.lang.String replacedText)
```

Use this setter to track progress and status of placeholder replacement.

getFigureCountBefore

```
public int getFigureCountBefore()
```

Returns number of figures with caption before this placeholder.

getTableCountBefore

```
public int getTableCountBefore()
```

Returns number of tables with caption before this placeholder.

getReplacedText

```
public java.lang.String getReplacedText()
```

(continued from last page)

Returns the replacement text, as set by the user with `setReplacedText(String)`. Initial value is null and is never changed by this class itself.

incrementFigureBefore

`public void` **`incrementFigureBefore`**`()`

Increments the number of figures before this placeholder.

incrementTableBefore

`public void` **`incrementTableBefore`**`()`

Increments the number of tables before this placeholder.

addFigure

`public int` **`addFigure`**`()`

Returns the index to be used to reference the added figure caption.

getFigureCount

`public int` **`getFigureCount`**`()`

Returns the last figure caption index by the current end of the range.

addTable

`public int` **`addTable`**`()`

Returns the index to be used to reference the added table caption.

getTableCount

`public int` **`getTableCount`**`()`

Returns the last table caption index by the current end of the range.

toString

`public java.lang.String` **`toString`**`()`

org.tanjakostic.jcleancim.docgen.writer
Interface Range

All Known Implementing Classes:
AbstractRange

public interface **Range**
extends

Technology-independent abstraction for range in documents.
Parameters:
`O` - technology-specific type to access range object.

Method Summary

abstract int	getEnd()
abstract java.lang.Object	getObject()
abstract int	getStart()
abstract java.lang.String	getText()
abstract void	setEnd(int idx)
abstract void	setStart(int idx)
abstract void	setStartEnd(int idxStart, int idxEnd)
abstract void	setText(java.lang.String newText)

Methods

getStart

```
public abstract int getStart()
```

setStart

```
public abstract void setStart(int idx)
```

getEnd

```
public abstract int getEnd()
```

(continued from last page)

setEnd

```
public abstract void setEnd(int idx)
```

setStartEnd

```
public abstract void setStartEnd(int idxStart,
         int idxEnd)
```

getText

```
public abstract java.lang.String getText()
```

setText

```
public abstract void setText(java.lang.String newText)
```

getObject

```
public abstract java.lang.Object getObject()
```

org.tanjakostic.jcleancim.docgen.writer
Class Style

```
java.lang.Object
  +-java.lang.Enum
    +-org.tanjakostic.jcleancim.docgen.writer.Style
```

All Implemented Interfaces:
java.io.Serializable, java.lang.Comparable

public final class **Style**
extends java.lang.Enum

Maps document in-built styles as replacement for our desired styles (in IEC template), to allow for doc generation even with a non-IEC template.

Field Summary

public static final	fig
public static final	figcapt
public static final	h1
public static final	h2
public static final	h3
public static final	h4
public static final	h5
public static final	h6
public static final	h7
public static final	h8
public static final	h9
public static	LLEVEL Detailed logging for Style initialisation.
public static final	para
public static final	tabcapt
public static final	tabcell

public static final	tabhead
public static final	toc1
public static final	toc2
public static final	toc3
public static final	toc4
public static final	toc5
public static final	toc6
public static final	toc7
public static final	toc8
public static final	toc9

Method Summary

static Style	getHeadingStyle(int outlineLevel)
java.lang.String	getName() Returns *the* style name to be used for this style.
java.util.List	getPreferredNames() Returns non-empty list of style names, configured by user for this style.
static Style	getTOCStyle(int outlineLevel)
java.util.Map	getUsableStyles() Returns all usable styles found in the open document.
java.util.Map	getUsableStyles(boolean inclBuiltIn, boolean inclCustom) Returns map of existing styles as found in an open document, which can be used for writing into document.
static void	initPreferred(java.util.List tocStylePrefixes, java.util.List headingStylePrefixes, java.util.List paraStyles, java.util.List figStyles, java.util.List tabheadStyles, java.util.List tabcellStyles, java.util.List figcaptStyles, java.util.List tabcaptStyles) Adds style names from user configuration in order of preferrence (first is best).
static void	initUsable(java.util.Map existingStyles) Initialises styles obtained from the document.
boolean	isHeading() Returns whether this is a heading style (one out of 9).
boolean	isRecognised(java.lang.String name) Returns whether name can be identified as a style name.

boolean	isTOC() Returns whether this is a TOC style (one out of 9).
static boolean	isTOC(java.lang.String styleName)
void	logAfter()
static void	reset() Resets all dynamically set values and collections; ensure to call as a part of clean-up process when existing the app.
java.lang.StringBuilder	toShortString()
java.lang.String	toString()
static Style	valueOf(java.lang.String name)
static Style[]	values()

Methods inherited from class java.lang.Enum
clone, compareTo, equals, finalize, getDeclaringClass, hashCode, name, ordinal, toString, valueOf

Methods inherited from class java.lang.Object
clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface java.lang.Comparable
compareTo

Fields

para

public static final org.tanjakostic.jcleancim.docgen.writer.Style **para**

fig

public static final org.tanjakostic.jcleancim.docgen.writer.Style **fig**

tabhead

public static final org.tanjakostic.jcleancim.docgen.writer.Style **tabhead**

tabcell

public static final org.tanjakostic.jcleancim.docgen.writer.Style **tabcell**

figcapt

public static final org.tanjakostic.jcleancim.docgen.writer.Style **figcapt**

tabcapt

public static final org.tanjakostic.jcleancim.docgen.writer.Style **tabcapt**

toc1

public static final org.tanjakostic.jcleancim.docgen.writer.Style **toc1**

toc2

public static final org.tanjakostic.jcleancim.docgen.writer.Style **toc2**

toc3

public static final org.tanjakostic.jcleancim.docgen.writer.Style **toc3**

toc4

public static final org.tanjakostic.jcleancim.docgen.writer.Style **toc4**

toc5

public static final org.tanjakostic.jcleancim.docgen.writer.Style **toc5**

toc6

public static final org.tanjakostic.jcleancim.docgen.writer.Style **toc6**

toc7

public static final org.tanjakostic.jcleancim.docgen.writer.Style **toc7**

(continued from last page)

toc8

```
public static final org.tanjakostic.jcleancim.docgen.writer.Style toc8
```

toc9

```
public static final org.tanjakostic.jcleancim.docgen.writer.Style toc9
```

h1

```
public static final org.tanjakostic.jcleancim.docgen.writer.Style h1
```

h2

```
public static final org.tanjakostic.jcleancim.docgen.writer.Style h2
```

h3

```
public static final org.tanjakostic.jcleancim.docgen.writer.Style h3
```

h4

```
public static final org.tanjakostic.jcleancim.docgen.writer.Style h4
```

h5

```
public static final org.tanjakostic.jcleancim.docgen.writer.Style h5
```

h6

```
public static final org.tanjakostic.jcleancim.docgen.writer.Style h6
```

h7

```
public static final org.tanjakostic.jcleancim.docgen.writer.Style h7
```

(continued from last page)

h8

```
public static final org.tanjakostic.jcleancim.docgen.writer.Style h8
```

h9

```
public static final org.tanjakostic.jcleancim.docgen.writer.Style h9
```

LLEVEL

```
public static org.apache.log4j.Level LLEVEL
```

Detailed logging for Style initialisation. Turn to INFO for debugging only.

Methods

values

```
public static Style[] values()
```

valueOf

```
public static Style valueOf(java.lang.String name)
```

getTOCStyle

```
public static Style getTOCStyle(int outlineLevel)
```

getHeadingStyle

```
public static Style getHeadingStyle(int outlineLevel)
```

initPreferred

```
public static void initPreferred(java.util.List tocStylePrefixes,
         java.util.List headingStylePrefixes,
         java.util.List paraStyles,
         java.util.List figStyles,
         java.util.List tabheadStyles,
         java.util.List tabcellStyles,
         java.util.List figcaptStyles,
         java.util.List tabcaptStyles)
```

Adds style names from user configuration in order of preferrence (first is best).

(continued from last page)

initUsable

```
public static void initUsable(java.util.Map existingStyles)
  throws ApplicationException
```

Initialises styles obtained from the document. Ensure to call this after you open the document to avoid exceptions on write to the document.

Parameters:
`existingStyles`

Throws:
`ApplicationException` - if cannot find a matching (at least built-in) style in the open document

logAfter

```
public void logAfter()
```

isTOC

```
public static boolean isTOC(java.lang.String styleName)
```

reset

```
public static void reset()
```

Resets all dynamically set values and collections; ensure to call as a part of clean-up process when existing the app.

Implementation note: Because this is an enumeration, instances are static and when running tests, for example, the collections do NOT get cleaned up. If changing implementation, to not use enumeration, then this would not be needed.

getPreferredNames

```
public java.util.List getPreferredNames()
```

Returns non-empty list of style names, configured by user for this style. Values are only indicative for writing: see `getUsableStyles(boolean, boolean)`.

getUsableStyles

```
public java.util.Map getUsableStyles()
```

Returns all usable styles found in the open document.

getUsableStyles

```
public java.util.Map getUsableStyles(boolean inclBuiltIn,
         boolean inclCustom)
```

Returns map of existing styles as found in an open document, which can be used for writing into document.

Parameters:
`inclBuiltIn` - if `true`, collection includes built-in styles
`inclCustom` - if `true`, collection includes custom styles

getName

`public java.lang.String **getName**()`

Returns *the* style name to be used for this style.

isRecognised

`public boolean **isRecognised**(java.lang.String name)`

Returns whether `name` can be identified as a style name.

isTOC

`public boolean **isTOC**()`

Returns whether this is a TOC style (one out of 9).

isHeading

`public boolean **isHeading**()`

Returns whether this is a heading style (one out of 9).

toShortString

`public java.lang.StringBuilder **toShortString**()`

toString

`public java.lang.String **toString**()`

org.tanjakostic.jcleancim.docgen.writer
Interface Writer

All Subinterfaces:
WordWriter

All Known Implementing Classes:
AbstractWriter

public interface **Writer**
extends

Interface to be implemented by all the UML documentation writers.

Field Summary

public static final	TOOL_CUSTOM_DOC_PROP Name for custom document property holding the application version. Value: **jCleanCim**
public static final	UML_CUSTOM_DOC_PROP Name for custom document property holding the UML model file name. Value: **uml**

Method Summary

abstract java.util.Map	getDocumentMetadata() Returns (potentially empty) custom document properties.
abstract WriterInput	getInput() Returns input used for writing.
abstract java.lang.String	getInputFileNames() Return names of one or more input files used by this writer.
abstract java.lang.String	getOutputFileNames() Return names of one or more output files created by this writer.
abstract java.util.Set	getSupportedFormats() Returns the set of supported formats, as file extensions; e.g., ".doc", ".xml".
abstract void	write() Writes the content from input.

Fields

TOOL_CUSTOM_DOC_PROP

public static final java.lang.String **TOOL_CUSTOM_DOC_PROP**

Name for custom document property holding the application version.
Constant value: **jCleanCim**

UML_CUSTOM_DOC_PROP

`public static final java.lang.String` **`UML_CUSTOM_DOC_PROP`**

Name for custom document property holding the UML model file name.
Constant value: **`uml`**

Methods

getInput

`public abstract` `WriterInput` **`getInput`**`()`

Returns input used for writing. In addition to the actual UML model documentation, the input contains also the required configuration options.

getInputFileNames

`public abstract java.lang.String` **`getInputFileNames`**`()`

Return names of one or more input files used by this writer.

getOutputFileNames

`public abstract java.lang.String` **`getOutputFileNames`**`()`

Return names of one or more output files created by this writer.

getSupportedFormats

`public abstract java.util.Set` **`getSupportedFormats`**`()`

Returns the set of supported formats, as file extensions; e.g., ".doc", ".xml".

getDocumentMetadata

`public abstract java.util.Map` **`getDocumentMetadata`**`()`

Returns (potentially empty) custom document properties. These may be useful to trace meta-information, such as application name and version, the source kind and version, etc.

write

`public abstract void` **`write`**`()`

Writes the content from input.

org.tanjakostic.jcleancim.docgen.writer
Class WriterInput

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.docgen.writer.WriterInput
```

Direct Known Subclasses:
WAXWriterInput, WordWriterInput

public abstract class **WriterInput**
extends java.lang.Object

Group of parameters to construct any documentation writer.

Constructor Summary

protected	`WriterInput(Config cfg, java.lang.String appVersion, java.lang.String modelFileName, boolean skipTiming)` Constructor for testing only.

Method Summary

`java.lang.String`	`getAppVersion()` Returns application version.
`java.lang.String`	`getModelFileName()` Returns name of the model file whose documentation is to be written; potentially empty string.
static `java.lang.String`	`getNameFromModelPath(java.lang.String modelFileAbsPath)`
`boolean`	`isSkipTiming()` Returns whether to skip logging ellapsed times.

Methods inherited from class `java.lang.Object`

`clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait`

Constructors

WriterInput

```
protected WriterInput(Config cfg,
                      java.lang.String appVersion,
                      java.lang.String modelFileName,
                      boolean skipTiming)
```

Constructor for testing only.

Methods

(continued from last page)

getNameFromModelPath

```
protected static java.lang.String getNameFromModelPath(java.lang.String
modelFileAbsPath)
```

getModelFileName

```
public final java.lang.String getModelFileName()
```

Returns name of the model file whose documentation is to be written; potentially empty string.

getAppVersion

```
public final java.lang.String getAppVersion()
```

Returns application version.

isSkipTiming

```
public final boolean isSkipTiming()
```

Returns whether to skip logging ellapsed times.

Package

org.tanjakostic.jcleancim.docgen.writer.word

Classes specific to writing MS documents.

Main classes are:

- `WordWriter` - interface for writing free-form documentation content by replacing placeholders found in input MS Word file, to produce the output MS Word file.
- `WordHelper` - interface for formatting and inserting text, tables, figures, etc. into a MS Word file.
- `AbstractWordWriter` - implementation of the above two for template methods common to binary COM API and text Open XML API.
- `WordWriterInput` - input arguments common to binary COM API and text Open XML API.

org.tanjakostic.jcleancim.docgen.writer.word
Class AbstractWordWriter

```
java.lang.Object
  |-org.tanjakostic.jcleancim.docgen.writer.AbstractWriter
    |-org.tanjakostic.jcleancim.docgen.writer.word.AbstractWordWriter
```

All Implemented Interfaces:
WordHelper, WordWriter, Writer

Direct Known Subclasses:
DocxWordWriter, DocWordWriter

---

public abstract class **AbstractWordWriter**
extends AbstractWriter
implements Writer, WordWriter, WordHelper

Parameters:
O - technology-specific type to access range object.

Fields inherited from interface org.tanjakostic.jcleancim.docgen.writer.Writer
TOOL_CUSTOM_DOC_PROP, UML_CUSTOM_DOC_PROP

Fields inherited from interface org.tanjakostic.jcleancim.docgen.writer.Writer
TOOL_CUSTOM_DOC_PROP, UML_CUSTOM_DOC_PROP

Fields inherited from interface org.tanjakostic.jcleancim.docgen.writer.word.WordHelper
PAGE_WIDTH, POINTS_FOR_1CM

Constructor Summary

protected	AbstractWordWriter(WordWriterInput input) Constructor (a) initialises caption labels, and (b) copies input template into output file that will be filled with this writer.

Method Summary

java.lang.String	appendTextInNewParagraphWithStyle(Range range, TextDescription newText, Style style)
void	clearUndoCache()
Cursor	closeAndReopenDoc(CursorList cursors, Cursor currentCursor)
Caption	createCaption(CaptionKind figure, Range range)
Cursor	createCursor(Placeholder ph, Range limited)

void	createWordApp()
void	exitAppAndSaveDocument()
CursorList	getCursors()
WordWriterInput	getInput()
java.lang.String	getInputFileNames()
java.lang.String	getOutputFileNames()
java.lang.String	getWordAppName()
java.lang.String	getWordAppVersion()
void	initDocgenOptimisationOptions()
boolean	isInTOC(Range range)
static double	pointForPerc(int perc)
CursorList	scanHyperlinkPlaceholderRanges(java.lang.String pattern)
CursorList	scanPlaceholderRanges(java.lang.String pattern, java.util.List figCaptionRanges, java.util.List tabCaptionRanges)
void	setDocgenOptimisationOptions()
java.lang.String	toString()
void	unsetDocgenOptimisationOptions()
void	write()
Cursor	writeAbbrTable(Cursor initCursor, PackageDoc packageDoc)
void	writeByTest(WordHelper.PostProcessor pp)
Cursor	writeClassFromPackage(Cursor cursor, ClassDoc doc, Style headStyle)
Cursor	writeDataIndex(Cursor initCursor, PackageDoc packageDoc)
Cursor	writeDiagram(Cursor cursor, FigureDoc doc)
Cursor	writeExplicitClass(Cursor cursor, ClassDoc doc)
Cursor	writeFcTable(Cursor initCursor, PackageDoc packageDoc)

Cursor	writeLnMapPackage(Cursor initCursor, PackageDoc packageDoc)
Cursor	writePackage(Cursor initCursor, PackageDoc doc, boolean isRoot)
Cursor	writePresCondTable(Cursor initCursor, PackageDoc packageDoc)
Cursor	writeProperties(Cursor initCursor, PropertiesDoc doc)
Cursor	writeSclEnum(Cursor cursor, PackageDoc packageDoc)
Cursor	writeTrgOpTable(Cursor initCursor, PackageDoc packageDoc)

Methods inherited from class org.tanjakostic.jcleancim.docgen.writer.AbstractWriter

getDocumentMetadata

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.docgen.writer.Writer

getDocumentMetadata, getInput, getInputFileNames, getOutputFileNames, getSupportedFormats, write

Methods inherited from interface org.tanjakostic.jcleancim.docgen.writer.word.WordWriter

applyCloseReopen, writeAbbrTable, writeClassFromPackage, writeDataIndex, writeDiagram, writeExplicitClass, writeFcTable, writeLnMapPackage, writePackage, writePresCondTable, writeProperties, writeSclEnum, writeTrgOpTable

Methods inherited from interface org.tanjakostic.jcleancim.docgen.writer.Writer

getDocumentMetadata, getInput, getInputFileNames, getOutputFileNames, getSupportedFormats, write

Methods inherited from interface org.tanjakostic.jcleancim.docgen.writer.word.WordHelper

appendHtmlTextInNewParagraphWithStyle, appendNewLine, appendRawTextInNewParagraphWithStyle, appendText, appendTextInNewParagraph, appendTextInNewParagraphWithStyle, appendTextWithStyle, clearUndoCache, closeAndReopenDoc, closeDoc, collapseRangeToEnd, collapseRangeToStart, collectCaptionParagraphsAndFixLabelsAlsoInTOCs, createCaption, createCursor, createPatternFinder, createRange, createWordApp, duplicateRange, exitAppAndSaveDocument, getCursors, getCustomDocProperties, getDocumentAsRange, getExistingStyles, getRangeParagraphCount, getRangeParagraphOutlineLevel, getRangeParagraphStyleName, getWordAppName, getWordAppVersion, initDocgenOptimisationOptions, insertBookmark, insertCaptionRef, insertFigure, insertFigureCaption, insertHyperlink, insertTable, insertTableCaption, isInTOC, isRangeWithTable, moveStartChar, openDoc, prependNewLine, prependText, scanHyperlinkPlaceholderRanges, scanPlaceholderRanges, setCustomDocProperties, setDocgenOptimisationOptions, unsetDocgenOptimisationOptions, updateFields, updateTablesOf, writeByTest

Constructors

AbstractWordWriter

`protected AbstractWordWriter(WordWriterInput input)`

Constructor (a) initialises caption labels, and (b) copies input template into output file that will be filled with this writer.

Parameters:
`input`

Methods

createWordApp

`public void createWordApp()`

This default implementation does nothing.

getWordAppName

`public java.lang.String getWordAppName()`

This default implementation returns empty string.

getWordAppVersion

`public java.lang.String getWordAppVersion()`

This default implementation returns empty string.

exitAppAndSaveDocument

`public void exitAppAndSaveDocument()`

This default implementation does nothing.

initDocgenOptimisationOptions

`public void initDocgenOptimisationOptions()`

This default implementation does nothing.

setDocgenOptimisationOptions

`public void setDocgenOptimisationOptions()`

(continued from last page)

This default implementation does nothing.

unsetDocgenOptimisationOptions

```
public void unsetDocgenOptimisationOptions()
```

This default implementation does nothing.

scanPlaceholderRanges

```
public CursorList scanPlaceholderRanges(java.lang.String pattern,
        java.util.List figCaptionRanges,
        java.util.List tabCaptionRanges)
```

scanHyperlinkPlaceholderRanges

```
public CursorList scanHyperlinkPlaceholderRanges(java.lang.String pattern)
```

pointForPerc

```
protected static double pointForPerc(int perc)
```

clearUndoCache

```
public void clearUndoCache()
```

This default implementation does nothing.

writeByTest

```
public final void writeByTest(WordHelper.PostProcessor pp)
```

getCursors

```
public final CursorList getCursors()
```

isInTOC

```
public final boolean isInTOC(Range range)
```

createCursor

```
public final Cursor createCursor(Placeholder ph,
        Range limited)
```

createCaption

```
public final Caption createCaption(CaptionKind figure,
        Range range)
```

closeAndReopenDoc

```
public Cursor closeAndReopenDoc(CursorList cursors,
        Cursor currentCursor)
```

This default implementation just returns `currentCursor`, without any closing/reopening. If you need to actually close/reopen the document (as a means of optimising performance), override this method.

appendTextInNewParagraphWithStyle

```
public java.lang.String appendTextInNewParagraphWithStyle(Range range,
        TextDescription newText,
        Style style)
```

You'll always use this one for regular text and tables.

writePackage

```
public Cursor writePackage(Cursor initCursor,
        PackageDoc doc,
        boolean isRoot)
```

writeDataIndex

```
public Cursor writeDataIndex(Cursor initCursor,
        PackageDoc packageDoc)
```

writeLnMapPackage

```
public Cursor writeLnMapPackage(Cursor initCursor,
        PackageDoc packageDoc)
```

(continued from last page)

writePresCondTable

```
public Cursor writePresCondTable(Cursor initCursor,
        PackageDoc packageDoc)
```

writeFcTable

```
public Cursor writeFcTable(Cursor initCursor,
        PackageDoc packageDoc)
```

writeTrgOpTable

```
public Cursor writeTrgOpTable(Cursor initCursor,
        PackageDoc packageDoc)
```

writeAbbrTable

```
public Cursor writeAbbrTable(Cursor initCursor,
        PackageDoc packageDoc)
```

writeSclEnum

```
public Cursor writeSclEnum(Cursor cursor,
        PackageDoc packageDoc)
```

writeExplicitClass

```
public Cursor writeExplicitClass(Cursor cursor,
        ClassDoc doc)
```

writeClassFromPackage

```
public Cursor writeClassFromPackage(Cursor cursor,
        ClassDoc doc,
        Style headStyle)
```

writeProperties

```
public Cursor writeProperties(Cursor initCursor,
        PropertiesDoc doc)
```

(continued from last page)

writeDiagram

```
public Cursor writeDiagram(Cursor cursor,
        FigureDoc doc)
```

getInput

```
public final WordWriterInput getInput()
```

getInputFileNames

```
public final java.lang.String getInputFileNames()
```

getOutputFileNames

```
public final java.lang.String getOutputFileNames()
```

write

```
public final void write()
```

toString

```
public java.lang.String toString()
```

org.tanjakostic.jcleancim.docgen.writer.word
Interface WordHelper

All Known Implementing Classes:
AbstractWordWriter

public interface **WordHelper**
extends

Nested Class Summary

class	WordHelper.PostProcessor WordHelper.PostProcessor

Field Summary

public static final	PAGE_WIDTH Page width in cm. Value: **16**
public static final	POINTS_FOR_1CM Number of points for 1cm (from vba doc). Value: **28.35**

Method Summary

abstract java.lang.String	appendHtmlTextInNewParagraphWithStyle(Range range, java.lang.String newMarkup, Style style)
abstract void	appendNewLine(Range range)
abstract java.lang.String	appendRawTextInNewParagraphWithStyle(Range range, java.lang.String newText, Style style)
abstract java.lang.String	appendText(Range range, java.lang.String newText)
abstract java.lang.String	appendTextInNewParagraph(Range range, java.lang.String newText)
abstract java.lang.String	appendTextInNewParagraphWithStyle(Range range, TextDescription newText, Style style)
abstract java.lang.String	appendTextWithStyle(Range range, java.lang.String newText, Style style)
abstract void	clearUndoCache() When you have large documents and you use a binary (COM) API, you will want to call this one regularly (e.g., for each class doc), so you don't get Word pop-up windows "memory insufficient.
abstract Cursor	closeAndReopenDoc(CursorList cursors, Cursor currentCursor)

abstract void	closeDoc() Closes and saves the MS Word document.
abstract void	collapseRangeToEnd(Range range)
abstract void	collapseRangeToStart(Range range)
abstract java.util.Map	collectCaptionParagraphsAndFixLabelsAlsoInTOCs() Expected to be called after styles and caption labels get properly initialised from the current open document.
abstract Caption	createCaption(CaptionKind figure, Range range)
abstract Cursor	createCursor(Placeholder ph, Range limited)
abstract WordPatternFinder	createPatternFinder(java.lang.String pattern)
abstract Range	createRange(java.lang.Object object)
abstract void	createWordApp() Where applicable, launches (and caches) the MS Word application.
abstract Range	duplicateRange(Range range) FIXME: could go to Range?
abstract void	exitAppAndSaveDocument() Saves MS Word document (and where applicable, exits MS Word application).
abstract CursorList	getCursors()
abstract java.util.Map	getCustomDocProperties()
abstract Range	getDocumentAsRange()
abstract java.util.Map	getExistingStyles() Returns non-empty map of non-null styles read from an open document, with style name as key.
abstract int	getRangeParagraphCount(Range range)
abstract int	getRangeParagraphOutlineLevel(Range range, int paraIdx)
abstract java.lang.String	getRangeParagraphStyleName(Range range, int paraIdx)
abstract java.lang.String	getWordAppName() Returns the MS Word application name.
abstract java.lang.String	getWordAppVersion() Returns the MS Word application version.
abstract void	initDocgenOptimisationOptions() Where applicable, initialises and stores MS Word application options (speed of doc generation) to original values.

abstract void	insertBookmark(Range range, java.lang.String label)
abstract void	insertCaptionRef(Range insertPointRange, CaptionKind kind, int tabNumber, boolean introBeforeCaption) Inserts reference to the caption tabNumber at the start of insertPointRange.
abstract void	insertFigure(Range range, java.io.File pic) Input range includes any potential text within a paragraph.
abstract java.lang.Object	insertFigureCaption(Range range, int expectedSeqNum, java.lang.String captionText)
abstract void	insertHyperlink(Range range, java.lang.String textToDisplay, java.lang.String url)
abstract int	insertTable(Range range, PropertiesDoc doc, Style tabhead, boolean addBookmarks)
abstract void	insertTableCaption(Range range, int expectedSeqNum, java.lang.String captionText)
abstract boolean	isInTOC(Range range)
abstract boolean	isRangeWithTable(java.lang.Object range)
abstract void	moveStartChar(Range range, int count)
abstract void	openDoc() Creates (and caches) the MS Word document.
abstract void	prependNewLine(Range range)
abstract void	prependText(Range range, java.lang.String newText) Prepends text; prepended paragraph will have the same style as the one in range.
abstract CursorList	scanHyperlinkPlaceholderRanges(java.lang.String pattern) This one scans for the 3rd time the document, for hyperlinks, after all the writing has completed after the 2nd scan.
abstract CursorList	scanPlaceholderRanges(java.lang.String pattern, java.util.List figCaptionRanges, java.util.List tabCaptionRanges) This one scans the whole initial document and initialises placeholders, with text, ranges and counts of existing tables and figures (captions) before each of them.
abstract void	setCustomDocProperties(java.util.Map newCustomProps)
abstract void	setDocgenOptimisationOptions() Where applicable, sets MS Word application options to speed performance of doc generation.
abstract void	unsetDocgenOptimisationOptions() Where applicable, brings back the original MS Word application options to values stored with the call to initDocgenOptimisationOptions().
abstract void	updateFields()

abstract void	updateTablesOf(java.lang.String what)
abstract void	writeByTest(WordHelper.PostProcessor pp) If test wants to only read, pass a null pp.

Fields

PAGE_WIDTH

`public static final int PAGE_WIDTH`

Page width in cm.
Constant value: **16**

POINTS_FOR_1CM

`public static final double POINTS_FOR_1CM`

Number of points for 1cm (from vba doc).
Constant value: **28.35**

Methods

writeByTest

`public abstract void writeByTest(WordHelper.PostProcessor pp)`

If test wants to only read, pass a null pp. Otherwise, to perform writing different from the real implementation (for testing purposes), ensure to pass in non-null pp that implements test-specific writing.

createWordApp

`public abstract void createWordApp()`

Where applicable, launches (and caches) the MS Word application.

getWordAppName

`public abstract java.lang.String getWordAppName()`

Returns the MS Word application name.

getWordAppVersion

`public abstract java.lang.String getWordAppVersion()`

Returns the MS Word application version.

openDoc

```
public abstract void openDoc()
  throws java.io.IOException
```

Creates (and caches) the MS Word document.

(continued from last page)

closeDoc

```
public abstract void closeDoc()
  throws java.io.IOException
```

Closes and saves the MS Word document.

exitAppAndSaveDocument

```
public abstract void exitAppAndSaveDocument()
  throws java.io.IOException
```

Saves MS Word document (and where applicable, exits MS Word application).

initDocgenOptimisationOptions

```
public abstract void initDocgenOptimisationOptions()
```

Where applicable, initialises and stores MS Word application options (speed of doc generation) to original values.

setDocgenOptimisationOptions

```
public abstract void setDocgenOptimisationOptions()
```

Where applicable, sets MS Word application options to speed performance of doc generation.

unsetDocgenOptimisationOptions

```
public abstract void unsetDocgenOptimisationOptions()
```

Where applicable, brings back the original MS Word application options to values stored with the call to `initDocgenOptimisationOptions()`.

insertBookmark

```
public abstract void insertBookmark(Range range,
         java.lang.String label)
```

insertHyperlink

```
public abstract void insertHyperlink(Range range,
         java.lang.String textToDisplay,
         java.lang.String url)
```

getExistingStyles

```
public abstract java.util.Map getExistingStyles()
```

Returns non-empty map of non-null styles read from an open document, with style name as key.

updateFields

```
public abstract void updateFields()
```

updateTablesOf

```
public abstract void updateTablesOf(java.lang.String what)
```

getCustomDocProperties

```
public abstract java.util.Map getCustomDocProperties()
```

setCustomDocProperties

```
public abstract void setCustomDocProperties(java.util.Map newCustomProps)
```

getCursors

```
public abstract CursorList getCursors()
```

isInTOC

```
public abstract boolean isInTOC(Range range)
```

closeAndReopenDoc

```
public abstract Cursor closeAndReopenDoc(CursorList cursors,
         Cursor currentCursor)
```

collectCaptionParagraphsAndFixLabelsAlsoInTOCs

```
public abstract java.util.Map collectCaptionParagraphsAndFixLabelsAlsoInTOCs()
```

Expected to be called after styles and caption labels get properly initialised from the current open document. To support non-English versions of Word, we have lot of fixes to apply, in particular with respect to how Word handles caption labels and TOCs of figures and tables. We must handle document with pre-existing captions and TOCs, so we must retrofit those to work with the deduced caption labels and styles.

createRange

```
public abstract Range createRange(java.lang.Object object)
```

createCursor

```
public abstract Cursor createCursor(Placeholder ph,
         Range limited)
```

createCaption

```
public abstract Caption createCaption(CaptionKind figure,
         Range range)
```

scanPlaceholderRanges

```
public abstract CursorList scanPlaceholderRanges(java.lang.String pattern,
         java.util.List figCaptionRanges,
         java.util.List tabCaptionRanges)
```

This one scans the whole initial document and initialises placeholders, with text, ranges and counts of existing tables and figures (captions) before each of them. This is essential for correct references to table and figure captions that we create on the fly. Returns the list of cursors correctly initialised.

scanHyperlinkPlaceholderRanges

```
public abstract CursorList scanHyperlinkPlaceholderRanges(java.lang.String pattern)
```

This one scans for the 3rd time the document, for hyperlinks, after all the writing has completed after the 2nd scan.

createPatternFinder

```
public abstract WordPatternFinder createPatternFinder(java.lang.String pattern)
```

clearUndoCache

```
public abstract void clearUndoCache()
```

When you have large documents and you use a binary (COM) API, you will want to call this one regularly (e.g., for each class doc), so you don't get Word pop-up windows "memory insufficient. Do you want to continue?"

getDocumentAsRange

```
public abstract Range getDocumentAsRange()
```

duplicateRange

```
public abstract Range duplicateRange(Range range)
```

FIXME: could go to Range?

prependText

```
public abstract void prependText(Range range,
         java.lang.String newText)
```

Prepends text; prepended paragraph will have the same style as the one in range. FIXME: move to Range?

(continued from last page)

getRangeParagraphOutlineLevel

```
public abstract int getRangeParagraphOutlineLevel(Range range,
         int paraIdx)
```

appendTextInNewParagraphWithStyle

```
public abstract java.lang.String appendTextInNewParagraphWithStyle(Range range,
         TextDescription newText,
         Style style)
```

appendRawTextInNewParagraphWithStyle

```
public abstract java.lang.String appendRawTextInNewParagraphWithStyle(Range range,
         java.lang.String newText,
         Style style)
```

appendHtmlTextInNewParagraphWithStyle

```
public abstract java.lang.String appendHtmlTextInNewParagraphWithStyle(Range range,
         java.lang.String newMarkup,
         Style style)
```

appendTextWithStyle

```
public abstract java.lang.String appendTextWithStyle(Range range,
         java.lang.String newText,
         Style style)
```

insertFigureCaption

```
public abstract java.lang.Object insertFigureCaption(Range range,
         int expectedSeqNum,
         java.lang.String captionText)
```

insertTableCaption

```
public abstract void insertTableCaption(Range range,
         int expectedSeqNum,
         java.lang.String captionText)
```

(continued from last page)

insertCaptionRef

```
public abstract void insertCaptionRef(Range insertPointRange,
         CaptionKind kind,
         int tabNumber,
         boolean introBeforeCaption)
```

Inserts reference to the caption `tabNumber` at the start of `insertPointRange`.

Parameters:

`introBeforeCaption` - TODO

insertFigure

```
public abstract void insertFigure(Range range,
         java.io.File pic)
```

Input `range` includes any potential text within a paragraph. The method will overwrite that text with the figure from `pic`, and resulting range spans to the start of the paragraph following the inserted `pic`.

insertTable

```
public abstract int insertTable(Range range,
         PropertiesDoc doc,
         Style tabhead,
         boolean addBookmarks)
```

getRangeParagraphStyleName

```
public abstract java.lang.String getRangeParagraphStyleName(Range range,
         int paraIdx)
```

getRangeParagraphCount

```
public abstract int getRangeParagraphCount(Range range)
```

prependNewLine

```
public abstract void prependNewLine(Range range)
```

appendNewLine

```
public abstract void appendNewLine(Range range)
```

appendText

```
public abstract java.lang.String appendText(Range range,
         java.lang.String newText)
```

appendTextInNewParagraph

```
public abstract java.lang.String appendTextInNewParagraph(Range range,
         java.lang.String newText)
```

isRangeWithTable

```
public abstract boolean isRangeWithTable(java.lang.Object range)
```

collapseRangeToEnd

```
public abstract void collapseRangeToEnd(Range range)
```

collapseRangeToStart

```
public abstract void collapseRangeToStart(Range range)
```

moveStartChar

```
public abstract void moveStartChar(Range range,
         int count)
```

org.tanjakostic.jcleancim.docgen.writer.word
Interface WordHelper.PostProcessor

public interface **WordHelper.PostProcessor**
extends

Used for testing only, to play with pure Word stuff, without the notion of the model.

Method Summary

abstract void	postProcess()

Methods

postProcess

```
public abstract void postProcess()
```

org.tanjakostic.jcleancim.docgen.writer.word
Interface WordPatternFinder

public interface **WordPatternFinder**
extends

Method Summary

abstract Range	getRange()
abstract boolean	hasMore()

Methods

getRange

public abstract Range **getRange**()

hasMore

public abstract boolean **hasMore**()

org.tanjakostic.jcleancim.docgen.writer.word
Interface WordWriter

All Superinterfaces:
Writer

All Known Implementing Classes:
AbstractWordWriter

public interface **WordWriter**
extends Writer

Fields inherited from interface org.tanjakostic.jcleancim.docgen.writer.Writer
TOOL_CUSTOM_DOC_PROP, UML_CUSTOM_DOC_PROP

Method Summary

abstract boolean	applyCloseReopen() Returns whether to apply close/reopen hack (may be needed for performance reasons).
abstract Cursor	writeAbbrTable(Cursor initCursor, PackageDoc packageDoc)
abstract Cursor	writeClassFromPackage(Cursor cursor, ClassDoc doc, Style headStyle) Writes all related to a class at the end of range in initCursor.
abstract Cursor	writeDataIndex(Cursor initCursor, PackageDoc packageDoc)
abstract Cursor	writeDiagram(Cursor cursor, FigureDoc doc) Writes all related to a diagram at the end of range in cursor.
abstract Cursor	writeExplicitClass(Cursor cursor, ClassDoc doc) Writes all related to a class, including the title, at the start of range in cursor.
abstract Cursor	writeFcTable(Cursor initCursor, PackageDoc packageDoc)
abstract Cursor	writeLnMapPackage(Cursor initCursor, PackageDoc packageDoc)
abstract Cursor	writePackage(Cursor initCursor, PackageDoc doc, boolean isRoot) Writes all related to a package.
abstract Cursor	writePresCondTable(Cursor initCursor, PackageDoc packageDoc)
abstract Cursor	writeProperties(Cursor initCursor, PropertiesDoc doc) Writes a set of properties as a table at the end of range in cursor.
abstract Cursor	writeSclEnum(Cursor cursor, PackageDoc packageDoc)
abstract Cursor	writeTrgOpTable(Cursor initCursor, PackageDoc packageDoc)

Methods inherited from interface org.tanjakostic.jcleancim.docgen.writer.Writer

getDocumentMetadata, getInput, getInputFileNames, getOutputFileNames, getSupportedFormats, write

Methods

writePackage

```
public abstract Cursor writePackage(Cursor initCursor,
         PackageDoc doc,
         boolean isRoot)
```

Writes all related to a package. For root package, starts from the range in the `initCursor`.

writeDataIndex

```
public abstract Cursor writeDataIndex(Cursor initCursor,
         PackageDoc packageDoc)
```

writeLnMapPackage

```
public abstract Cursor writeLnMapPackage(Cursor initCursor,
         PackageDoc packageDoc)
```

writePresCondTable

```
public abstract Cursor writePresCondTable(Cursor initCursor,
         PackageDoc packageDoc)
```

writeFcTable

```
public abstract Cursor writeFcTable(Cursor initCursor,
         PackageDoc packageDoc)
```

writeTrgOpTable

```
public abstract Cursor writeTrgOpTable(Cursor initCursor,
         PackageDoc packageDoc)
```

writeAbbrTable

```
public abstract Cursor writeAbbrTable(Cursor initCursor,
         PackageDoc packageDoc)
```

(continued from last page)

writeSclEnum

```
public abstract Cursor writeSclEnum(Cursor cursor,
        PackageDoc packageDoc)
```

writeExplicitClass

```
public abstract Cursor writeExplicitClass(Cursor cursor,
        ClassDoc doc)
```

Writes all related to a class, including the title, at the start of range in `cursor`.

writeClassFromPackage

```
public abstract Cursor writeClassFromPackage(Cursor cursor,
        ClassDoc doc,
        Style headStyle)
```

Writes all related to a class at the end of range in `initCursor`.

writeProperties

```
public abstract Cursor writeProperties(Cursor initCursor,
        PropertiesDoc doc)
```

Writes a set of properties as a table at the end of range in `cursor`. Used to write all related to a set of class properties (attributes/literals, or association ends, or operations), or for a collection of one type of properties from one or more packages or classes that need to be put in a table format.

writeDiagram

```
public abstract Cursor writeDiagram(Cursor cursor,
        FigureDoc doc)
```

Writes all related to a diagram at the end of range in `cursor`.

applyCloseReopen

```
public abstract boolean applyCloseReopen()
```

Returns whether to apply close/reopen hack (may be needed for performance reasons).

org.tanjakostic.jcleancim.docgen.writer.word
Class WordWriterInput

```
java.lang.Object
  +-org.tanjakostic.jcleancim.docgen.writer.WriterInput
    +-org.tanjakostic.jcleancim.docgen.writer.word.WordWriterInput
```

public class **WordWriterInput**
extends WriterInput

Constructor Summary

public	WordWriterInput(Config cfg, FreeFormDocumentation freeFormDoc) Constructor.
public	WordWriterInput(java.lang.String appVersion, java.lang.String modelFileName, boolean skipTiming, java.util.Map packageDocs, java.util.Map classDocs, ModelFinder finder, java.lang.String inTemplatePath, java.lang.String outFilePath, boolean useBinaryDoc, boolean introToFigureBefore, int saveRecloseEvery, boolean isDeep, boolean useHyperlinks, java.util.List tocStylePrefixes, java.util.List headingStylePrefixes, java.util.List paraStyles, java.util.List figStyles, java.util.List tabheadStyles, java.util.List tabcellStyles, java.util.List figcaptStyles, java.util.List tabcaptStyles, BookmarkRegistry bmRegistry) Constructor, useful to create an instance if you don't have configuration.

Method Summary

BookmarkRegistry	getBookmarkRegistry() Returns bookmarks registry (used for hyperlinks).
java.util.Map	getClassDocs() Returns "flattened" map of class documentation instances, with *qualified* class name as key (to allow to quickly find the class name from what is read in the placeholder).
java.util.List	getFigcaptStyles() Returns non-empty list of figure caption styles in preferred order of use (last is default).
java.util.List	getFigStyles() Returns non-empty list of figure styles in preferred order of use (last is default).
ModelFinder	getFinder() Returns object that can find in the UML model items specified in placeholders.
java.util.List	getHeadingStylePrefixes() Returns non-empty list of prefixes for heading style in preferred order of use (last is default).
java.lang.String	getInTemplatePath() Returns absolute path of the file used as template for documentation, empty string if template is not used.
java.lang.String	getOutFilePath() Returns absolute path of the file to which to write documentation.

java.util.Map	getPackageDocs() Returns "flattened" map of package documentation instances, with package name as key (to allow to quickly find the package name from what is read in the placeholder).
java.util.List	getParaStyles() Returns non-empty list of paragraph text styles in preferred order of use (last is default).
int	getSaveRecloseEvery() Returns the number of tables (captions) to print before closing and reopening the file; this is an optimisation option that may not be applicable to all writers.
java.util.List	getTabcaptStyles() Returns non-empty list of table caption styles in preferred order of use (last is default).
java.util.List	getTabcellStyles() Returns non-empty list of table cell styles in preferred order of use (last is default).
java.util.List	getTabheadStyles() Returns non-empty list of table head styles in preferred order of use (last is default).
java.util.List	getTocStylePrefixes() Returns non-empty list of prefixes for TOC style in preferred order of use (last is default).
boolean	isDeep() Returns whether to write content for UML packages.
boolean	isIntroToFigureBefore() Returns whether to force figure introduction sentence before the figure.
boolean	isUseBinaryDoc() Returns whether to force use of COM API (binary .doc format).
boolean	isUseHyperlinks() Returns whether to generate hyperlinks (for types of properties).

Methods inherited from class org.tanjakostic.jcleancim.docgen.writer.WriterInput
getAppVersion, getModelFileName, getNameFromModelPath, isSkipTiming

Methods inherited from class java.lang.Object
clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Constructors

WordWriterInput

```
public WordWriterInput(Config cfg,
                       FreeFormDocumentation freeFormDoc)
```

Constructor.

Parameters:

cfg

freeFormDoc - "flattened" map of package documentation instances, with package name as key (to allow to quickly find the package name from what is read in the placeholder). If null or empty, placeholders dealing with packages will all have error and empty content.

WordWriterInput

```
public WordWriterInput(java.lang.String appVersion,
                       java.lang.String modelFileName,
                       boolean skipTiming,
                       java.util.Map packageDocs,
                       java.util.Map classDocs,
                       ModelFinder finder,
                       java.lang.String inTemplatePath,
                       java.lang.String outFilePath,
                       boolean useBinaryDoc,
                       boolean introToFigureBefore,
                       int saveRecloseEvery,
                       boolean isDeep,
                       boolean useHyperlinks,
                       java.util.List tocStylePrefixes,
                       java.util.List headingStylePrefixes,
                       java.util.List paraStyles,
                       java.util.List figStyles,
                       java.util.List tabheadStyles,
                       java.util.List tabcellStyles,
                       java.util.List figcaptStyles,
                       java.util.List tabcaptStyles,
                       BookmarkRegistry bmRegistry)
```

Constructor, useful to create an instance if you don't have configuration.

Parameters:

`appVersion`

`modelFileName`

`skipTiming`

`packageDocs` - "flattened" map of package documentation instances, with package name as key (to allow to quickly find the package name from what is read in the placeholder). If null or empty, placeholders dealing with packages will all have error and empty content.

`classDocs`

`finder` - model facade; if null, most of placeholders will have error.

`inTemplatePath`

`outFilePath`

`useBinaryDoc`

`introToFigureBefore`

`saveRecloseEvery`

`isDeep`

`useHyperlinks`

`tocStylePrefixes`

`headingStylePrefixes`

`paraStyles`

`figStyles`

`tabheadStyles`

`tabcellStyles`

`figcaptStyles`

`tabcaptStyles`

`bmRegistry`

Methods

getPackageDocs

```
public final java.util.Map getPackageDocs()
```

Returns "flattened" map of package documentation instances, with package name as key (to allow to quickly find the package name from what is read in the placeholder). If null or empty, placeholders dealing with packages will by definition all report error.

getClassDocs

`public final java.util.Map getClassDocs()`

Returns "flattened" map of class documentation instances, with *qualified* class name as key (to allow to quickly find the class name from what is read in the placeholder). If null or empty, placeholders dealing with classes will by definition all report error.

getFinder

`public ModelFinder getFinder()`

Returns object that can find in the UML model items specified in placeholders. If null, most of placeholders will by definition report error.

getInTemplatePath

`public java.lang.String getInTemplatePath()`

Returns absolute path of the file used as template for documentation, empty string if template is not used.

getOutFilePath

`public java.lang.String getOutFilePath()`

Returns absolute path of the file to which to write documentation.

isUseBinaryDoc

`public boolean isUseBinaryDoc()`

Returns whether to force use of COM API (binary .doc format).

isIntroToFigureBefore

`public boolean isIntroToFigureBefore()`

Returns whether to force figure introduction sentence before the figure.

getSaveRecloseEvery

`public int getSaveRecloseEvery()`

Returns the number of tables (captions) to print before closing and reopening the file; this is an optimisation option that may not be applicable to all writers.

isDeep

`public boolean isDeep()`

Returns whether to write content for UML packages. Value false is useful for analysing placeholders in the template document, without writing the whole content.

isUseHyperlinks

`public boolean isUseHyperlinks()`

Returns whether to generate hyperlinks (for types of properties).

getTocStylePrefixes

`public java.util.List getTocStylePrefixes()`

Returns non-empty list of prefixes for TOC style in preferred order of use (last is default).

getHeadingStylePrefixes

`public java.util.List getHeadingStylePrefixes()`

Returns non-empty list of prefixes for heading style in preferred order of use (last is default).

getParaStyles

`public java.util.List getParaStyles()`

Returns non-empty list of paragraph text styles in preferred order of use (last is default).

getFigStyles

`public java.util.List getFigStyles()`

Returns non-empty list of figure styles in preferred order of use (last is default).

getTabheadStyles

`public java.util.List getTabheadStyles()`

Returns non-empty list of table head styles in preferred order of use (last is default).

getTabcellStyles

`public java.util.List getTabcellStyles()`

Returns non-empty list of table cell styles in preferred order of use (last is default).

getFigcaptStyles

`public java.util.List getFigcaptStyles()`

Returns non-empty list of figure caption styles in preferred order of use (last is default).

getTabcaptStyles

`public java.util.List getTabcaptStyles()`

Returns non-empty list of table caption styles in preferred order of use (last is default).

getBookmarkRegistry

`public BookmarkRegistry getBookmarkRegistry()`

Returns bookmarks registry (used for hyperlinks).

Package

org.tanjakostic.jcleancim.docgen.writer.word.doc

Classes specific to writing MS documents by means of automation API through Java-COM bridge (Jacob); this implementation is extremely slow (it requires MS Word application and its COM API is just slow), but it supports both .doc and .docx MS Word formats.

All classes are implementations of interfaces from `org.tanjakostic.jcleancim.docgen.writer`.

org.tanjakostic.jcleancim.docgen.writer.word.doc
Class DocWordWriter

```
java.lang.Object
  |-org.tanjakostic.jcleancim.docgen.writer.AbstractWriter
    |-org.tanjakostic.jcleancim.docgen.writer.word.AbstractWordWriter
      |-org.tanjakostic.jcleancim.docgen.writer.word.doc.DocWordWriter
```

All Implemented Interfaces:
Writer, WordHelper, WordWriter

---

public class **DocWordWriter**
extends AbstractWordWriter

This was a huuuuuuuuuge pain, but satisfies the needs of `DocWordWriter` ! Most of it TTD-ed.

Because jacob has no API for constants, when we have to provide constants as arguments to VBA methods, we found the values for those constants with ObjectType browser in VBA (macro) editor.

In general, we split the processing into two steps:

1. scanning placeholders (and recording ranges), and,
2. overwriting placeholders with the content.

For the case where hyperlink creation is enabled, in overwriting user-defined placeholder (with the content from e.g. UML packages), we actually insert / write internal hyperlink placeholders that have information about the text we'd like to see and the bookmark we'd like to link it to. Then we repeat the above two steps once more, when we know all the content within the document and all the bookmarks available to link to.

Field Summary

public static final	FILE_EXTENSIONS
protected static final	OLD_CAPT Value: **true**
public static final	ORIG_CREATE_TABLE Value: **true**
public static final	SEP_AFTER_CAPTION_NUM Value: **–**

Fields inherited from interface `org.tanjakostic.jcleancim.docgen.writer.Writer`
TOOL_CUSTOM_DOC_PROP, UML_CUSTOM_DOC_PROP

Fields inherited from interface `org.tanjakostic.jcleancim.docgen.writer.Writer`
TOOL_CUSTOM_DOC_PROP, UML_CUSTOM_DOC_PROP

Fields inherited from interface `org.tanjakostic.jcleancim.docgen.writer.word.WordHelper`

PAGE_WIDTH, POINTS_FOR_1CM

Constructor Summary

public	DocWordWriter(WordWriterInput input) Constructs this instance and copies input file into output directory; this writer will write into that copy to produce the final document, by replacing placeholders found in it.

Method Summary

java.lang.String	appendHtmlTextInNewParagraphWithStyle(Range range, java.lang.String newMarkup, Style style)
void	appendNewLine(Range range)
java.lang.String	appendRawTextInNewParagraphWithStyle(Range range, java.lang.String newText, Style style)
java.lang.String	appendText(Range range, java.lang.String newText)
java.lang.String	appendTextInNewParagraph(Range range, java.lang.String newText)
java.lang.String	appendTextWithStyle(Range range, java.lang.String newText, Style style)
boolean	applyCloseReopen()
static java.lang.String	asCsv(PropertiesDoc doc, java.lang.String cellSep)
void	clearUndoCache()
Cursor	closeAndReopenDoc(CursorList cursors, Cursor currentCursor) This default implementation just returns `currentCursor`, without any closing/reopening. If you need to actually close/reopen the document (as a means of optimising performance), override this method.
void	closeDoc()
void	collapseRangeToEnd(Range range)
void	collapseRangeToStart(Range range)
java.util.Map	collectCaptionParagraphsAndFixLabelsAlsoInTOCs()
WordPatternFinder	createPatternFinder(java.lang.String msPattern)
Range	createRange(com.jacob.com.Dispatch object)
void	createWordApp()
Range	duplicateRange(Range range)

void	exitAppAndSaveDocument()
java.util.Map	getCustomDocProperties()
Range	getDocumentAsRange()
java.util.Map	getExistingStyles()
int	getRangeParagraphCount(Range range)
int	getRangeParagraphOutlineLevel(Range range, int paraIdx)
java.lang.String	getRangeParagraphStyleName(Range range, int paraIdx)
java.util.Set	getSupportedFormats()
java.lang.String	getWordAppName()
java.lang.String	getWordAppVersion()
void	initDocgenOptimisationOptions()
void	insertBookmark(Range range, java.lang.String label)
void	insertCaptionRef(Range range, CaptionKind kind, int idx, boolean introBeforeCaption)
void	insertFigure(Range range, java.io.File pic)
com.jacob.com.Dispatch	insertFigureCaption(Range range, int expectedSeqNum, java.lang.String text)
void	insertHyperlink(Range range, java.lang.String textToDisplay, java.lang.String url)
int	insertTable(Range range, PropertiesDoc doc, Style tabhead, boolean addBookmarks)
void	insertTableCaption(Range range, int expectedSeqNum, java.lang.String text)
boolean	isRangeWithTable(com.jacob.com.Dispatch range)
void	moveStartChar(Range range, int count)
void	openDoc()
void	prependNewLine(Range range)
void	prependText(Range range, java.lang.String newText)

void	setCustomDocProperties(java.util.Map newCustomProps)
void	setDocgenOptimisationOptions()
void	unsetDocgenOptimisationOptions()
void	updateFields()
void	updateTablesOf(java.lang.String what)

Methods inherited from class org.tanjakostic.jcleancim.docgen.writer.word.AbstractWordWriter

appendTextInNewParagraphWithStyle, clearUndoCache, closeAndReopenDoc, createCaption, createCursor, createWordApp, exitAppAndSaveDocument, getCursors, getInput, getInputFileNames, getOutputFileNames, getWordAppName, getWordAppVersion, initDocgenOptimisationOptions, isInTOC, pointForPerc, scanHyperlinkPlaceholderRanges, scanPlaceholderRanges, setDocgenOptimisationOptions, toString, unsetDocgenOptimisationOptions, write, writeAbbrTable, writeByTest, writeClassFromPackage, writeDataIndex, writeDiagram, writeExplicitClass, writeFcTable, writeLnMapPackage, writePackage, writePresCondTable, writeProperties, writeSclEnum, writeTrgOpTable

Methods inherited from class org.tanjakostic.jcleancim.docgen.writer.AbstractWriter

getDocumentMetadata

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.docgen.writer.Writer

getDocumentMetadata, getInput, getInputFileNames, getOutputFileNames, getSupportedFormats, write

Methods inherited from interface org.tanjakostic.jcleancim.docgen.writer.word.WordWriter

applyCloseReopen, writeAbbrTable, writeClassFromPackage, writeDataIndex, writeDiagram, writeExplicitClass, writeFcTable, writeLnMapPackage, writePackage, writePresCondTable, writeProperties, writeSclEnum, writeTrgOpTable

Methods inherited from interface org.tanjakostic.jcleancim.docgen.writer.Writer

getDocumentMetadata, getInput, getInputFileNames, getOutputFileNames, getSupportedFormats, write

Methods inherited from interface org.tanjakostic.jcleancim.docgen.writer.word.WordHelper

appendHtmlTextInNewParagraphWithStyle, appendNewLine, appendRawTextInNewParagraphWithStyle, appendText, appendTextInNewParagraph, appendTextInNewParagraphWithStyle, appendTextWithStyle, clearUndoCache, closeAndReopenDoc, closeDoc, collapseRangeToEnd, collapseRangeToStart, collectCaptionParagraphsAndFixLabelsAlsoInTOCs, createCaption, createCursor, createPatternFinder, createRange, createWordApp, duplicateRange, exitAppAndSaveDocument, getCursors, getCustomDocProperties, getDocumentAsRange, getExistingStyles, getRangeParagraphCount, getRangeParagraphOutlineLevel, getRangeParagraphStyleName, getWordAppName, getWordAppVersion, initDocgenOptimisationOptions, insertBookmark, insertCaptionRef, insertFigure, insertFigureCaption, insertHyperlink, insertTable, insertTableCaption, isInTOC, isRangeWithTable, moveStartChar, openDoc, prependNewLine, prependText, scanHyperlinkPlaceholderRanges, scanPlaceholderRanges, setCustomDocProperties, setDocgenOptimisationOptions, unsetDocgenOptimisationOptions, updateFields, updateTablesOf, writeByTest

Fields

ORIG_CREATE_TABLE

`public static final boolean` **`ORIG_CREATE_TABLE`**

Constant value: **`true`**

OLD_CAPT

`protected static final boolean` **`OLD_CAPT`**

Constant value: **`true`**

SEP_AFTER_CAPTION_NUM

`public static final java.lang.String` **`SEP_AFTER_CAPTION_NUM`**

Constant value: **`-`**

FILE_EXTENSIONS

`public static final java.util.List` **`FILE_EXTENSIONS`**

Constructors

DocWordWriter

`public` **`DocWordWriter`**`(WordWriterInput input)`

Constructs this instance and copies input file into output directory; this writer will write into that copy to produce the final document, by replacing placeholders found in it.

Throws:

`UnsupportedInputFormatException` - if configured with template in unsupported format

(continued from last page)

`UnsupportedOutputFormatException` - if configured with output in unsupported format
`IOException` - if fails to copy input template into output directory.

Methods

asCsv

```
public static java.lang.String asCsv(PropertiesDoc doc,
         java.lang.String cellSep)
```

createWordApp

```
public void createWordApp()
```

This default implementation does nothing.

getWordAppName

```
public java.lang.String getWordAppName()
```

This default implementation returns empty string.

getWordAppVersion

```
public java.lang.String getWordAppVersion()
```

This default implementation returns empty string.

openDoc

```
public void openDoc()
```

closeDoc

```
public void closeDoc()
```

exitAppAndSaveDocument

```
public void exitAppAndSaveDocument()
```

This default implementation does nothing.

(continued from last page)

initDocgenOptimisationOptions

```
public void initDocgenOptimisationOptions()
```

This default implementation does nothing.

setDocgenOptimisationOptions

```
public void setDocgenOptimisationOptions()
```

This default implementation does nothing.

unsetDocgenOptimisationOptions

```
public void unsetDocgenOptimisationOptions()
```

This default implementation does nothing.

insertBookmark

```
public void insertBookmark(Range range,
         java.lang.String label)
```

insertHyperlink

```
public void insertHyperlink(Range range,
         java.lang.String textToDisplay,
         java.lang.String url)
```

getExistingStyles

```
public java.util.Map getExistingStyles()
```

It will create all custom styles, and only those built-in styles we are interested in.

updateFields

```
public void updateFields()
```

updateTablesOf

```
public void updateTablesOf(java.lang.String what)
```

getCustomDocProperties

```
public java.util.Map getCustomDocProperties()
```

setCustomDocProperties

```
public void setCustomDocProperties(java.util.Map newCustomProps)
```

closeAndReopenDoc

```
public Cursor closeAndReopenDoc(CursorList cursors,
        Cursor currentCursor)
```

This default implementation just returns `currentCursor`, without any closing/reopening. If you need to actually close/reopen the document (as a means of optimising performance), override this method.

Implementation of closing/reopening the MS Word document.

collectCaptionParagraphsAndFixLabelsAlsoInTOCs

```
public java.util.Map collectCaptionParagraphsAndFixLabelsAlsoInTOCs()
```

Implementation note: In earlier versions, tried with GetCrossReferenceItems on document, but that was not reliable, so now doing manual house-keeping to achieve predictable results.

createRange

```
public Range createRange(com.jacob.com.Dispatch object)
```

createPatternFinder

```
public final WordPatternFinder createPatternFinder(java.lang.String msPattern)
```

clearUndoCache

```
public void clearUndoCache()
```

This default implementation does nothing.

getDocumentAsRange

```
public Range getDocumentAsRange()
```

duplicateRange

```
public Range duplicateRange(Range range)
```

prependText

```
public void prependText(Range range,
         java.lang.String newText)
```

getRangeParagraphOutlineLevel

```
public int getRangeParagraphOutlineLevel(Range range,
         int paraIdx)
```

appendRawTextInNewParagraphWithStyle

```
public java.lang.String appendRawTextInNewParagraphWithStyle(Range range,
         java.lang.String newText,
         Style style)
```

appendHtmlTextInNewParagraphWithStyle

```
public java.lang.String appendHtmlTextInNewParagraphWithStyle(Range range,
         java.lang.String newMarkup,
         Style style)
```

appendTextWithStyle

```
public java.lang.String appendTextWithStyle(Range range,
         java.lang.String newText,
         Style style)
```

insertFigureCaption

```
public com.jacob.com.Dispatch insertFigureCaption(Range range,
         int expectedSeqNum,
         java.lang.String text)
```

insertTableCaption

```
public void insertTableCaption(Range range,
         int expectedSeqNum,
         java.lang.String text)
```

insertCaptionRef

```
public void insertCaptionRef(Range range,
         CaptionKind kind,
         int idx,
         boolean introBeforeCaption)
```

This will fail with exception if caption not found.

insertFigure

```
public void insertFigure(Range range,
         java.io.File pic)
```

insertTable

```
public int insertTable(Range range,
         PropertiesDoc doc,
         Style tabhead,
         boolean addBookmarks)
```

Returns expanded range to the end of the table. We do not collapse original range here, as we assume the caption will need to be inserted before the table.

getRangeParagraphStyleName

```
public java.lang.String getRangeParagraphStyleName(Range range,
         int paraIdx)
```

getRangeParagraphCount

```
public int getRangeParagraphCount(Range range)
```

prependNewLine

```
public void prependNewLine(Range range)
```

appendNewLine

```
public void appendNewLine(Range range)
```

appendText

```
public java.lang.String appendText(Range range,
         java.lang.String newText)
```

(continued from last page)

appendTextInNewParagraph

```
public java.lang.String appendTextInNewParagraph(Range range,
         java.lang.String newText)
```

isRangeWithTable

```
public boolean isRangeWithTable(com.jacob.com.Dispatch range)
```

collapseRangeToEnd

```
public void collapseRangeToEnd(Range range)
```

collapseRangeToStart

```
public void collapseRangeToStart(Range range)
```

moveStartChar

```
public void moveStartChar(Range range,
         int count)
```

applyCloseReopen

```
public boolean applyCloseReopen()
```

getSupportedFormats

```
public java.util.Set getSupportedFormats()
```

Package

org.tanjakostic.jcleancim.docgen.writer.word.docx

Classes specific to writing MS documents in OpenXML format; this implementation is fast (it does not require MS Word application), but it supports only .docx MS Word format.

All classes are implementations of interfaces from `org.tanjakostic.jcleancim.docgen.writer`.

org.tanjakostic.jcleancim.docgen.writer.word.docx
Class DocxWordWriter

```
java.lang.Object
  +-org.tanjakostic.jcleancim.docgen.writer.AbstractWriter
    +-org.tanjakostic.jcleancim.docgen.writer.word.AbstractWordWriter
      +-org.tanjakostic.jcleancim.docgen.writer.word.docx.DocxWordWriter
```

All Implemented Interfaces:
Writer, WordHelper, WordWriter

---

public class **DocxWordWriter**
extends AbstractWordWriter

Field Summary

public static final	FILE_EXTENSIONS

Fields inherited from interface org.tanjakostic.jcleancim.docgen.writer.Writer
TOOL_CUSTOM_DOC_PROP, UML_CUSTOM_DOC_PROP

Fields inherited from interface org.tanjakostic.jcleancim.docgen.writer.Writer
TOOL_CUSTOM_DOC_PROP, UML_CUSTOM_DOC_PROP

Fields inherited from interface org.tanjakostic.jcleancim.docgen.writer.word.WordHelper
PAGE_WIDTH, POINTS_FOR_1CM

Constructor Summary

public	DocxWordWriter(WordWriterInput input) Constructor.

Method Summary

java.lang.String	appendHtmlTextInNewParagraphWithStyle(Range range, java.lang.String newMarkup, Style style)
void	appendNewLine(Range range)
java.lang.String	appendRawTextInNewParagraphWithStyle(Range range, java.lang.String newText, Style style)
java.lang.String	appendText(Range range, java.lang.String newText)
java.lang.String	appendTextInNewParagraph(Range range, java.lang.String newText)

java.lang.String	appendTextWithStyle(Range range, java.lang.String newText, Style style)
boolean	applyCloseReopen()
void	closeDoc()
void	collapseRangeToEnd(Range range)
void	collapseRangeToStart(Range range)
java.util.Map	collectCaptionParagraphsAndFixLabelsAlsoInTOCs()
WordPatternFinder	createPatternFinder(java.lang.String pattern)
Range	createRange(java.lang.Object object)
Range	duplicateRange(Range range)
java.util.Map	getCustomDocProperties()
Range	getDocumentAsRange()
java.util.Map	getExistingStyles()
int	getRangeParagraphCount(Range range)
int	getRangeParagraphOutlineLevel(Range range, int paraIdx)
java.lang.String	getRangeParagraphStyleName(Range range, int paraIdx)
java.util.Set	getSupportedFormats()
void	insertBookmark(Range range, java.lang.String label)
void	insertCaptionRef(Range insertPointRange, CaptionKind kind, int figNumber, boolean introBeforeCaption)
void	insertFigure(Range range, java.io.File pic)
java.lang.Object	insertFigureCaption(Range range, int expectedSeqNum, java.lang.String captionText)
void	insertHyperlink(Range range, java.lang.String textToDisplay, java.lang.String url)
int	insertTable(Range range, PropertiesDoc doc, Style tabhead, boolean addBookmarks)
void	insertTableCaption(Range range, int expectedSeqNum, java.lang.String captionText)

boolean	isRangeWithTable(java.lang.Object range)
void	moveStartChar(Range range, int count)
void	openDoc()
void	prependNewLine(Range range)
void	prependText(Range range, java.lang.String newText)
void	setCustomDocProperties(java.util.Map newCustomProps)
void	updateFields()
void	updateTablesOf(java.lang.String what)

Methods inherited from class org.tanjakostic.jcleancim.docgen.writer.word.AbstractWordWriter

appendTextInNewParagraphWithStyle, clearUndoCache, closeAndReopenDoc, createCaption, createCursor, createWordApp, exitAppAndSaveDocument, getCursors, getInput, getInputFileNames, getOutputFileNames, getWordAppName, getWordAppVersion, initDocgenOptimisationOptions, isInTOC, pointForPerc, scanHyperlinkPlaceholderRanges, scanPlaceholderRanges, setDocgenOptimisationOptions, toString, unsetDocgenOptimisationOptions, write, writeAbbrTable, writeByTest, writeClassFromPackage, writeDataIndex, writeDiagram, writeExplicitClass, writeFcTable, writeLnMapPackage, writePackage, writePresCondTable, writeProperties, writeSclEnum, writeTrgOpTable

Methods inherited from class org.tanjakostic.jcleancim.docgen.writer.AbstractWriter

getDocumentMetadata

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.docgen.writer.Writer

getDocumentMetadata, getInput, getInputFileNames, getOutputFileNames, getSupportedFormats, write

Methods inherited from interface org.tanjakostic.jcleancim.docgen.writer.word.WordWriter

applyCloseReopen, writeAbbrTable, writeClassFromPackage, writeDataIndex, writeDiagram, writeExplicitClass, writeFcTable, writeLnMapPackage, writePackage, writePresCondTable, writeProperties, writeSclEnum, writeTrgOpTable

Methods inherited from interface org.tanjakostic.jcleancim.docgen.writer.Writer

getDocumentMetadata, getInput, getInputFileNames, getOutputFileNames, getSupportedFormats, write

Methods inherited from interface org.tanjakostic.jcleancim.docgen.writer.word.WordHelper

appendHtmlTextInNewParagraphWithStyle, appendNewLine, appendRawTextInNewParagraphWithStyle, appendText, appendTextInNewParagraph, appendTextInNewParagraphWithStyle, appendTextWithStyle, clearUndoCache, closeAndReopenDoc, closeDoc, collapseRangeToEnd, collapseRangeToStart, collectCaptionParagraphsAndFixLabelsAlsoInTOCs, createCaption, createCursor, createPatternFinder, createRange, createWordApp, duplicateRange, exitAppAndSaveDocument, getCursors, getCustomDocProperties, getDocumentAsRange, getExistingStyles, getRangeParagraphCount, getRangeParagraphOutlineLevel, getRangeParagraphStyleName, getWordAppName, getWordAppVersion, initDocgenOptimisationOptions, insertBookmark, insertCaptionRef, insertFigure, insertFigureCaption, insertHyperlink, insertTable, insertTableCaption, isInTOC, isRangeWithTable, moveStartChar, openDoc, prependNewLine, prependText, scanHyperlinkPlaceholderRanges, scanPlaceholderRanges, setCustomDocProperties, setDocgenOptimisationOptions, unsetDocgenOptimisationOptions, updateFields, updateTablesOf, writeByTest

Fields

FILE_EXTENSIONS

```
public static final java.util.List FILE_EXTENSIONS
```

Constructors

DocxWordWriter

```
public DocxWordWriter(WordWriterInput input)
```

Constructor.

Parameters:
input

Throws:
UnsupportedInputFormatException
UnsupportedOutputFormatException
IOException

Methods

openDoc

```
public void openDoc()
  throws java.io.IOException
```

closeDoc

```
public void closeDoc()
  throws java.io.IOException
```

insertBookmark

```
public void insertBookmark(Range range,
         java.lang.String label)
```

insertHyperlink

```
public void insertHyperlink(Range range,
         java.lang.String textToDisplay,
         java.lang.String url)
```

getExistingStyles

```
public java.util.Map getExistingStyles()
```

updateFields

```
public void updateFields()
```

updateTablesOf

```
public void updateTablesOf(java.lang.String what)
```

getCustomDocProperties

```
public java.util.Map getCustomDocProperties()
```

setCustomDocProperties

```
public void setCustomDocProperties(java.util.Map newCustomProps)
```

collectCaptionParagraphsAndFixLabelsAlsoInTOCs

```
public java.util.Map collectCaptionParagraphsAndFixLabelsAlsoInTOCs()
```

createRange

```
public Range createRange(java.lang.Object object)
```

createPatternFinder

```
public WordPatternFinder createPatternFinder(java.lang.String pattern)
```

getDocumentAsRange

```
public Range getDocumentAsRange()
```

duplicateRange

```
public Range duplicateRange(Range range)
```

prependText

```
public void prependText(Range range,
         java.lang.String newText)
```

getRangeParagraphOutlineLevel

```
public int getRangeParagraphOutlineLevel(Range range,
         int paraIdx)
```

appendRawTextInNewParagraphWithStyle

```
public java.lang.String appendRawTextInNewParagraphWithStyle(Range range,
         java.lang.String newText,
         Style style)
```

appendHtmlTextInNewParagraphWithStyle

```
public java.lang.String appendHtmlTextInNewParagraphWithStyle(Range range,
         java.lang.String newMarkup,
         Style style)
```

appendTextWithStyle

```
public java.lang.String appendTextWithStyle(Range range,
         java.lang.String newText,
         Style style)
```

(continued from last page)

insertFigureCaption

```
public java.lang.Object insertFigureCaption(Range range,
         int expectedSeqNum,
         java.lang.String captionText)
```

insertTableCaption

```
public void insertTableCaption(Range range,
         int expectedSeqNum,
         java.lang.String captionText)
```

insertCaptionRef

```
public void insertCaptionRef(Range insertPointRange,
         CaptionKind kind,
         int figNumber,
         boolean introBeforeCaption)
```

insertFigure

```
public void insertFigure(Range range,
         java.io.File pic)
```

insertTable

```
public int insertTable(Range range,
         PropertiesDoc doc,
         Style tabhead,
         boolean addBookmarks)
```

getRangeParagraphStyleName

```
public java.lang.String getRangeParagraphStyleName(Range range,
         int paraIdx)
```

getRangeParagraphCount

```
public int getRangeParagraphCount(Range range)
```

prependNewLine

```
public void prependNewLine(Range range)
```

appendNewLine

```
public void appendNewLine(Range range)
```

appendText

```
public java.lang.String appendText(Range range,
         java.lang.String newText)
```

appendTextInNewParagraph

```
public java.lang.String appendTextInNewParagraph(Range range,
         java.lang.String newText)
```

isRangeWithTable

```
public boolean isRangeWithTable(java.lang.Object range)
```

collapseRangeToEnd

```
public void collapseRangeToEnd(Range range)
```

collapseRangeToStart

```
public void collapseRangeToStart(Range range)
```

moveStartChar

```
public void moveStartChar(Range range,
         int count)
```

applyCloseReopen

```
public boolean applyCloseReopen()
```

getSupportedFormats

```
public java.util.Set getSupportedFormats()
```

Package

org.tanjakostic.jcleancim.docgen.writer.xml

Classes specific to writing IEC 61850 name space definition files.

The main classes are:

- `WAXWriter` - writes the UML model content to XML file.

org.tanjakostic.jcleancim.docgen.writer.xml
Class WAXWriter

```
java.lang.Object
  |-org.tanjakostic.jcleancim.docgen.writer.AbstractWriter
    |-org.tanjakostic.jcleancim.docgen.writer.xml.WAXWriter
```

All Implemented Interfaces:
Writer

---

public class **WAXWriter**
extends AbstractWriter

Writes UML model content in XML format for Web access.

Field Summary

public static final	FILE_EXTENSIONS

Fields inherited from interface org.tanjakostic.jcleancim.docgen.writer.Writer

TOOL_CUSTOM_DOC_PROP, UML_CUSTOM_DOC_PROP

Constructor Summary

public	WAXWriter(WAXWriterInput input) Constructs this instance and copies schemas into output directory.

Method Summary

WAXWriterInput	getInput()
java.lang.String	getInputFileNames()
java.lang.String	getOutputFileNames()
java.util.Set	getSupportedFormats()
void	write()

Methods inherited from class org.tanjakostic.jcleancim.docgen.writer.AbstractWriter

getDocumentMetadata

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.docgen.writer.Writer

getDocumentMetadata, getInput, getInputFileNames, getOutputFileNames, getSupportedFormats, write

Fields

FILE_EXTENSIONS

public static final java.util.List **FILE_EXTENSIONS**

Constructors

WAXWriter

public **WAXWriter**(WAXWriterInput input)

Constructs this instance and copies schemas into output directory.

Throws:
IOException - if fails to copy schemas into output directory.

Methods

getInput

public WAXWriterInput **getInput**()

getInputFileNames

public java.lang.String **getInputFileNames**()

getOutputFileNames

public java.lang.String **getOutputFileNames**()

getSupportedFormats

public java.util.Set **getSupportedFormats**()

write

public void **write**()

org.tanjakostic.jcleancim.docgen.writer.xml
Class WAXWriterInput

```
java.lang.Object
  |-org.tanjakostic.jcleancim.docgen.writer.WriterInput
    |-org.tanjakostic.jcleancim.docgen.writer.xml.WAXWriterInput
```

public class **WAXWriterInput**
extends WriterInput

Constructor Summary

public	WAXWriterInput(Config cfg, FixedFormDocumentation fixedFormDocumentation) Constructor.
public	WAXWriterInput(java.lang.String appVersion, java.lang.String modelFileName, boolean skipTiming, FixedFormDocumentation fixedFormDocumentation, java.lang.String inXsdWebaccessPath, java.lang.String outXmlSpecPath, java.lang.String outXmlDocPath, java.lang.String outXsdWebaccessPath) Constructor, useful to create an instance if you don't have configuration.

Method Summary

FixedFormDocumentation	getFixedFormDocumentation()
java.lang.String	getInXsdWebaccessPath()
java.lang.String	getOutXmlDocPath()
java.lang.String	getOutXmlSpecPath()
java.lang.String	getOutXsdWebaccessPath()

Methods inherited from class org.tanjakostic.jcleancim.docgen.writer.WriterInput

getAppVersion, getModelFileName, getNameFromModelPath, isSkipTiming

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Constructors

(continued from last page)

WAXWriterInput

```
public WAXWriterInput(Config cfg,
                      FixedFormDocumentation fixedFormDocumentation)
```

Constructor.

Parameters:

`cfg`

`fixedFormDocumentation` - FIXME "flattened" map of package documentation instances, with package name as key (to allow to quickly find the package name from what is read in the placeholder). If null or empty, placeholders dealing with packages will all have error and empty content.

WAXWriterInput

```
public WAXWriterInput(java.lang.String appVersion,
                      java.lang.String modelFileName,
                      boolean skipTiming,
                      FixedFormDocumentation fixedFormDocumentation,
                      java.lang.String inXsdWebaccessPath,
                      java.lang.String outXmlSpecPath,
                      java.lang.String outXmlDocPath,
                      java.lang.String outXsdWebaccessPath)
```

Constructor, useful to create an instance if you don't have configuration.

Parameters:

`appVersion`

`modelFileName`

`fixedFormDocumentation` - scoped package docs, categorised by nature, then by name space.

`inXsdWebaccessPath`

`outXmlSpecPath`

`outXmlDocPath`

`outXsdWebaccessPath`

Methods

getFixedFormDocumentation

```
public FixedFormDocumentation getFixedFormDocumentation()
```

getInXsdWebaccessPath

```
public java.lang.String getInXsdWebaccessPath()
```

getOutXmlSpecPath

```
public java.lang.String getOutXmlSpecPath()
```

getOutXmlDocPath

```
public java.lang.String getOutXmlDocPath()
```

(continued from last page)

getOutXsdWebaccessPath

```
public java.lang.String getOutXsdWebaccessPath()
```

Package

org.tanjakostic.jcleancim.experimental.builder.rdfs

FIXME: This is experimental and absolutely not tested - don't use!

The package contains the `RdfsParser`, which can parse the CIM profiles in org.tanjakostic.jcleancim.common.Config#XSD_EXT format, as generated with CIMTool.

The result of parsing is contained in `RdfsModel` and its related classes.

Implementation note: An option was to use jena libraries, but it is pretty big with all of its dependencies. We therefore opted to adapt an old parser that was parsing RDFS generated with XPetal from Rose long time ago.

org.tanjakostic.jcleancim.experimental.builder.rdfs
Class CimSchemaException

```
java.lang.Object
  |-java.lang.Throwable
    |-java.lang.Exception
      |-org.tanjakostic.jcleancim.experimental.builder.rdfs.CimSchemaException
```

All Implemented Interfaces:
java.io.Serializable

---

public class **CimSchemaException**
extends java.lang.Exception

Base for schema-related exceptions.

Constructor Summary

public	CimSchemaException()
public	CimSchemaException(java.lang.String message)
public	CimSchemaException(java.lang.Throwable cause)
public	CimSchemaException(java.lang.String msg, java.lang.Throwable cause)

Methods inherited from class `java.lang.Throwable`

addSuppressed, fillInStackTrace, getCause, getLocalizedMessage, getMessage, getStackTrace, getSuppressed, initCause, printStackTrace, printStackTrace, printStackTrace, setStackTrace, toString

Methods inherited from class `java.lang.Object`

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

---

Constructors

CimSchemaException

public **CimSchemaException**()

CimSchemaException

public **CimSchemaException**(java.lang.String message)

(continued from last page)

CimSchemaException

```
public CimSchemaException(java.lang.Throwable cause)
```

CimSchemaException

```
public CimSchemaException(java.lang.String msg,
                          java.lang.Throwable cause)
```

org.tanjakostic.jcleancim.experimental.builder.rdfs
Class RdfsClass

```
java.lang.Object
  |-org.tanjakostic.jcleancim.experimental.builder.rdfs.RdfsElem
    |-org.tanjakostic.jcleancim.experimental.builder.rdfs.RdfsClass
```

public final class **RdfsClass**
extends RdfsElem

CIM RDF Schema element representing the UML class.

Fields inherited from class `org.tanjakostic.jcleancim.experimental.builder.rdfs.RdfsElem`
`invalidCharsPattern`

Method Summary

boolean	`equals(java.lang.Object obj)`
java.lang.String	`getAttrInitValue(java.lang.String attrName)` Returns cached initial value for the attribute (deduced by reflection).
java.util.List	`getDiffs(RdfsElem other)`
java.util.List	`getEnumLiterals()`
java.lang.String	`getKind()` Here the kinds of classes:
java.util.Map	`getSubclasses()`
int	`hashCode()`
boolean	`isCompoundClass()`
boolean	`isDatatypeClass()`
boolean	`isEnumClass()`
boolean	`isPrimitiveClass()`
boolean	`isSubclass()`
java.lang.String	`toString()`
java.lang.String	`toStringLong()`

Methods inherited from class `org.tanjakostic.jcleancim.experimental.builder.rdfs.RdfsElem`

equals, formatDiff, getAbout, getComment, getDiffs, getKind, getLabel, getModel, getName, getPackage, getSchemaLabel, hashCode, toString, toStringLong

Methods inherited from class `java.lang.Object`

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods

isSubclass

public boolean **isSubclass**()

getSubclasses

public java.util.Map **getSubclasses**()

getEnumLiterals

public java.util.List **getEnumLiterals**()

isPrimitiveClass

public boolean **isPrimitiveClass**()

isDatatypeClass

public boolean **isDatatypeClass**()

isEnumClass

public boolean **isEnumClass**()

isCompoundClass

public boolean **isCompoundClass**()

(continued from last page)

getKind

`public java.lang.String **getKind**()`

Here the kinds of classes:

```
 First level classes:  non-sub (root)   Enum, Datatype, Compound, Primitive (leaf) Other
(leaf)
 Sub-class:            sub
 Deepest sub-class:    sub (leaf)
```

Returns:
string describing the kind of this class.

getAttrInitValue

`public java.lang.String **getAttrInitValue**(java.lang.String attrName)`

Returns cached initial value for the attribute (deduced by reflection).

Parameters:
`attrName`

Returns:
cached initial value for the attribute (deduced by reflection).

toString

`public java.lang.String **toString**()`

toStringLong

`public java.lang.String **toStringLong**()`

hashCode

`public int **hashCode**()`

Uses all instance fields except for _model.

equals

`public boolean **equals**(java.lang.Object obj)`

(continued from last page)

Uses all instance fields except for _model. We do track differences for relevant fields, but we ignore the difference (i.e., don't return false) in following two cases:

- _package field when dialects are different and this element is not an RdfsClass (RDF has info on package for classes only, so in case of RdfsClass we always compare normally).
- _comment field when dialects are different (with OWL, we have to generate attributes of datatype classes, as well as 4 CIM primitive classes, so we have no original comments).

getDiffs

```
public final java.util.List getDiffs(RdfsElem other)
```

Calculates differences between this instance and `other` and returns them in a list of Strings. While `RdfsElem.equals(Object)` method ignores some known differences between elements in different dialects, this method catches them all.

Implementation note: Subclasses that override this method are expected to first call super.runDiff().

org.tanjakostic.jcleancim.experimental.builder.rdfs
Class RdfsElem

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.experimental.builder.rdfs.RdfsElem
```

Direct Known Subclasses:
RdfsClass, RdfsEnumLiteral, RdfsPackage, RdfsProperty

---

public abstract class **RdfsElem**
extends java.lang.Object

Base class with common implementation for all CIM RDF/OWL Schema elements.

This implementation delegates issue tracking and logging to its `model`.

Field Summary

public static final	`invalidCharsPattern` Pattern of valid characters for all the CIM tokens.

Constructor Summary

protected	`RdfsElem(RdfsModel model, org.w3c.dom.Element elem)` Initialises fields common to RDF elements from DOM Element.
protected	`RdfsElem(RdfsModel model, java.lang.String about, java.lang.String label, java.lang.String comment, java.lang.String pckage, boolean validateAbout)` Initialises fields common to RDF elements independently of DOM Element.

Method Summary

boolean	`equals(java.lang.Object obj)` Uses all instance fields except for _model.
RdfsDifference	`formatDiff(java.lang.String field, java.lang.String thisVal, java.lang.String otherVal, RdfsElem other)`
java.lang.String	`getAbout()`
java.lang.String	`getComment()`
java.util.List	`getDiffs(RdfsElem other)` Calculates differences between this instance and `other` and returns them in a list of Strings.
abstract java.lang.String	`getKind()` Returns the string describing the kind of this element.
java.lang.String	`getLabel()`
RdfsModel	`getModel()` Returns the model containing this element.

java.lang.String	getName()
java.lang.String	getPackage()
java.lang.String	getSchemaLabel()
int	hashCode() Uses all instance fields except for _model.
java.lang.String	toString()
java.lang.String	toStringLong()

Methods inherited from class java.lang.Object
clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Fields

invalidCharsPattern

```
public static final java.util.regex.Pattern invalidCharsPattern
```

Pattern of valid characters for all the CIM tokens.

Constructors

RdfsElem

```
protected RdfsElem(RdfsModel model,
                   org.w3c.dom.Element elem)
```

Initialises fields common to RDF elements from DOM Element.

Parameters:

`model`

`elem` - DOM element

Throws:

`CimSchemaException` - if about attribute contains an invalid URI.

RdfsElem

```
protected RdfsElem(RdfsModel model,
                   java.lang.String about,
                   java.lang.String label,
                   java.lang.String comment,
                   java.lang.String pckage,
                   boolean validateAbout)
```

Initialises fields common to RDF elements independently of DOM Element.

Parameters:

`about` - non-null (schemaLabel#name)

(continued from last page)

`label` - non-null
`comment` - if null, will be set to empty string
`pckage` - if null, will be set to "?"
`validateAbout` - whether to validate `about`.

Throws:
`CimSchemaException` - if about attribute contains an invalid URI.

Methods

getKind

`public abstract java.lang.String` **`getKind`**`()`

Returns the string describing the kind of this element.

Returns:
string describing the kind of this element.

getModel

`public RdfsModel` **`getModel`**`()`

Returns the model containing this element.

Returns:
model containing this element.

getAbout

`public final java.lang.String` **`getAbout`**`()`

getName

`public final java.lang.String` **`getName`**`()`

getSchemaLabel

`public final java.lang.String` **`getSchemaLabel`**`()`

getComment

`public final java.lang.String` **`getComment`**`()`

getLabel

`public final java.lang.String` **`getLabel`**`()`

getPackage

```
public final java.lang.String getPackage()
```

getDiffs

```
public java.util.List getDiffs(RdfsElem other)
```

Calculates differences between this instance and `other` and returns them in a list of Strings. While `equals(Object)` method ignores some known differences between elements in different dialects, this method catches them all.

Implementation note: Subclasses that override this method are expected to first call super.runDiff().

Parameters:
`other`

formatDiff

```
protected final RdfsDifference formatDiff(java.lang.String field,
         java.lang.String thisVal,
         java.lang.String otherVal,
         RdfsElem other)
```

toString

```
public java.lang.String toString()
```

toStringLong

```
public java.lang.String toStringLong()
```

hashCode

```
public int hashCode()
```

Uses all instance fields except for _model.

equals

```
public boolean equals(java.lang.Object obj)
```

Uses all instance fields except for _model. We do track differences for relevant fields, but we ignore the difference (i.e., don't return false) in following two cases:

- _package field when dialects are different and this element is not an RdfsClass (RDF has info on package for classes only, so in case of RdfsClass we always compare normally).
- _comment field when dialects are different (with OWL, we have to generate attributes of datatype classes, as well as 4 CIM primitive classes, so we have no original comments).

org.tanjakostic.jcleancim.experimental.builder.rdfs
Class RdfsEnumLiteral

```
java.lang.Object
  |-org.tanjakostic.jcleancim.experimental.builder.rdfs.RdfsElem
    |-org.tanjakostic.jcleancim.experimental.builder.rdfs.RdfsEnumLiteral
```

public final class **RdfsEnumLiteral**
extends RdfsElem

CIM RDF Schema element representing the UML attribute defined on an enum class.

Fields inherited from class `org.tanjakostic.jcleancim.experimental.builder.rdfs.RdfsElem`
`invalidCharsPattern`

Method Summary

boolean	equals(java.lang.Object obj)
java.util.List	getDiffs(RdfsElem other)
java.lang.String	getKind()
int	hashCode()
java.lang.String	toString()
java.lang.String	toStringLong()

Methods inherited from class `org.tanjakostic.jcleancim.experimental.builder.rdfs.RdfsElem`
equals, formatDiff, getAbout, getComment, getDiffs, getKind, getLabel, getModel, getName, getPackage, getSchemaLabel, hashCode, toString, toStringLong

Methods inherited from class `java.lang.Object`
clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods

getKind

public java.lang.String **getKind**()

Returns the string describing the kind of this element.

(continued from last page)

toString

`public java.lang.String toString()`

toStringLong

`public java.lang.String toStringLong()`

hashCode

`public int hashCode()`

Uses all instance fields except for _model.

equals

`public boolean equals(java.lang.Object obj)`

Uses all instance fields except for _model. We do track differences for relevant fields, but we ignore the difference (i.e., don't return false) in following two cases:

- _package field when dialects are different and this element is not an RdfsClass (RDF has info on package for classes only, so in case of RdfsClass we always compare normally).
- _comment field when dialects are different (with OWL, we have to generate attributes of datatype classes, as well as 4 CIM primitive classes, so we have no original comments).

getDiffs

`public final java.util.List getDiffs(RdfsElem other)`

Calculates differences between this instance and `other` and returns them in a list of Strings. While `RdfsElem.equals(Object)` method ignores some known differences between elements in different dialects, this method catches them all.

Implementation note: Subclasses that override this method are expected to first call super.runDiff().

org.tanjakostic.jcleancim.experimental.builder.rdfs
Class RdfsModel

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.experimental.builder.rdfs.RdfsModel
```

public final class **RdfsModel**
extends java.lang.Object

The model content is, at the start, reflecting the content of the parsed RDF/OWL schema, and at the end, after a number of validations and consistency checks, it contains the representation of the corrected UML CIM.

Constructor Summary

public	RdfsModel(java.lang.String name) Constructor.

Method Summary

void	build(XmlDocument rdfSchema) Traverses the loaded rdfSchema and stores all of its elements.
java.util.List	calcDiffClasses(RdfsModel other, boolean isDeep)
java.util.List	calcDiffEnumLiterals(RdfsModel other, boolean isDeep)
java.util.List	calcDiffPackages(RdfsModel other, boolean isDeep)
java.util.List	calcDiffProps(RdfsModel other, boolean isDeep)
java.util.List	calcDiffs(RdfsModel other)
void	clear() Clears the cache.
RdfsProperty	findProperty(java.lang.String prop) Returns the cached property for the given name, null if not found.
java.util.Map	getClasses() Returns all parsed classes.
java.util.Map	getDatatypeClasses()
java.lang.String	getDiffsAsCSV(RdfsModel other) Returns string containing all the differences in CSV format.
java.util.Map	getEnumClasses()
java.util.Map	getEnumLiterals() Returns all parsed enumeration literals.

java.util.Map	getFirstLevelClasses()
java.util.Map	getPackages() Returns all parsed packages.
java.util.Map	getPrimitiveClasses()
java.util.Map	getProps() Returns all parsed properties.
java.lang.String	getSchemaLabel() Returns the label of the schema for this hierarchy.
java.util.Map	getSubClasses()

Methods inherited from class java.lang.Object
clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Constructors

RdfsModel

```
public RdfsModel(java.lang.String name)
```

Constructor.

Methods

build

```
public void build(XmlDocument rdfSchema)
  throws CimSchemaException
```

Traverses the loaded `rdfSchema` and stores all of its elements. We first create individual elements, and on the way, their respective constructors try to fix those problems that are inherent to the kind of element itself (i.e., where there are no mutual dependencies between different element types).

Parameters:
`rdfSchema`

Throws:
`CimSchemaException` - if no schema, or more than one schema found.

findProperty

```
public RdfsProperty findProperty(java.lang.String prop)
```

Returns the cached property for the given name, null if not found.

Parameters:
`prop`

Returns:

(continued from last page)

cached property for the given name, null if not found.

calcDiffs

```
public java.util.List calcDiffs(RdfsModel other)
```

getDiffsAsCSV

```
public java.lang.String getDiffsAsCSV(RdfsModel other)
```

Returns string containing all the differences in CSV format.

Returns:
string containing all the differences in CSV format.

calcDiffPackages

```
public java.util.List calcDiffPackages(RdfsModel other,
         boolean isDeep)
```

calcDiffClasses

```
public java.util.List calcDiffClasses(RdfsModel other,
         boolean isDeep)
```

calcDiffProps

```
public java.util.List calcDiffProps(RdfsModel other,
         boolean isDeep)
```

calcDiffEnumLiterals

```
public java.util.List calcDiffEnumLiterals(RdfsModel other,
         boolean isDeep)
```

getSchemaLabel

```
public java.lang.String getSchemaLabel()
```

Returns the label of the schema for this hierarchy.

Returns:
the label of the schema for this hierarchy.

getPackages

```
public java.util.Map getPackages()
```

Returns all parsed packages.

(continued from last page)

Returns:
all parsed packages.

getClasses

`public java.util.Map getClasses()`

Returns all parsed classes.

Returns:
all parsed classes.

getEnumClasses

`public java.util.Map getEnumClasses()`

getDatatypeClasses

`public java.util.Map getDatatypeClasses()`

getPrimitiveClasses

`public java.util.Map getPrimitiveClasses()`

getSubClasses

`public java.util.Map getSubClasses()`

getFirstLevelClasses

`public java.util.Map getFirstLevelClasses()`

getProps

`public java.util.Map getProps()`

Returns all parsed properties.

Returns:
all parsed properties.

getEnumLiterals

`public java.util.Map getEnumLiterals()`

Returns all parsed enumeration literals.

(continued from last page)

Returns:
all parsed enumeration literals.

clear

```
public void clear()
```

Clears the cache. Use this method after you've got the class hierarchy, to release some memory that is not used anymore.

org.tanjakostic.jcleancim.experimental.builder.rdfs
Class RdfsPackage

```
java.lang.Object
  |-org.tanjakostic.jcleancim.experimental.builder.rdfs.RdfsElem
    |-org.tanjakostic.jcleancim.experimental.builder.rdfs.RdfsPackage
```

public final class **RdfsPackage**
extends RdfsElem

CIM RDF Schema element representing the UML package.

Fields inherited from class `org.tanjakostic.jcleancim.experimental.builder.rdfs.RdfsElem`
`invalidCharsPattern`

Method Summary

`java.lang.String`	`getKind()`

Methods inherited from class `org.tanjakostic.jcleancim.experimental.builder.rdfs.RdfsElem`
`equals`, `formatDiff`, `getAbout`, `getComment`, `getDiffs`, `getKind`, `getLabel`, `getModel`, `getName`, `getPackage`, `getSchemaLabel`, `hashCode`, `toString`, `toStringLong`

Methods inherited from class `java.lang.Object`
`clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait`

Methods

getKind

`public java.lang.String` **`getKind`**`()`

Returns the string describing the kind of this element.

org.tanjakostic.jcleancim.experimental.builder.rdfs
Class RdfsParser

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.experimental.builder.rdfs.RdfsParser
```

public class **RdfsParser**
extends java.lang.Object

Parses CIM RDF Schema file (created from CIM UML model with CIMTool).

Constructor Summary

public	RdfsParser(java.io.File rdfSchemaFile) Constructs an instance from the file with the CIM RDF Schema.

Method Summary

void	diffSchemas(java.io.File diffFile, RdfsModel otherModel) Saves the differences in the given file.
RdfsModel	getModel()

Methods inherited from class `java.lang.Object`
clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Constructors

RdfsParser

public **RdfsParser**(java.io.File rdfSchemaFile)

Constructs an instance from the file with the CIM RDF Schema.

Parameters:
`rdfSchemaFile` - input: the CIM RDF Schema file.

Throws:
CimSchemaException - if file cannot be found.

Methods

getModel

public RdfsModel **getModel**()

(continued from last page)

diffSchemas

```
public void diffSchemas(java.io.File diffFile,
         RdfsModel otherModel)
  throws java.io.IOException
```

Saves the differences in the given file.

Parameters:

`diffFile` - file where to store the differences.
`otherModel` - other model.

Throws:

`IOException`

org.tanjakostic.jcleancim.experimental.builder.rdfs
Class RdfsProperty

```
java.lang.Object
  |-org.tanjakostic.jcleancim.experimental.builder.rdfs.RdfsElem
    |-org.tanjakostic.jcleancim.experimental.builder.rdfs.RdfsProperty
```

public final class **RdfsProperty**
extends RdfsElem

CIM RDF Schema element representing the UML attribute or association end (role).

Nested Class Summary

class	RdfsProperty.Kind RdfsProperty.Kind

Fields inherited from class org.tanjakostic.jcleancim.experimental.builder.rdfs.RdfsElem

invalidCharsPattern

Constructor Summary

public	RdfsProperty(RdfsModel model, java.lang.String about, java.lang.String label, java.lang.String comment, java.lang.String pckage, boolean validateAbout, java.lang.String domain, RdfsProperty.Kind kind, java.lang.String dataType, java.lang.String range, java.lang.String invRoleName, java.lang.String multiplicity) Constructor.

Method Summary

boolean	equals(java.lang.Object obj) Uses all the instance fields, except for _kind.
java.lang.String	getDatatype()
java.util.List	getDiffs(RdfsElem other)
java.lang.String	getDomain()
java.lang.String	getInvRoleName()
java.lang.String	getKind()
java.lang.String	getMultiplicity()
java.lang.String	getNameAndMultiplicity()
java.lang.String	getRange()

int	hashCode() Uses all the instance fields, except for _kind.
boolean	isAssocEnd()
boolean	isDatatypeAttr()
boolean	isEnumAttr()
static boolean	isPrimitiveType(java.lang.String className)
boolean	isSimpleAttr()
java.lang.String	toString()
java.lang.String	toStringLong()

Methods inherited from class org.tanjakostic.jcleancim.experimental.builder.rdfs.RdfsElem

equals, formatDiff, getAbout, getComment, getDiffs, getKind, getLabel, getModel, getName, getPackage, getSchemaLabel, hashCode, toString, toStringLong

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Constructors

RdfsProperty

```
public RdfsProperty(RdfsModel model,
                    java.lang.String about,
                    java.lang.String label,
                    java.lang.String comment,
                    java.lang.String pckage,
                    boolean validateAbout,
                    java.lang.String domain,
                    RdfsProperty.Kind kind,
                    java.lang.String dataType,
                    java.lang.String range,
                    java.lang.String invRoleName,
                    java.lang.String multiplicity)
```

Constructor.

Parameters:

model
about
label
comment
pckage
validateAbout
domain
kind

(continued from last page)

`range`
`invRoleName` - null for attribute
`multiplicity` - empty string for attribute

Throws:
`CimSchemaException`

Methods

isAssocEnd

```
public boolean isAssocEnd()
```

isPrimitiveType

```
public static boolean isPrimitiveType(java.lang.String className)
```

isSimpleAttr

```
public boolean isSimpleAttr()
```

isDatatypeAttr

```
public boolean isDatatypeAttr()
```

isEnumAttr

```
public boolean isEnumAttr()
```

getDomain

```
public java.lang.String getDomain()
```

getRange

```
public java.lang.String getRange()
```

getDatatype

```
public java.lang.String getDatatype()
```

(continued from last page)

getInvRoleName

`public java.lang.String getInvRoleName()`

getMultiplicity

`public java.lang.String getMultiplicity()`

getNameAndMultiplicity

`public java.lang.String getNameAndMultiplicity()`

getKind

`public java.lang.String getKind()`

Returns the string describing the kind of this element.

toString

`public java.lang.String toString()`

toStringLong

`public java.lang.String toStringLong()`

hashCode

`public int hashCode()`

Uses all the instance fields, except for _kind.

equals

`public boolean equals(java.lang.Object obj)`

Uses all the instance fields, except for _kind. Also, if dialects are different, compares only upper limit in multiplicity.

getDiffs

`public final java.util.List getDiffs(RdfsElem other)`

Calculates differences between this instance and `other` and returns them in a list of Strings. While `RdfsElem.equals(Object)` method ignores some known differences between elements in different dialects, this method catches them all.

Implementation note: Subclasses that override this method are expected to first call super.runDiff().

org.tanjakostic.jcleancim.experimental.builder.rdfs
Class RdfsProperty.Kind

```
java.lang.Object
  |-java.lang.Enum
    |-org.tanjakostic.jcleancim.experimental.builder.rdfs.RdfsProperty.Kind
```

All Implemented Interfaces:
java.io.Serializable, java.lang.Comparable

---

public static final class **RdfsProperty.Kind**
extends java.lang.Enum

We need this one with OWL, since we cannot distinguish between dataType and range (OWL always uses range plus some other tags).

Field Summary

public static final	assocEnd
public static final	compoundAttr
public static final	datatypeAttr
public static final	enumAttr
public static final	simpleAttr
public static final	unknown

Method Summary

static RdfsProperty.Kind	valueOf(java.lang.String name)
static RdfsProperty.Kind[]	values()

Methods inherited from class `java.lang.Enum`

clone, compareTo, equals, finalize, getDeclaringClass, hashCode, name, ordinal, toString, valueOf

Methods inherited from class `java.lang.Object`

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface `java.lang.Comparable`

compareTo

Fields

simpleAttr

```
public static final
org.tanjakostic.jcleancim.experimental.builder.rdfs.RdfsProperty.Kind simpleAttr
```

datatypeAttr

```
public static final
org.tanjakostic.jcleancim.experimental.builder.rdfs.RdfsProperty.Kind datatypeAttr
```

enumAttr

```
public static final
org.tanjakostic.jcleancim.experimental.builder.rdfs.RdfsProperty.Kind enumAttr
```

compoundAttr

```
public static final
org.tanjakostic.jcleancim.experimental.builder.rdfs.RdfsProperty.Kind compoundAttr
```

assocEnd

```
public static final
org.tanjakostic.jcleancim.experimental.builder.rdfs.RdfsProperty.Kind assocEnd
```

unknown

```
public static final
org.tanjakostic.jcleancim.experimental.builder.rdfs.RdfsProperty.Kind unknown
```

Methods

values

```
public static RdfsProperty.Kind[] values()
```

valueOf

```
public static RdfsProperty.Kind valueOf(java.lang.String name)
```

(continued from last page)

org.tanjakostic.jcleancim.experimental.builder.rdfs
Class XmlAttribute

```
java.lang.Object
  |-org.tanjakostic.jcleancim.experimental.builder.rdfs.XmlTag
    |-org.tanjakostic.jcleancim.experimental.builder.rdfs.XmlAttribute
```

public class **XmlAttribute**
extends XmlTag

XML attributes used in the CIM RDF/OWL Schema.

Field Summary

public static final	resource applicable to rdfs:subClassOf, rdfs:domain, rdfs:range, rdf:type, ---

Method Summary

java.lang.String	getValue(org.w3c.dom.Element parent) Returns the value of this attribute on parent element, or null if the parent is null or does not have this attribute.

Methods inherited from class org.tanjakostic.jcleancim.experimental.builder.rdfs.XmlTag

getName, getQName, getURI, toString

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Fields

resource

public static final org.tanjakostic.jcleancim.experimental.builder.rdfs.XmlAttribute **resource**

applicable to rdfs:subClassOf, rdfs:domain, rdfs:range, rdf:type, ---

(OWL) applicable to owl:inverseOf, --- (and j.1:hasStereotype, owl:sameAs, ---)

(RDF) applicable to cims:inverseRoleName, --- cims:belongsToCategory, cims:dataType, cims:multiplicity

Methods

getValue

public java.lang.String **getValue**(org.w3c.dom.Element parent)

Returns the value of this attribute on parent element, or null if the parent is null or does not have this attribute.

(continued from last page)

Parameters:
`parent`

Returns:
value of this attribute on `parent` element

org.tanjakostic.jcleancim.experimental.builder.rdfs
Class XmlChildElement

```
java.lang.Object
  |-org.tanjakostic.jcleancim.experimental.builder.rdfs.XmlTag
    |-org.tanjakostic.jcleancim.experimental.builder.rdfs.XmlChildElement
```

public class **XmlChildElement**
extends XmlTag

XML child element (child of rdf:Description) used in the CIM RDF Schema.

Field Summary

public static final	belongsToCategory applicable to rdfs:Class; it's CIM package of the class
public static final	inverseRoleName applicable to association end (rdf:Description, with type=rdf:Property)
public static final	range applicable to association end or attribute having compound type
public static final	subClassOf applicable to (non-enum, non-dt, non-compound) class
public static final	type applicable to rdf:Description (attribute of enum class) to designate enum class

Method Summary

java.util.List	getAllOfThisKind(org.w3c.dom.Element parent) Returns all child elements of parent having this kind (tag).
static java.lang.String	getResourceName(org.w3c.dom.Element parent, XmlChildElement chiElemKind) Returns the fragment of the uri which is the resource attribute on child element chiElemKind of parent if existing, null otherwise.
static java.util.List	getResourceNames(org.w3c.dom.Element parent, XmlChildElement chiElemKind) Returns the fragments of the uris which are the resource attribute on all child elements chiElemKind of parent.
java.lang.String	getResourceValue(org.w3c.dom.Element parent) Returns value of resource attribute on single parent's child of this kind if it exists, null otherwise.
java.util.List	getResourceValues(org.w3c.dom.Element parent) Returns values of resource attribute on all parent's children of this kind if they exist, empty list otherwise.
java.lang.String	getText(org.w3c.dom.Element parent) Returns trimmed text contents of single parent's child of this kind (tag) if it exists, empty string otherwise.

static java.net.URI	getValidatedUri(java.lang.String uri) Returns validated URI for given string.

Methods inherited from class `org.tanjakostic.jcleancim.experimental.builder.rdfs.XmlTag`

getName, getQName, getURI, toString

Methods inherited from class `java.lang.Object`

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Fields

subClassOf

public static final org.tanjakostic.jcleancim.experimental.builder.rdfs.XmlChildElement **subClassOf**

applicable to (non-enum, non-dt, non-compound) class

type

public static final org.tanjakostic.jcleancim.experimental.builder.rdfs.XmlChildElement **type**

applicable to rdf:Description (attribute of enum class) to designate enum class

range

public static final org.tanjakostic.jcleancim.experimental.builder.rdfs.XmlChildElement **range**

applicable to association end or attribute having compound type

inverseRoleName

public static final org.tanjakostic.jcleancim.experimental.builder.rdfs.XmlChildElement **inverseRoleName**

applicable to association end (rdf:Description, with type=rdf:Property)

belongsToCategory

public static final org.tanjakostic.jcleancim.experimental.builder.rdfs.XmlChildElement **belongsToCategory**

applicable to rdfs:Class; it's CIM package of the class

Methods

getText

public java.lang.String **getText**(org.w3c.dom.Element parent)

Returns trimmed text contents of single `parent`'s child of this kind (tag) if it exists, empty string otherwise.

(continued from last page)

Parameters:
`parent`

Returns:
text if element exists, empty string otherwise.

getAllOfThisKind

```
public java.util.List getAllOfThisKind(org.w3c.dom.Element parent)
```

Returns all child elements of `parent` having this kind (tag).

getResourceValue

```
public java.lang.String getResourceValue(org.w3c.dom.Element parent)
```

Returns value of resource attribute on single `parent`'s child of this kind if it exists, null otherwise.

Parameters:
`parent`

Returns:
value of resource attribute on single `parent`'s child of this kind if it exists, null otherwise.

getResourceValues

```
public java.util.List getResourceValues(org.w3c.dom.Element parent)
```

Returns values of resource attribute on all `parent`'s children of this kind if they exist, empty list otherwise.

getValidatedUri

```
public static java.net.URI getValidatedUri(java.lang.String uri)
  throws CimSchemaException
```

Returns validated URI for given string.

Parameters:
`uri`

Returns:
validated URI for given string.

Throws:
`CimSchemaException` - if given string is syntactically invalid URI.

getResourceName

```
public static java.lang.String getResourceName(org.w3c.dom.Element parent,
          XmlChildElement chiElemKind)
```

Returns the fragment of the uri which is the resource attribute on child element `chiElemKind` of `parent` if existing, null otherwise. Use this method only when sure that the URI has already been validated.

Parameters:
`parent`
`chiElemKind`

(continued from last page)

Returns:

name of the resource.

getResourceNames

```
public static java.util.List getResourceNames(org.w3c.dom.Element parent,
          XmlChildElement chiElemKind)
```

Returns the fragments of the uris which are the resource attribute on all child elements `chiElemKind` of `parent`. This is the helper for those child element kinds that can be multiple for a parent (e.g., type, subClassOf).

Parameters:

`parent`

`chiElemKind`

Returns:

the names of the resources.

org.tanjakostic.jcleancim.experimental.builder.rdfs
Class XmlDocument

```
java.lang.Object
  |-org.tanjakostic.jcleancim.xml.WellformedDOM
    |-org.tanjakostic.jcleancim.experimental.builder.rdfs.XmlDocument
```

public class **XmlDocument**
extends WellformedDOM

Reads RDF Schema and provides classified DOM elements to be used when building model.

Constructor Summary

public	`XmlDocument(java.io.File file)` Constructs the DOM Document from CIM RDF Schema file, determines the dialect, fixes URIs if they end with white space, and classifies DOM elements by moving them from DOM Document root into internal maps.
public	`XmlDocument(java.lang.String xml)` Same as `XmlDocument(File)`, but constructs the DOM document from xml string.

Method Summary

java.util.Map	`getClasses()` Returns the CIM UML classes (including datatypes, enums and compounds) in this document.
java.util.Map	`getDuplicates()` Returns elements in this document that have same name.
int	`getElementCount()` Returns the total number of top elements in this document.
java.util.Map	`getEnumLiterals()` Returns the CIM UML enumeration literals in this document.
java.util.Map	`getPackages()` Returns the CIM UML packages in this document.
java.util.Map	`getProperties()` Returns the CIM UML attributes and association ends in this document.
java.util.Map	`getUnclassifiedTopElements()` Returns the CIM UML enumeration literals in this document.

Methods inherited from class `org.tanjakostic.jcleancim.xml.WellformedDOM`

`asInputStream`, `asXmlString`, `getDocument`, `getFile`, `getNsCache`

Methods inherited from class `java.lang.Object`

`clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait`

Constructors

XmlDocument

`public XmlDocument(java.io.File file)`

Constructs the DOM Document from CIM RDF Schema file, determines the dialect, fixes URIs if they end with white space, and classifies DOM elements by moving them from DOM Document root into internal maps.

XmlDocument

`public XmlDocument(java.lang.String xml)`

Same as `XmlDocument(File)`, but constructs the DOM document from xml string.

Methods

getElementCount

`public int getElementCount()`

Returns the total number of top elements in this document.

Returns:
total number of top elements in this document.

getDuplicates

`public java.util.Map getDuplicates()`

Returns elements in this document that have same name.

Returns:
elements in this document that have same name.

getPackages

`public java.util.Map getPackages()`

Returns the CIM UML packages in this document.

getClasses

`public java.util.Map getClasses()`

Returns the CIM UML classes (including datatypes, enums and compounds) in this document.

getProperties

`public java.util.Map getProperties()`

Returns the CIM UML attributes and association ends in this document.

(continued from last page)

getEnumLiterals

`public java.util.Map` **`getEnumLiterals`**`()`

Returns the CIM UML enumeration literals in this document.

getUnclassifiedTopElements

`public java.util.Map` **`getUnclassifiedTopElements`**`()`

Returns the CIM UML enumeration literals in this document.

org.tanjakostic.jcleancim.experimental.builder.rdfs
Class XmlElement

```
java.lang.Object
  |-org.tanjakostic.jcleancim.experimental.builder.rdfs.XmlTag
    |-org.tanjakostic.jcleancim.experimental.builder.rdfs.XmlElement
```

public class **XmlElement**
extends XmlTag

XML element, direct child of the RDF schema. All methods are static and take as argument DOM Element.

Method Summary

static java.util.List	deduceInverseRoleNames(org.w3c.dom.Element parent) Tries to deduce the name of the UML inverse association end from different child elements, to suit different RDF dialects.
static java.lang.String	deducePackageName(org.w3c.dom.Element elem) Tries to deduce the name of the UML package from different child elements.
static java.util.List	deduceRangeNames(org.w3c.dom.Element elem) Tries to deduce the name of the UML class that is range for a property from different child elements, to suit different RDF dialects.
static java.util.List	getChildrenWithResource(org.w3c.dom.Element parent, XmlNs ns) Returns all child elements of parent that have 'resource' attribute.
static boolean	isClass(org.w3c.dom.Element parent)
static boolean	isCompoundClass(org.w3c.dom.Element parent) Returns whether parent is a UML compound class.
static boolean	isDatatypeClass(org.w3c.dom.Element parent) Returns whether parent is a UML datatype class.
static boolean	isEnumClass(org.w3c.dom.Element parent) Returns whether parent is a UML enumerated class.
static boolean	isEnumLiteral(org.w3c.dom.Element parent)
static boolean	isPackage(org.w3c.dom.Element parent)
static boolean	isPrimitiveClass(org.w3c.dom.Element parent) Returns whether parent is a UML enumerated class.
static boolean	isProperty(org.w3c.dom.Element parent)
static void	normaliseToRdf(XmlNs ns, org.w3c.dom.Element parent) Peforms several potential modifications:

Methods inherited from class org.tanjakostic.jcleancim.experimental.builder.rdfs.XmlTag

getName, getQName, getURI, toString

Methods inherited from class `java.lang.Object`

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods

getChildrenWithResource

```
public static java.util.List getChildrenWithResource(org.w3c.dom.Element parent,
         XmlNs ns)
```

Returns all child elements of `parent` that have 'resource' attribute.

Parameters:
`parent`

Returns:
all child elements of `parent` that have resource attribute.

normaliseToRdf

```
public static void normaliseToRdf(XmlNs ns,
         org.w3c.dom.Element parent)
```

Peforms several potential modifications:

- removes "Package_" substring from 'about' attribute of `parent` and 'resource' attribute of its child, when applicable
- prepends the URI of `ns` to 'about' attribute of `parent` and 'resource' attribute of all its children, if they use default namespace syntax (i.e., start with "#")
- replaces XML primitive types in children with CIM primitive types

Parameters:
`ns` - namespace, used to prepend its URI.
`parent`

isPackage

```
public static boolean isPackage(org.w3c.dom.Element parent)
```

isClass

```
public static boolean isClass(org.w3c.dom.Element parent)
```

isProperty

```
public static boolean isProperty(org.w3c.dom.Element parent)
```

(continued from last page)

isEnumLiteral

`public static boolean **isEnumLiteral**(org.w3c.dom.Element parent)`

isPrimitiveClass

`public static boolean **isPrimitiveClass**(org.w3c.dom.Element parent)`

Returns whether `parent` is a UML enumerated class.

isEnumClass

`public static boolean **isEnumClass**(org.w3c.dom.Element parent)`

Returns whether `parent` is a UML enumerated class.

isCompoundClass

`public static boolean **isCompoundClass**(org.w3c.dom.Element parent)`

Returns whether `parent` is a UML compound class.

isDatatypeClass

`public static boolean **isDatatypeClass**(org.w3c.dom.Element parent)`

Returns whether `parent` is a UML datatype class.

deduceInverseRoleNames

`public static java.util.List **deduceInverseRoleNames**(org.w3c.dom.Element parent)`

Tries to deduce the name of the UML inverse association end from different child elements, to suit different RDF dialects. For ill-defined model, values may be multiple, thus we return a list of names. If none is found, returns empty list.

Parameters:
`parent` - Parent element under whose child elements to search for inverse association end name.

Returns:
name of the UML inverse association end.

deduceRangeNames

`public static java.util.List **deduceRangeNames**(org.w3c.dom.Element elem)`

Tries to deduce the name of the UML class that is range for a property from different child elements, to suit different RDF dialects. For ill-defined model, values may be multiple, thus we return a list of names. If none is found, returns empty list.

Parameters:
`elem` - Parent element under whose child elements to search for range name.

Returns:
name of the UML class that is range for a property.

(continued from last page)

deducePackageName

```
public static java.lang.String deducePackageName(org.w3c.dom.Element elem)
```

Tries to deduce the name of the UML package from different child elements. If none is found, returns null.

Parameters:

`elem` - Parent element under whose child elements to search for package name.

Returns:

name of the UML package

org.tanjakostic.jcleancim.experimental.builder.rdfs
Class XmlNamespace

```
java.lang.Object
  |-org.tanjakostic.jcleancim.xml.XmlNs
    |-org.tanjakostic.jcleancim.experimental.builder.rdfs.XmlNamespace
```

public class **XmlNamespace**
extends XmlNs

Common namespaces found in CIM RDF/OWL Schema.

Field Summary

public static final	cim
public static final	cims
public static	dc
public static	msg
public static	owl
public static final	rdf
public static final	rdfs
public static final	uml
public static final	xml
public static final	xsd (OWL) Note that the namespace is not read from the schema, but is used to replace CIM primitive types (Float, String, Integer, Boolean, Decimal, Date, Time, DateTime, Duration) with a child element such as e.g., `<rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>`.

Fields inherited from class org.tanjakostic.jcleancim.xml.XmlNs

FRAG_SEP, xsi

Constructor Summary

public	XmlNamespace(java.lang.String prefix, java.lang.String uri)
public	XmlNamespace(java.lang.String prefix, java.lang.String uri, NamespaceCache cache)

Methods inherited from class `org.tanjakostic.jcleancim.xml.XmlNs`
`getPrefix`, `getUri`, `getUriWithoutFragmentSeparator`, `qName`, `toString`

Methods inherited from class `java.lang.Object`
`clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait`

Fields

cim

`public static final org.tanjakostic.jcleancim.xml.XmlNs` **`cim`**

rdfs

`public static final org.tanjakostic.jcleancim.xml.XmlNs` **`rdfs`**

rdf

`public static final org.tanjakostic.jcleancim.xml.XmlNs` **`rdf`**

xml

`public static final org.tanjakostic.jcleancim.xml.XmlNs` **`xml`**

xsd

`public static final org.tanjakostic.jcleancim.xml.XmlNs` **`xsd`**

(OWL) Note that the namespace is not read from the schema, but is used to replace CIM primitive types (Float, String, Integer, Boolean, Decimal, Date, Time, DateTime, Duration) with a child element such as e.g., `<rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>`.

owl

`public static org.tanjakostic.jcleancim.xml.XmlNs` **`owl`**

cims

`public static final org.tanjakostic.jcleancim.xml.XmlNs` **`cims`**

(continued from last page)

uml

```
public static final org.tanjakostic.jcleancim.xml.XmlNs uml
```

msg

```
public static org.tanjakostic.jcleancim.xml.XmlNs msg
```

dc

```
public static org.tanjakostic.jcleancim.xml.XmlNs dc
```

Constructors

XmlNamespace

```
public XmlNamespace(java.lang.String prefix,
                    java.lang.String uri)
```

XmlNamespace

```
public XmlNamespace(java.lang.String prefix,
                    java.lang.String uri,
                    NamespaceCache cache)
```

org.tanjakostic.jcleancim.experimental.builder.rdfs
Class XmlResourceValue

```
java.lang.Object
  |-org.tanjakostic.jcleancim.experimental.builder.rdfs.XmlTag
    |-org.tanjakostic.jcleancim.experimental.builder.rdfs.XmlResourceValue
```

public class **XmlResourceValue**
extends XmlTag

Values used for resource attribute (), used in RDF and OWL dialects.

Field Summary

public static final	attribute
public static final	Class UML class with rdf:type rdf:resource="http://www.w3.org/2000/01/rdf-schema#Class"
public static final	Datatype
public static final	enumeration
public static final	Package
public static final	Property

Method Summary

static java.lang.String	getCimPrimitiveClassResourceValue(java.lang.String xsUri)
static java.util.Collection	getCimPrimitiveClassResourceValues()

Methods inherited from class org.tanjakostic.jcleancim.experimental.builder.rdfs.XmlTag

getName, getQName, getURI, toString

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Fields

(continued from last page)

Package

```
public static final
org.tanjakostic.jcleancim.experimental.builder.rdfs.XmlResourceValue Package
```

Datatype

```
public static final
org.tanjakostic.jcleancim.experimental.builder.rdfs.XmlResourceValue Datatype
```

Property

```
public static final
org.tanjakostic.jcleancim.experimental.builder.rdfs.XmlResourceValue Property
```

Class

```
public static final
org.tanjakostic.jcleancim.experimental.builder.rdfs.XmlResourceValue Class
```

UML class with rdf:type rdf:resource="http://www.w3.org/2000/01/rdf-schema#Class"

enumeration

```
public static final
org.tanjakostic.jcleancim.experimental.builder.rdfs.XmlResourceValue enumeration
```

attribute

```
public static final
org.tanjakostic.jcleancim.experimental.builder.rdfs.XmlResourceValue attribute
```

Methods

getCimPrimitiveClassResourceValue

```
public static java.lang.String getCimPrimitiveClassResourceValue(java.lang.String
xsUri)
```

getCimPrimitiveClassResourceValues

```
public static java.util.Collection getCimPrimitiveClassResourceValues()
```

org.tanjakostic.jcleancim.experimental.builder.rdfs
Class XmlTag

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.experimental.builder.rdfs.XmlTag
```

Direct Known Subclasses:
XmlAttribute, XmlChildElement, XmlElement, XmlResourceValue

---

public abstract class **XmlTag**
extends java.lang.Object

Common implementation for various tags that appear in CIM RDF Schema.

Constructor Summary

protected	XmlTag(XmlNs ns, java.lang.String name) Constructor.

Method Summary

java.lang.String	getName() Returns the name of this tag (e.g., about).
java.lang.String	getQName() Returns the qualified name of this tag (e.g., rdf:about).
java.lang.String	getURI() Returns the URI of this tag (e.g., http://...#about).
java.lang.String	toString() Returns qualified name of this tag as string (e.g., rdf:about).

Methods inherited from class `java.lang.Object`
clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

---

Constructors

XmlTag

```
protected XmlTag(XmlNs ns,
                 java.lang.String name)
```

Constructor.

Parameters:
ns
name

(continued from last page)

Methods

getQName

`protected final java.lang.String` **`getQName`**`()`

Returns the qualified name of this tag (e.g., rdf:about).

getName

`public final java.lang.String` **`getName`**`()`

Returns the name of this tag (e.g., about).

getURI

`public final java.lang.String` **`getURI`**`()`

Returns the URI of this tag (e.g., http://...#about).

toString

`public java.lang.String` **`toString`**`()`

Returns qualified name of this tag as string (e.g., rdf:about).

Package

org.tanjakostic.jcleancim.experimental.builder.xsd

The package contains the `ModelBuilderFromProfiles`, which can parse: the CIM profiles in org.tanjakostic.jcleancim.common.Config#XSD_EXT format, as generated with CIMTool.

The result of parsing is contained in `Profile` and its related classes.

Implementation note: At present, CIMTool does not produce usable RDFS mapping for profiles targeted at XSD syntax (i.e. using local types and no inheritance). Therefore, XSD syntax, although extremely complex, was the only choice available at the moment of this writing. If we can get propoper RDFS mappings for this kind of complex profiles, that will definitely be the preferred syntax.

org.tanjakostic.jcleancim.experimental.builder.xsd
Class ModelBuilderFromProfiles

```
java.lang.Object
  |-org.tanjakostic.jcleancim.builder.AbstractModelBuilder
    |-org.tanjakostic.jcleancim.experimental.builder.xsd.ModelBuilderFromProfiles
```

All Implemented Interfaces:
ModelBuilder

public class **ModelBuilderFromProfiles**
extends AbstractModelBuilder

Builds in-memory model from profile files, as specified by configuration. Each profile is read from one profile file and represented in the in-memory model as one model package.

Field Summary

public static final	MODEL_PACKAGE_NAME Value: **TC57CIMProfiles**

Constructor Summary

public	ModelBuilderFromProfiles(Config cfg)

Method Summary

UmlModel	build()
DiagramExporter	createDiagramExporter()
XMIExporter	createXMIExporter()
java.util.Map	getProfiles() Returns profiles per owner.

Methods inherited from class org.tanjakostic.jcleancim.builder.AbstractModelBuilder

build, createDiagramExporter, createXMIExporter, getCfg, getDiagramExporter, getXMIExporter

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.builder.ModelBuilder

build, getCfg, getDiagramExporter, getXMIExporter

Fields

MODEL_PACKAGE_NAME

`public static final java.lang.String` **`MODEL_PACKAGE_NAME`**

Constant value: **`TC57CIMProfiles`**

Constructors

ModelBuilderFromProfiles

`public` **`ModelBuilderFromProfiles`**`(Config cfg)`

Methods

getProfiles

`public java.util.Map` **`getProfiles`**`()`

Returns profiles per owner.

build

`public UmlModel` **`build`**`()`

createDiagramExporter

`protected DiagramExporter` **`createDiagramExporter`**`()`

Creates exporter of UML diagrams where applicable (otherwise, can be just a stub).

createXMIExporter

`protected XMIExporter` **`createXMIExporter`**`()`

Returns exporter to XMI where applicable (otherwise, can be just a stub).

org.tanjakostic.jcleancim.experimental.builder.xsd
Class Profile

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.experimental.builder.xsd.Profile
```

public class **Profile**
extends java.lang.Object

Class that parses and analyses an .xsd profile and stores its content in-memory. It is then used to build one package in the regular in-memory model.

Implementation note: I'm using dumb and trivial XPath expressions, which might not be optimal...

FIXME: Consider using new `XmlSchemaDOM`!

Field Summary

public static final	FRAGMENT_SEPARATOR Value: **35**
public static final	TARGET_NS_PREFIX Value: **m**
public static final	XPATH_ATTR_AND_ASSOC_ENDS Value: **//@sawsdl:modelReference/parent::xs:element**
public static final	XPATH_CHOICE_ASSOC_ENDS Value: **//@sawsdl:modelReference/parent::xs:choice**
public static final	XPATH_CLASSES_AND_COMPOUNDS Value: **//@sawsdl:modelReference/parent::xs:complexType**
public static final	XPATH_LITERALS Value: **//xs:restriction/child::xs:enumeration**
public static final	XPATH_LOCAL_DOC Value: **xs:annotation/xs:documentation**

Constructor Summary

public	Profile(Config cfg, java.io.File xsdFile) Constructor.

Method Summary

java.util.Map	getAttributesAndAssocEnds()

java.util.Map	getCimNamespaces() Returns namespace information for all model references found in the profile, with prefix as key, and URI as value.
java.util.Collection	getCimPrimitives()
java.util.Map	getClassesAndCompounds()
java.util.Map	getDatatypes()
java.util.Map	getEnums()
java.lang.String	getEnvelopeName() Returns envelope name (in instance file, this will be the root element).
java.util.Map	getLiterals()
java.lang.String	getName() Returns profile name (deduced from the file name, without extension); for envelope name, use getEnvelopeName().
java.util.List	getNamespaces()
static java.util.Map	getPrimitives()
java.util.List	getSubdirNames() Returns list of names, deduced from the profile file path.
java.util.Map	getTargetNamespace() Returns the target namespace information, with prefix #TARGET_NS_PREFIX as key, and URI as value.
java.util.Set	getUnclassifieds()
java.io.File	getXsdFile() Returns the file that has been used for initalisation, null if the profile has been created from a string.
boolean	hasInconsistentEnvelopeName() Returns whether this profile follows the convention to have the envelope name same as the profile name.
boolean	hasInconsistentNamespace() Returns whether this profile follows the convention to have the namespace end with the profile name (followed by the URI fragment separator #FRAGMENT_SEPARATOR).
java.lang.String	toString()

Methods inherited from class `java.lang.Object`
`clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait`

Fields

(continued from last page)

TARGET_NS_PREFIX

```
public static final java.lang.String TARGET_NS_PREFIX
```

Constant value: **m**

FRAGMENT_SEPARATOR

```
public static final char FRAGMENT_SEPARATOR
```

Constant value: **35**

XPATH_LOCAL_DOC

```
public static final java.lang.String XPATH_LOCAL_DOC
```

Constant value: **xs:annotation/xs:documentation**

XPATH_CLASSES_AND_COMPOUNDS

```
public static final java.lang.String XPATH_CLASSES_AND_COMPOUNDS
```

Constant value: **//@sawsdl:modelReference/parent::xs:complexType**

XPATH_ATTR_AND_ASSOC_ENDS

```
public static final java.lang.String XPATH_ATTR_AND_ASSOC_ENDS
```

Constant value: **//@sawsdl:modelReference/parent::xs:element**

XPATH_CHOICE_ASSOC_ENDS

```
public static final java.lang.String XPATH_CHOICE_ASSOC_ENDS
```

Constant value: **//@sawsdl:modelReference/parent::xs:choice**

XPATH_LITERALS

```
public static final java.lang.String XPATH_LITERALS
```

Constant value: **//xs:restriction/child::xs:enumeration**

Constructors

Profile

```
public Profile(Config cfg,
               java.io.File xsdFile)
```

Constructor.

TODO: ctor from XmlString

(continued from last page)

Parameters:
`cfg`
`xsdFile`

Methods

getPrimitives

`public static java.util.Map **getPrimitives**()`

getSubdirNames

`public java.util.List **getSubdirNames**()`

Returns list of names, deduced from the profile file path. The first element is the name of the `OwningWg` for this profile, the remaining names correspond to the names of `UmlPackage`-s to be created recursiverly.

getName

`public java.lang.String **getName**()`

Returns profile name (deduced from the file name, without extension); for envelope name, use `getEnvelopeName()`.

getEnvelopeName

`public java.lang.String **getEnvelopeName**()`

Returns envelope name (in instance file, this will be the root element).

getXsdFile

`public java.io.File **getXsdFile**()`

Returns the file that has been used for initalisation, null if the profile has been created from a string.

getTargetNamespace

`public java.util.Map **getTargetNamespace**()`

Returns the target namespace information, with prefix #TARGET_NS_PREFIX as key, and URI as value.

getNamespaces

`public java.util.List **getNamespaces**()`

getCimNamespaces

`public java.util.Map **getCimNamespaces**()`

Returns namespace information for all model references found in the profile, with prefix as key, and URI as value.

(continued from last page)

getClassesAndCompounds

`public java.util.Map getClassesAndCompounds()`

getCimPrimitives

`public java.util.Collection getCimPrimitives()`

getEnums

`public java.util.Map getEnums()`

getDatatypes

`public java.util.Map getDatatypes()`

getAttributesAndAssocEnds

`public java.util.Map getAttributesAndAssocEnds()`

getLiterals

`public java.util.Map getLiterals()`

getUnclassifieds

`public java.util.Set getUnclassifieds()`

hasInconsistentEnvelopeName

`public boolean hasInconsistentEnvelopeName()`

Returns whether this profile follows the convention to have the envelope name same as the profile name.

hasInconsistentNamespace

`public boolean hasInconsistentNamespace()`

Returns whether this profile follows the convention to have the namespace end with the profile name (followed by the URI fragment separator #FRAGMENT_SEPARATOR).

toString

`public java.lang.String toString()`

(continued from last page)

Package
org.tanjakostic.jcleancim.model

Classes being created by a builder or with the API (code) to hold the in-memory UML model.

Note that the EA API is terribly slow, and that is why we do heavy caching of everything that we read from the EA file. Afterwards, except for updating and pasting diagrams to clipboard (for doc generation), we are completely detached from EA and work with these classes in-memory.

Important classes and interfaces are:

- `UmlObject` - interface defining methods all UML elements in our model should implement.
- `UmlKind` - interface implemented by various *Kind enumerations, to allow for consistent displaying of kind/category/type information. Note that we could have designed and implemented subclasses of basic UML elements to reflect the categorisation, but it would have lead to proliferation of classes here for not a big deal of required functionality. Also, the resulting API would have likely been more complex to use, so we just sticked to this simple solution for the moment.
- `AbstractUmlObject` - abstract class implementing some of the methods of `UmlObject`, and from which most of UML elements in our model inherit. It also provides a couple of utility static methods that handle collections of `UmlObject`.
- `UmlObjectData` - value object holding attributes common to all `UmlObject`-s, used as instance variable in `AbstractUmlObject`. This makes it easier to populate the instances on creation, by avoiding a big number of parameters to constructors of concrete `UmlObject`-s.
- `UmlModel` - class that holds the configuration `Config` and all the concrete elements of the model. An instance of `UmlModel` can be populated by a builder or simply through the API (with the explicit code, like in tests). Elements of the UML model are arranged in hierarchies (package, subpackage...) starting from model packages (`UmlPackage.Kind.MODEL`). This class also internaly caches the major UML elements in hash maps (per UUID as string), to allow for fast searches without using `instanceof` operator.
- various Uml* classes, most of them inheriting from `AbstractUmlObject` and implementing `UmlObject`.
- `VersionInfo` - version information, as read from version classes expected to be found in top packages.

TODO:

org.tanjakostic.jcleancim.model
Class AbstractUmlObject

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.model.AbstractUmlObject
```

All Implemented Interfaces:
UmlObject

Direct Known Subclasses:
UmlAssociation, UmlAssociationEnd, UmlAttribute, UmlConstraint, UmlDependency, UmlDiagram, UmlOperation, UmlParameter, UmlSkipped, UmlStructure

---

public abstract class **AbstractUmlObject**
extends java.lang.Object
implements UmlObject

Common implementation of several methods and static utility methods for manipulating collections of `UmlObject`-s.

Implementation note: Uses composition with `UmlObjectData` for all the fields that are initialised in the constructor, and in `toShortString(boolean, String, boolean)` relies on abstract methods, to be implemented by concrete subtypes.

Field Summary

public static final	`CLASS_SEPARATOR` Class separator, for qualified names. Value: **.**
public static final	`NULL_OBJ_NAME` Value: **null**
public static final	`PACKAGE_SEPARATOR` Package separator, for qualified names. Value: **::**

Constructor Summary

protected	`AbstractUmlObject(UmlObjectData objData)` Constructor.

Method Summary

static java.util.List	`addDeprecAndInf(UmlObject o)` Returns potentially empty list of deprecation and informative qualifiers for o from its deprecation and informative status, not from its stereotypes.
java.lang.String	`addTaggedValue(java.lang.String name, java.lang.String value)`
static void	`appendRemainingCustomStereotypes(java.util.List tokens, UmlObject o, java.util.Map builtins)` Extends tokens with custom (=non-built-in) stereotypes that are not already contained in tokens.

static java.util.Map	classifyPerScope(java.util.Collection objects, java.util.EnumSet scope) Returns map of objects indexed per scope; skips null objects.
static java.util.Map	classifyPerScopePerTag(java.util.Map tags, java.util.EnumSet scope) Returns map of objects indexed per scope, then per tag name tags; skips null objects.
static java.util.Map	classifyPerTag(java.util.Map tags, java.util.EnumSet scope) Returns restricted map with objects that have given scope.
static MapOfCollections	collectDuplicateDescriptions(java.util.Collection objects) Returns those objects that have the same description (trimmed), indexed by that description; skips null objects.
static MapOfCollections	collectDuplicateNames(java.util.Collection objects) Returns those objects that have the same name, indexed by name; skips null objects.
static java.util.List	collectForScope(java.util.Collection objects, java.util.EnumSet scope) Returns list of objects that belong to a scope; skips null objects.
static java.util.List	collectNames(java.util.Collection objects) Returns list of names.
static java.util.List	collectQNames(java.util.Collection objects, boolean includeOwner) Returns list of qualified names, with prepended owner if includeOwner=true.
static java.util.Set	findAllForName(java.util.Collection objects, java.lang.String name) Returns set of objects with given name; skips null objects.
static UmlObject	findWithSameUuidAndLog(org.apache.log4j.Level level, UmlObject asker, java.util.Collection objects, java.lang.String uuid) Returns the object with uuid found in objects and logs the message with level; returns null otherwise.
java.lang.String	getAlias()
TextDescription	getDescription()
TextDescription	getHtmlDescription()
java.lang.Integer	getId()
abstract UmlKind	getKind()
java.lang.String	getName()
abstract Nature	getNature()
abstract OwningWg	getOwner()
abstract java.lang.String	getQualifiedName()
java.lang.String	getSince()
UmlStereotype	getStereotype()

java.util.Map	getTaggedValues()
java.util.Set	getUnallowedTagNames()
java.lang.String	getUuid()
UmlVisibility	getVisibility()
Namespace	initFromTags()
boolean	isDeprecated()
boolean	isInformative()
static void	saveTags(UmlObject o, java.util.Map destination) Stores object indexed by all of its tag names.
java.lang.String	toShortString(boolean includeId, boolean isNameQualified)
java.lang.String	toShortString(boolean includeId, java.lang.String qualifier, boolean isNameQualified) Similar to toShortString(boolean, boolean), but allows to specify a qualifier.
void	validateTag(java.lang.String name, java.lang.String value) Subclasses should overwride this method in case some validation about the tagged value is needed before adding it.

Methods inherited from class `java.lang.Object`

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface `org.tanjakostic.jcleancim.model.UmlObject`

addTaggedValue, getAlias, getDescription, getHtmlDescription, getId, getKind, getName, getNamespace, getNature, getOwner, getPredefinedTagNames, getQualifiedName, getSince, getStereotype, getTaggedValues, getUnallowedTagNames, getUuid, getVisibility, isDeprecated, isInformative, toShortString

Fields

NULL_OBJ_NAME

public static final java.lang.String **NULL_OBJ_NAME**

Constant value: **null**

PACKAGE_SEPARATOR

public static final java.lang.String **PACKAGE_SEPARATOR**

Package separator, for qualified names.

(continued from last page)

Constant value: `::`

CLASS_SEPARATOR

`public static final java.lang.String` **`CLASS_SEPARATOR`**

Class separator, for qualified names.
Constant value: `.`

Constructors

AbstractUmlObject

`protected` **`AbstractUmlObject`**`(UmlObjectData objData)`

Constructor.

Parameters:
`objData`

Methods

collectNames

`public static java.util.List` **`collectNames`**`(java.util.Collection objects)`

Returns list of names. For a null object in `objects`, returns the name `NULL_OBJ_NAME`.

collectQNames

`public static java.util.List` **`collectQNames`**`(java.util.Collection objects, boolean includeOwner)`

Returns list of qualified names, with prepended owner if `includeOwner=true`. For a null object in `objects`, returns the name `NULL_OBJ_NAME`.

collectDuplicateNames

`public static MapOfCollections` **`collectDuplicateNames`**`(java.util.Collection objects)`

Returns those `objects` that have the same name, indexed by name; skips null objects.

findAllForName

`public static java.util.Set` **`findAllForName`**`(java.util.Collection objects, java.lang.String name)`

Returns set of objects with given name; skips null objects.

collectDuplicateDescriptions

`public static MapOfCollections` **`collectDuplicateDescriptions`**`(java.util.Collection objects)`

Returns those `objects` that have the same description (trimmed), indexed by that description; skips null objects.

(continued from last page)

classifyPerScope

```
public static java.util.Map classifyPerScope(java.util.Collection objects,
         java.util.EnumSet scope)
```

Returns map of objects indexed per `scope`; skips null objects. For a simple list, use `collectForScope(Collection, EnumSet)`.

collectForScope

```
public static java.util.List collectForScope(java.util.Collection objects,
         java.util.EnumSet scope)
```

Returns list of objects that belong to a `scope`; skips null objects. For a map indexed per scope, use `classifyPerScope(Collection, EnumSet)`.

classifyPerScopePerTag

```
public static java.util.Map classifyPerScopePerTag(java.util.Map tags,
         java.util.EnumSet scope)
```

Returns map of objects indexed per `scope`, then per tag name `tags`; skips null objects. For a simple map indexed per tag name only, use `classifyPerTag(Map, EnumSet)`.

classifyPerTag

```
public static java.util.Map classifyPerTag(java.util.Map tags,
         java.util.EnumSet scope)
```

Returns restricted map with objects that have given scope.

saveTags

```
public static void saveTags(UmlObject o,
         java.util.Map destination)
```

Stores object indexed by all of its tag names.

findWithSameUuidAndLog

```
public static UmlObject findWithSameUuidAndLog(org.apache.log4j.Level level,
         UmlObject asker,
         java.util.Collection objects,
         java.lang.String uuid)
```

Returns the object with `uuid` found in `objects` and logs the message with `level`; returns null otherwise.

addDeprecAndInf

```
public static java.util.List addDeprecAndInf(UmlObject o)
```

Returns potentially empty list of deprecation and informative qualifiers for `o` from its deprecation and informative status, not from its stereotypes. Namely, although we do have and here use these stereotypes as tokens, the deprecation and informative status for an object are not exclusively determined from object's stereotype, but are often derived (e.g., content of deprecated class or package gets also deprecated, even without every attribute and relation having deprecated stereotype).

(continued from last page)

appendRemainingCustomStereotypes

```
public static void appendRemainingCustomStereotypes(java.util.List tokens,
         UmlObject o,
         java.util.Map builtins)
```

Extends `tokens with custom (=non-built-in) stereotypes that are not already contained in` tokens. This is useful not only to collect custom stereotype tokens, but also when you have a stereotype token be built-in for one object nature, and custom for another.

getId

```
public final java.lang.Integer getId()
```

getUuid

```
public java.lang.String getUuid()
```

getSince

```
public java.lang.String getSince()
```

getOwner

```
public abstract OwningWg getOwner()
```

initFromTags

```
protected Namespace initFromTags()
```

getNature

```
public abstract Nature getNature()
```

isInformative

```
public boolean isInformative()
```

This default implementation returns true if objects steretypes include UmlStereotype#INFORMATIVE. If there are additional criteria for deriving informative status, ensure to invoke this default implementation first.

See Also:

`UmlObject.isInformative()`

(continued from last page)

getVisibility

```
public final UmlVisibility getVisibility()
```

getKind

```
public abstract UmlKind getKind()
```

getName

```
public final java.lang.String getName()
```

getAlias

```
public java.lang.String getAlias()
```

getQualifiedName

```
public abstract java.lang.String getQualifiedName()
```

toShortString

```
public java.lang.String toShortString(boolean includeId,
         boolean isNameQualified)
```

toShortString

```
protected java.lang.String toShortString(boolean includeId,
         java.lang.String qualifier,
         boolean isNameQualified)
```

Similar to `toShortString(boolean, boolean)`, but allows to specify a qualifier.

Parameters:

`qualifier` - optional qualifier, specific to subtype

getDescription

```
public final TextDescription getDescription()
```

getHtmlDescription

```
public final TextDescription getHtmlDescription()
```

getStereotype

```
public final UmlStereotype getStereotype()
```

isDeprecated

```
public boolean isDeprecated()
```

This default implementation returns whether the stereotype string of this object contains the string `UmlStereotype.DEPRECATED`.

getUnallowedTagNames

```
public final java.util.Set getUnallowedTagNames()
```

addTaggedValue

```
public final java.lang.String addTaggedValue(java.lang.String name,
         java.lang.String value)
  throws InvalidTagException
```

validateTag

```
protected void validateTag(java.lang.String name,
         java.lang.String value)
```

Subclasses should overwride this method in case some validation about the tagged value is needed before adding it. This default implementation is a no-op.

Parameters:

`name`

`value`

getTaggedValues

```
public final java.util.Map getTaggedValues()
```

org.tanjakostic.jcleancim.model
Class InvalidTagException

```
java.lang.Object
  |-java.lang.Throwable
    |-java.lang.Exception
      |-java.lang.RuntimeException
        |-org.tanjakostic.jcleancim.model.InvalidTagException
```

All Implemented Interfaces:
java.io.Serializable

---

public class **InvalidTagException**
extends java.lang.RuntimeException

Constructor Summary

public	`InvalidTagException(java.lang.String message, java.lang.Throwable cause)` Constructor.
public	`InvalidTagException(java.lang.String message)` Constructor.

Methods inherited from class `java.lang.Throwable`
`addSuppressed, fillInStackTrace, getCause, getLocalizedMessage, getMessage, getStackTrace, getSuppressed, initCause, printStackTrace, printStackTrace, printStackTrace, setStackTrace, toString`

Methods inherited from class `java.lang.Object`
`clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait`

Constructors

InvalidTagException

```
public InvalidTagException(java.lang.String message,
                           java.lang.Throwable cause)
```

Constructor.

Parameters:
`message`
`cause`

(continued from last page)

InvalidTagException

`public` **`InvalidTagException`**`(java.lang.String message)`

Constructor.

Parameters:
`message`

org.tanjakostic.jcleancim.model
Class NameDecomposition

```
java.lang.Object
   |
   +-org.tanjakostic.jcleancim.model.NameDecomposition
```

public class **NameDecomposition**
extends java.lang.Object

Constructor Summary

public	NameDecomposition(java.lang.String inputName, java.util.Map sortedAbbrTerms) Constructor.

Method Summary

static java.util.Map	createTerm(java.lang.String term, java.lang.String desc) Returns term with known description.
static java.util.Map	createUnknownTerm(java.lang.String term) Returns term with unknown description.
java.util.List	getDecomposedTerms()
java.lang.String	getInputName()
boolean	isMatched()
static boolean	isUnknown(java.util.Map termAndDesc) Returns true if the description part in termAndDesc is not known.
java.lang.String	toString()

Methods inherited from class java.lang.Object
clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Constructors

NameDecomposition

```
public NameDecomposition(java.lang.String inputName,
                         java.util.Map sortedAbbrTerms)
```

Constructor.

Methods

(continued from last page)

isUnknown

```
public static boolean isUnknown(java.util.Map termAndDesc)
```

Returns true if the description part in `termAndDesc` is not known.

createUnknownTerm

```
public static java.util.Map createUnknownTerm(java.lang.String term)
```

Returns term with unknown description.

createTerm

```
public static java.util.Map createTerm(java.lang.String term,
         java.lang.String desc)
```

Returns term with known description.

getInputName

```
public java.lang.String getInputName()
```

getDecomposedTerms

```
public java.util.List getDecomposedTerms()
```

isMatched

```
public boolean isMatched()
```

toString

```
public java.lang.String toString()
```

org.tanjakostic.jcleancim.model
Class Namespace

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.model.Namespace
```

public class **Namespace**
extends java.lang.Object

Container for a name-value pair to hold the URI and prefix for a namespace that can be associated to almost any UML object; independent of UML object nature.

Field Summary

public static final	EMPTY

Method Summary

static Namespace	create(java.lang.String uri, java.lang.String prefix)
java.lang.String	getAsDefaultNs() Returns XML expression usable when defining this namespace as default namespace.
java.lang.String	getAsMappedNs() Returns XML expression usable when defining this namespace with its prefix.
java.lang.String	getPrefix() Potentially empty namespace prefix.
java.lang.String	getUri() Potentially empty namespace URI.
boolean	hasPrefix()
boolean	hasURI()
java.lang.String	toString()

Methods inherited from class `java.lang.Object`

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Fields

EMPTY

public static final org.tanjakostic.jcleancim.model.Namespace **EMPTY**

Methods

create

```
public static Namespace create(java.lang.String uri,
         java.lang.String prefix)
```

getUri

```
public java.lang.String getUri()
```

Potentially empty namespace URI.

getPrefix

```
public java.lang.String getPrefix()
```

Potentially empty namespace prefix.

hasPrefix

```
public boolean hasPrefix()
```

hasURI

```
public boolean hasURI()
```

getAsMappedNs

```
public java.lang.String getAsMappedNs()
```

Returns XML expression usable when defining this namespace with its prefix.

getAsDefaultNs

```
public java.lang.String getAsDefaultNs()
```

Returns XML expression usable when defining this namespace as default namespace.

toString

```
public java.lang.String toString()
```

org.tanjakostic.jcleancim.model
Class NamespaceInfo

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.model.NamespaceInfo
```

public class **NamespaceInfo**
extends java.lang.Object

Content of namespace class. In case of CIM, the information is deduced from version class, while for IEC61850, the information is extracted from the namespace class.

Note that the constructor initialises only the "field", but not the list of dependencies; use `addDependency(NamespaceInfo)` to gradually add needed instances as you visit them.

Constructor Summary

public	`NamespaceInfo(java.lang.String id, java.lang.String version, java.lang.String date)` Constructs instance with empty revision and tissues, and with UML version same as namespace version; this is for CIM namespaces, which are deduced from UML version class.
public	`NamespaceInfo(java.lang.String id, java.lang.String version, java.lang.String revision, java.lang.String date, java.lang.String umlVersion, java.lang.String tissuesApplied)` Constructor for IEC61850 namespaces; after construction, you still need to add dependencies with `addDependency(NamespaceInfo)`.

Method Summary

boolean	`addDependency(NamespaceInfo namespace)` Adds `namespace` as dependency to this namespace, if the dependency is not cirular and returns whether addition happened.
static NamespaceInfo	`createCimInstance(VersionInfo versionInfo)` Factory method to construct CIM namespace info from `versionInfo`; see `NamespaceInfo(String, String, String)`.
static NamespaceInfo	`createIec61850Instance(UmlClass nsClass)`
java.lang.String	`getDate()`
java.util.Set	`getDependencies()` Returns all dependencies of this namespace.
java.util.Set	`getDependencyStrings()` Returns namespace strings of all the dependencies.
static java.lang.String	`getExpectedNamespaceClassName(Nature nature, java.lang.String name)` Returns the expected name for the namespace class, as per IEC TC57 UML models rules.
java.lang.String	`getId()`

java.lang.String	getName() Returns formatted string including id, version and revision.
java.lang.String	getRevision()
java.lang.String	getTissuesApplied()
java.lang.String	getUmlVersion()
java.lang.String	getVersion()
java.lang.String	toString()

Methods inherited from class java.lang.Object
clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Constructors

NamespaceInfo

```
public NamespaceInfo(java.lang.String id,
                     java.lang.String version,
                     java.lang.String date)
```

Constructs instance with empty revision and tissues, and with UML version same as namespace version; this is for CIM namespaces, which are deduced from UML version class. After construction, you still need to add dependencies with addDependency(NamespaceInfo).

Parameters:
id
version
date

NamespaceInfo

```
public NamespaceInfo(java.lang.String id,
                     java.lang.String version,
                     java.lang.String revision,
                     java.lang.String date,
                     java.lang.String umlVersion,
                     java.lang.String tissuesApplied)
```

Constructor for IEC61850 namespaces; after construction, you still need to add dependencies with addDependency(NamespaceInfo).

Parameters:
id
version
revision
date
umlVersion
tissuesApplied

(continued from last page)

Methods

getExpectedNamespaceClassName

```
public static java.lang.String getExpectedNamespaceClassName(Nature nature,
         java.lang.String name)
```

Returns the expected name for the namespace class, as per IEC TC57 UML models rules.

createCimInstance

```
public static NamespaceInfo createCimInstance(VersionInfo versionInfo)
```

Factory method to construct CIM namespace info from `versionInfo`; see `NamespaceInfo(String, String, String)`.

createIec61850Instance

```
public static NamespaceInfo createIec61850Instance(UmlClass nsClass)
```

getId

```
public java.lang.String getId()
```

getVersion

```
public java.lang.String getVersion()
```

getRevision

```
public java.lang.String getRevision()
```

getDate

```
public java.lang.String getDate()
```

getUmlVersion

```
public java.lang.String getUmlVersion()
```

getTissuesApplied

```
public java.lang.String getTissuesApplied()
```

getDependencies

`public java.util.Set **getDependencies**()`

Returns all dependencies of this namespace.

addDependency

`public boolean **addDependency**(NamespaceInfo namespace)`

Adds `namespace` as dependency to this namespace, if the dependency is not cirular and returns whether addition happened.

getName

`public java.lang.String **getName**()`

Returns formatted string including id, version and revision.

getDependencyStrings

`public java.util.Set **getDependencyStrings**()`

Returns namespace strings of all the dependencies.

toString

`public java.lang.String **toString**()`

org.tanjakostic.jcleancim.model
Class PresenceCondition

```
java.lang.Object
   |
   +-org.tanjakostic.jcleancim.model.PresenceCondition
```

public class **PresenceCondition**
extends java.lang.Object

Field Summary

public static final	ARG_CONDID Value: **(condID)**
public static final	ARG_N Value: **(n)**
public static final	ARG_SIBLING Value: **(sibling)**
public static final	F
public static final	M
public static final	NA
public static final	O
public static final	STEM_END_COND Stem end for presence conditions with 'condID' argument. Value: **cond**

Method Summary

java.lang.String	getArgs()
UmlConstraint	getConstraint() Returns potentially null constraint from which this presence condition has been created.
UmlAttribute	getDefinitionLiteral() Returns (potentially null) UML literal defining this presence condition; it is null in case there is an error in the model and the presence condition found in the class is not a standard one.
static java.util.Set	getNamesOfImplicits()
java.lang.String	getStem()

java.lang.String	getStemAndArgs() Returns stem(args) if there are arguments, otherwise just stem.
java.lang.String	getText()
boolean	isWithCondID() Returns true if this is a presence condition with the non-machine-processable argument ARG_CONDID (meaning the stem ends with STEM_END_COND), false otherwise.
java.lang.String	toString()

Methods inherited from class java.lang.Object
clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Fields

STEM_END_COND

public static final java.lang.String **STEM_END_COND**

Stem end for presence conditions with 'condID' argument.
Constant value: **cond**

ARG_CONDID

public static final java.lang.String **ARG_CONDID**

Constant value: **(condID)**

ARG_N

public static final java.lang.String **ARG_N**

Constant value: **(n)**

ARG_SIBLING

public static final java.lang.String **ARG_SIBLING**

Constant value: **(sibling)**

M

public static final org.tanjakostic.jcleancim.model.PresenceCondition **M**

O

public static final org.tanjakostic.jcleancim.model.PresenceCondition **O**

(continued from last page)

NA

`public static final org.tanjakostic.jcleancim.model.PresenceCondition NA`

F

`public static final org.tanjakostic.jcleancim.model.PresenceCondition F`

Methods

getNamesOfImplicits

`public static java.util.Set getNamesOfImplicits()`

getConstraint

`public UmlConstraint getConstraint()`

Returns potentially null constraint from which this presence condition has been created.

getDefinitionLiteral

`public UmlAttribute getDefinitionLiteral()`

Returns (potentially null) UML literal defining this presence condition; it is null in case there is an error in the model and the presence condition found in the class is not a standard one.

getStem

`public java.lang.String getStem()`

getArgs

`public java.lang.String getArgs()`

getText

`public java.lang.String getText()`

getStemAndArgs

`public java.lang.String getStemAndArgs()`

Returns `stem(args)` if there are arguments, otherwise just `stem`.

isWithCondID

`public boolean **isWithCondID**()`

Returns true if this is a presence condition with the non-machine-processable argument `ARG_CONDID` (meaning the stem ends with `STEM_END_COND`), false otherwise.

toString

`public java.lang.String **toString**()`

org.tanjakostic.jcleancim.model
Class TextDescription

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.model.TextDescription
```

Direct Known Subclasses:
CellText

---

public class **TextDescription**
extends java.lang.Object

Simple data structure to hold together the text description content and its format to facilitate writing of UML documentation that may be formatted.

Ensure you specify correct kind (text format), otherwise the result of writing the documentation may have unexpected formatting.

Nested Class Summary

class	TextDescription.TextKind TextDescription.TextKind

Field Summary

public static final	DEFAULT_KIND
public static final	DEFAULT_TEXT Value:
public static final	EMPTY_HTML
public static final	EMPTY_TXT
public final	kind
public final	text

Constructor Summary

public	TextDescription() Creates an instance with defaults.
public	TextDescription(java.lang.String text) Creates an instance with the content in `text` trimmed of whitespace, and DEFAULT_KIND.
public	TextDescription(java.lang.String text, TextDescription.TextKind kind) Constructor.

Method Summary

TextDescription	appendParagraph(java.lang.String paragraph) Returns new instance with the paragraph appended to the original text as a paragraph (for HTML, it will enclose paragraph into paragraph tags, and for text, it will first append a new line character then paragraph); or unmodified instance if paragraph is null or empty.
boolean	isEmpty()
TextDescription	prepend(java.lang.String prefix) Returns new instance with the prefix prepended to the original text; or unmodified instance if prefix is null or empty.
TextDescription	prepend(java.lang.String prefix, java.lang.Object o) Returns new instance with the prefix prepended to the original text; or unmodified instance if prefix is null or empty.
java.lang.String	toString()

Methods inherited from class java.lang.Object
clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Fields

DEFAULT_KIND

public static final org.tanjakostic.jcleancim.model.TextDescription.TextKind **DEFAULT_KIND**

DEFAULT_TEXT

public static final java.lang.String **DEFAULT_TEXT**

Constant value:

EMPTY_TXT

public static final org.tanjakostic.jcleancim.model.TextDescription **EMPTY_TXT**

EMPTY_HTML

public static final org.tanjakostic.jcleancim.model.TextDescription **EMPTY_HTML**

kind

public final org.tanjakostic.jcleancim.model.TextDescription.TextKind **kind**

text

```
public final java.lang.String text
```

Constructors

TextDescription

```
public TextDescription()
```

Creates an instance with defaults.

TextDescription

```
public TextDescription(java.lang.String text)
```

Creates an instance with the content in `text` trimmed of whitespace, and `DEFAULT_KIND`.

Parameters:
`text`

TextDescription

```
public TextDescription(java.lang.String text,
                       TextDescription.TextKind kind)
```

Constructor. If any argument is null, uses defaults.

Parameters:
`text` - text; will be trimmed of whitespace.
`kind` - kind of text; if `text` contains only whitespace and the argument is `TextDescription.TextKind.textWithNL`, the kind stored will be `TextDescription.TextKind.textNoNL`. It is the responsibility of the caller to properly set the kind of text according to the content in `text`, otherwise any writing may produce undesired results.

Methods

prepend

```
public TextDescription prepend(java.lang.String prefix)
```

Returns new instance with the `prefix` prepended to the original text; or unmodified instance if `prefix` is null or empty.

Parameters:
`prefix` - can be null/empty, but should not contain any markup or new line character (otherwise, result is undefined).

prepend

```
public TextDescription prepend(java.lang.String prefix,
        java.lang.Object o)
```

Returns new instance with the `prefix` prepended to the original text; or unmodified instance if `prefix` is null or empty.

Parameters:
`prefix` - can be null/empty, but should not contain any markup or new line character (otherwise, result is undefined).
`o` - if not null, may be used for logging warning condition.

appendParagraph

`public TextDescription appendParagraph(java.lang.String paragraph)`

Returns new instance with the `paragraph` appended to the original text as a paragraph (for HTML, it will enclose `paragraph` into paragraph tags, and for text, it will first append a new line character then `paragraph`); or unmodified instance if `paragraph` is null or empty.

Parameters:

`paragraph` - can be null/empty, can contain markup or new line character.

isEmpty

`public boolean isEmpty()`

toString

`public java.lang.String toString()`

org.tanjakostic.jcleancim.model
Class TextDescription.TextKind

```
java.lang.Object
  +-java.lang.Enum
    +-org.tanjakostic.jcleancim.model.TextDescription.TextKind
```

All Implemented Interfaces:
java.io.Serializable, java.lang.Comparable

---

public static final class **TextDescription.TextKind**
extends java.lang.Enum

Kind of text formatting that helps to optimise writing text to various formats.

Field Summary

public static final	htmlSnippet HTML snippet, the content of the document body section.
public static final	textNoNL Text without any formatting, without any new line characters.
public static final	textWithNL Text without any formatting, but having one or more new line characters.

Method Summary

static TextDescription.TextKind	valueOf(java.lang.String name)
static TextDescription.TextKind[]	values()

Methods inherited from class `java.lang.Enum`

clone, compareTo, equals, finalize, getDeclaringClass, hashCode, name, ordinal, toString, valueOf

Methods inherited from class `java.lang.Object`

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface `java.lang.Comparable`

compareTo

Fields

(continued from last page)

textNoNL

`public static final org.tanjakostic.jcleancim.model.TextDescription.TextKind` **`textNoNL`**

Text without any formatting, without any new line characters.

textWithNL

`public static final org.tanjakostic.jcleancim.model.TextDescription.TextKind` **`textWithNL`**

Text without any formatting, but having one or more new line characters.

htmlSnippet

`public static final org.tanjakostic.jcleancim.model.TextDescription.TextKind` **`htmlSnippet`**

HTML snippet, the content of the document `body` section.

Methods

values

`public static TextDescription.TextKind[]` **`values`**`()`

valueOf

`public static TextDescription.TextKind` **`valueOf`**`(java.lang.String name)`

org.tanjakostic.jcleancim.model
Class UML

```
java.lang.Object
   |
   +-org.tanjakostic.jcleancim.model.UML
```

public class **UML**
extends java.lang.Object

All the names from UML models that we rely on for various processing.

Changing any of these in the UML will require change in the code, in this file and potentially in one or more of the config.properties.

Field Summary

public static final	ACT Value: **ACT**
public static final	ATTR_attr Private attribute name on FCDA meta-model type, to hold actual DA type used. Value: **attr**
public static final	ATTR_ctlVal Private attribute name on DA meta-model type, to hold basic type for control parameter. Value: **ctlVal**
public static final	ATTR_date Attribute in a top-package or CIM profile version class (and 61850 namespace class). Value: **date**
public static final	ATTR_id Value: **id**
public static final	ATTR_revision Value: **revision**
public static final	ATTR_tissuesApplied Value: **tissuesApplied**
public static final	ATTR_val Private attribute name on DA meta-model type, to hold basic type used. Value: **val**
public static final	ATTR_version Attribute in a top-package or CIM profile version class (and 61850 namespace class). Value: **version**
public static final	BasePrimitiveCDC Value: **BasePrimitiveCDC**

public static final	CDCAnalogueInfo Value: **CDCAnalogueInfo**
public static final	CDCAnalogueSet Value: **CDCAnalogueSet**
public static final	CDCControl Value: **CDCControl**
public static final	CDCDescription Value: **CDCDescription**
public static final	CDCServiceTracking Value: **CDCServiceTracking**
public static final	CDCStatusInfo Value: **CDCStatusInfo**
public static final	CDCStatusSet Value: **CDCStatusSet**
public static final	CIM_DT_multiplier Value: **multiplier**
public static final	CIM_DT_unit Value: **unit**
public static final	CIM_DT_value Value: **value**
public static final	CIM_VERSION_CLASS_SUFFIX CIM-specific suffix for version class name. Value: **CIMVersion**
public static final	CLASS_abstract Qualifier for a class that is abstract. Value: **abstract**
public static final	ClcMth Value: **ClcMth**
public static final	ClcSrc Value: **ClcSrc**
public static final	CONSTR_TXT_maxIdx WG10 CDC multi-valued attributes may have allowed min index as note in named constraints. Value: **maxIdx**

public static final	CONSTR_TXT_minIdx WG10 CDC multi-valued attributes may have allowed max index as note in named constraints. Value: **minIdx**
public static final	CTS Value: **CTS**
public static final	DA Value: **DA**
public static final	DetailedDiagrams Reserved name for a package that contains diagrams that may be useful for information but not for printing into any generated spec. Value: **DetailedDiagrams**
public static final	DOC_FORMAT_STRING Format string, including the prefix for packages reserved for diagrams used for the template only, and that should not be printed with the content of the regular model. Value: **Doc%s**
public static final	ENC Value: **ENC**
public static final	ENG Value: **ENG**
public static final	ENS Value: **ENS**
public static final	ERY Value: **ERY**
public static final	FCDA_BL Value: **FCDA_BL**
public static final	FCDA_CF Value: **FCDA_CF**
public static final	FCDA_DC Value: **FCDA_DC**
public static final	FCDA_EX Value: **FCDA_EX**
public static final	FCDA_MX Value: **FCDA_MX**
public static final	FCDA_OR Value: **FCDA_OR**

public static final	FCDA_SE Value: **FCDA_SE**
public static final	FCDA_SG Value: **FCDA_SG**
public static final	FCDA_SP Value: **FCDA_SP**
public static final	FCDA_SR Value: **FCDA_SR**
public static final	FCDA_ST Value: **FCDA_ST**
public static final	FCDA_SV Value: **FCDA_SV**
public static final	IEC61850_NAMESPACE_CLASS_SUFFIX IEC61850-specific suffix for namespace class name. Value: **Namespace**
public static final	IEC61850_VERSION_CLASS_SUFFIX IEC61850-specific suffix for version class name. Value: **UMLVersion**
public static final	IGNORE_CASE_ABBREVS
public static final	IGNORE_CASE_DAS
public static final	IGNORE_CASE_ENUMS CIM classes for whose attributes/literals we don't verify upper/lower case and plural.
public static final	INF_PREFIX Prefix for informative sub-packages. Value: **Inf**
public static final	LLN0 Value: **LLN0**
public static final	PC_F Value: **F**
public static final	PC_M Value: **M**
public static final	PC_na Value: **na**

public static final	PC_O Value: **O**
public static final	PREF_DOName_Ieee Value: **Ieee**
public static final	PREF_LNGroup Value: **LNGroup**
public static final	PREF_P_ Value: **P_**
public static final	PREF_S_ Value: **S_**
public static final	SPC Value: **SPC**
public static final	SPS Value: **SPS**
public static final	StatisticsLN Value: **StatisticsLN**
public static final	SUFF_CONTROL Suffix for DerivedDA class name applicable for ENC. Value: **_control**
public static final	SUFF_Transient Value: **Transient**
public static final	SUPER_CDC Value: **CDC**
public static final	SUPER_COMP_DA Value: **ComposedDA**
public static final	SUPER_COMPOSED_FCDA Value: **ComposedFCDA**
public static final	SUPER_DA Value: **DA**
public static final	SUPER_ENUM_DA Value: **EnumDA**
public static final	SUPER_ENUM_FCDA Value: **EnumFCDA**

public static final	SUPER_FCDA Value: **FCDA**
public static final	SUPER_PACKED_ENUM_DA Value: **PackedEnumDA**
public static final	SUPER_PACKED_ENUM_FCDA Value: **PackedEnumFCDA**
public static final	SUPER_PACKED_FCDA Value: **PackedListFCDA**
public static final	SUPER_PACKED_PRIM_DA Value: **PackedPrimitiveDA**
public static final	SUPER_PRIM_DA Value: **PrimitiveDA**
public static final	TAG_moveAfter WG10 CDC and DA attributes may have tagged value, to move their position for printing. Value: **moveAfter**
public static final	TAG_SCL_emptyValue WG10 has some enum types with literal "none" that should be printed as empty value. Value: **scl:emptyValue**
public static final	TVN_cdcId WG10-specific tagged value, used in CDC tables (7-3). Value: **cdcId**
public static final	TVN_datId WG10-specific tagged value, used in DA tables (7-3). Value: **datId**
public static final	TVN_GUIDBasedOn Heavily used in CIM UML profiles, but never in canonical CIM. Value: **GUIDBasedOn**
public static final	TVN_iecRef Value: **iecRef**
public static final	TVN_ieeeRef Value: **ieeeRef**
public static final	TVN_nsprefix Tagged value name to specify namespace prefix for a UML object. Value: **nsprefix**
public static final	TVN_nsuri Tagged value name to specify namespace URI for a UML object. Value: **nsuri**

public static final	TVN_oldName WG10-specific tagged value to refer to old type name, used in core types only (7-2). Value: **oldName**
public static final	TVN_rsName Value: **rsName**
public static final	TVN_throws Exceptions in EA must be specified as tagged value with the tag name 'throws'. Value: **throws**

Methods inherited from class `java.lang.Object`
`clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait`

Fields

SUPER_PACKED_ENUM_DA

`public static final java.lang.String` **SUPER_PACKED_ENUM_DA**

Constant value: **PackedEnumDA**

SUPER_ENUM_DA

`public static final java.lang.String` **SUPER_ENUM_DA**

Constant value: **EnumDA**

SUPER_PACKED_PRIM_DA

`public static final java.lang.String` **SUPER_PACKED_PRIM_DA**

Constant value: **PackedPrimitiveDA**

SUPER_PRIM_DA

`public static final java.lang.String` **SUPER_PRIM_DA**

Constant value: **PrimitiveDA**

SUPER_COMP_DA

`public static final java.lang.String` **SUPER_COMP_DA**

Constant value: **ComposedDA**

(continued from last page)

SUPER_DA

`public static final java.lang.String` **`SUPER_DA`**

Constant value: **`DA`**

SUPER_PACKED_ENUM_FCDA

`public static final java.lang.String` **`SUPER_PACKED_ENUM_FCDA`**

Constant value: **`PackedEnumFCDA`**

SUPER_ENUM_FCDA

`public static final java.lang.String` **`SUPER_ENUM_FCDA`**

Constant value: **`EnumFCDA`**

SUPER_PACKED_FCDA

`public static final java.lang.String` **`SUPER_PACKED_FCDA`**

Constant value: **`PackedListFCDA`**

SUPER_COMPOSED_FCDA

`public static final java.lang.String` **`SUPER_COMPOSED_FCDA`**

Constant value: **`ComposedFCDA`**

SUPER_FCDA

`public static final java.lang.String` **`SUPER_FCDA`**

Constant value: **`FCDA`**

SUPER_CDC

`public static final java.lang.String` **`SUPER_CDC`**

Constant value: **`CDC`**

CDCDescription

`public static final java.lang.String` **`CDCDescription`**

Constant value: **`CDCDescription`**

CDCStatusInfo

`public static final java.lang.String` **`CDCStatusInfo`**

(continued from last page)

Constant value: **CDCStatusInfo**

CDCAnalogueInfo

`public static final java.lang.String` **CDCAnalogueInfo**

Constant value: **CDCAnalogueInfo**

CDCControl

`public static final java.lang.String` **CDCControl**

Constant value: **CDCControl**

CDCStatusSet

`public static final java.lang.String` **CDCStatusSet**

Constant value: **CDCStatusSet**

CDCAnalogueSet

`public static final java.lang.String` **CDCAnalogueSet**

Constant value: **CDCAnalogueSet**

CDCServiceTracking

`public static final java.lang.String` **CDCServiceTracking**

Constant value: **CDCServiceTracking**

ENS

`public static final java.lang.String` **ENS**

Constant value: **ENS**

ENC

`public static final java.lang.String` **ENC**

Constant value: **ENC**

ENG

`public static final java.lang.String` **ENG**

Constant value: **ENG**

ERY

`public static final java.lang.String` **`ERY`**

Constant value: **`ERY`**

SPS

`public static final java.lang.String` **`SPS`**

Constant value: **`SPS`**

ACT

`public static final java.lang.String` **`ACT`**

Constant value: **`ACT`**

SPC

`public static final java.lang.String` **`SPC`**

Constant value: **`SPC`**

CTS

`public static final java.lang.String` **`CTS`**

Constant value: **`CTS`**

BasePrimitiveCDC

`public static final java.lang.String` **`BasePrimitiveCDC`**

Constant value: **`BasePrimitiveCDC`**

FCDA_ST

`public static final java.lang.String` **`FCDA_ST`**

Constant value: **`FCDA_ST`**

FCDA_MX

`public static final java.lang.String` **`FCDA_MX`**

Constant value: **`FCDA_MX`**

(continued from last page)

FCDA_OR

`public static final java.lang.String` **`FCDA_OR`**

Constant value: **`FCDA_OR`**

FCDA_SV

`public static final java.lang.String` **`FCDA_SV`**

Constant value: **`FCDA_SV`**

FCDA_BL

`public static final java.lang.String` **`FCDA_BL`**

Constant value: **`FCDA_BL`**

FCDA_SP

`public static final java.lang.String` **`FCDA_SP`**

Constant value: **`FCDA_SP`**

FCDA_SE

`public static final java.lang.String` **`FCDA_SE`**

Constant value: **`FCDA_SE`**

FCDA_SG

`public static final java.lang.String` **`FCDA_SG`**

Constant value: **`FCDA_SG`**

FCDA_CF

`public static final java.lang.String` **`FCDA_CF`**

Constant value: **`FCDA_CF`**

FCDA_DC

`public static final java.lang.String` **`FCDA_DC`**

Constant value: **`FCDA_DC`**

FCDA_EX

`public static final java.lang.String` **`FCDA_EX`**

(continued from last page)

Constant value: **FCDA_EX**

FCDA_SR

`public static final java.lang.String` **FCDA_SR**

Constant value: **FCDA_SR**

DA

`public static final java.lang.String` **DA**

Constant value: **DA**

TAG_SCL_emptyValue

`public static final java.lang.String` **TAG_SCL_emptyValue**

WG10 has some enum types with literal "none" that should be printed as empty value.
Constant value: **scl:emptyValue**

TAG_moveAfter

`public static final java.lang.String` **TAG_moveAfter**

WG10 CDC and DA attributes may have tagged value, to move their position for printing.
Constant value: **moveAfter**

CONSTR_TXT_minIdx

`public static final java.lang.String` **CONSTR_TXT_minIdx**

WG10 CDC multi-valued attributes may have allowed max index as note in named constraints.
Constant value: **minIdx**

CONSTR_TXT_maxIdx

`public static final java.lang.String` **CONSTR_TXT_maxIdx**

WG10 CDC multi-valued attributes may have allowed min index as note in named constraints.
Constant value: **maxIdx**

TVN_rsName

`public static final java.lang.String` **TVN_rsName**

Constant value: **rsName**

TVN_ieeeRef

`public static final java.lang.String` **TVN_ieeeRef**

Constant value: **ieeeRef**

TVN_iecRef

`public static final java.lang.String` **`TVN_iecRef`**

Constant value: **`iecRef`**

TVN_datId

`public static final java.lang.String` **`TVN_datId`**

WG10-specific tagged value, used in DA tables (7-3).

Note that at present this is not really used as a tagged value but rather as a cludge for printing correct table title; it should actually be a part of some meta-model, similar to `TVN_cdcId`. Until we do have effectively that in a meta-model, I keep this one here for uniform processing.
Constant value: **`datId`**

TVN_cdcId

`public static final java.lang.String` **`TVN_cdcId`**

WG10-specific tagged value, used in CDC tables (7-3).
Constant value: **`cdcId`**

TVN_oldName

`public static final java.lang.String` **`TVN_oldName`**

WG10-specific tagged value to refer to old type name, used in core types only (7-2).
Constant value: **`oldName`**

StatisticsLN

`public static final java.lang.String` **`StatisticsLN`**

Constant value: **`StatisticsLN`**

LLN0

`public static final java.lang.String` **`LLN0`**

Constant value: **`LLN0`**

ClcMth

`public static final java.lang.String` **`ClcMth`**

Constant value: **`ClcMth`**

ClcSrc

`public static final java.lang.String` **`ClcSrc`**

Constant value: **`ClcSrc`**

PC_M

`public static final java.lang.String` **`PC_M`**

Constant value: **`M`**

PC_O

`public static final java.lang.String` **`PC_O`**

Constant value: **`O`**

PC_F

`public static final java.lang.String` **`PC_F`**

Constant value: **`F`**

PC_na

`public static final java.lang.String` **`PC_na`**

Constant value: **`na`**

IGNORE_CASE_DAS

`public static final java.util.Set` **`IGNORE_CASE_DAS`**

IGNORE_CASE_ABBREVS

`public static final java.util.Set` **`IGNORE_CASE_ABBREVS`**

PREF_LNGroup

`public static final java.lang.String` **`PREF_LNGroup`**

Constant value: **`LNGroup`**

PREF_DOName_Ieee

`public static final java.lang.String` **`PREF_DOName_Ieee`**

Constant value: **`Ieee`**

PREF_P_

`public static final java.lang.String` **`PREF_P_`**

(continued from last page)

Constant value: **P_**

PREF_S_

`public static final java.lang.String` **PREF_S_**

Constant value: **S_**

SUFF_Transient

`public static final java.lang.String` **SUFF_Transient**

Constant value: **Transient**

IEC61850_NAMESPACE_CLASS_SUFFIX

`public static final java.lang.String` **IEC61850_NAMESPACE_CLASS_SUFFIX**

IEC61850-specific suffix for namespace class name.
Constant value: **Namespace**

IEC61850_VERSION_CLASS_SUFFIX

`public static final java.lang.String` **IEC61850_VERSION_CLASS_SUFFIX**

IEC61850-specific suffix for version class name.
Constant value: **UMLVersion**

ATTR_id

`public static final java.lang.String` **ATTR_id**

Constant value: **id**

ATTR_revision

`public static final java.lang.String` **ATTR_revision**

Constant value: **revision**

ATTR_tissuesApplied

`public static final java.lang.String` **ATTR_tissuesApplied**

Constant value: **tissuesApplied**

ATTR_val

`public static final java.lang.String` **ATTR_val**

Private attribute name on DA meta-model type, to hold basic type used.
Constant value: **val**

(continued from last page)

SUFF_CONTROL

`public static final java.lang.String` **`SUFF_CONTROL`**

Suffix for DerivedDA class name applicable for `ENC`.
Constant value: **`_control`**

ATTR_ctlVal

`public static final java.lang.String` **`ATTR_ctlVal`**

Private attribute name on DA meta-model type, to hold basic type for control parameter.
Constant value: **`ctlVal`**

ATTR_attr

`public static final java.lang.String` **`ATTR_attr`**

Private attribute name on FCDA meta-model type, to hold actual DA type used.
Constant value: **`attr`**

CIM_DT_value

`public static final java.lang.String` **`CIM_DT_value`**

Constant value: **`value`**

CIM_DT_unit

`public static final java.lang.String` **`CIM_DT_unit`**

Constant value: **`unit`**

CIM_DT_multiplier

`public static final java.lang.String` **`CIM_DT_multiplier`**

Constant value: **`multiplier`**

TVN_GUIDBasedOn

`public static final java.lang.String` **`TVN_GUIDBasedOn`**

Heavily used in CIM UML profiles, but never in canonical CIM.
Constant value: **`GUIDBasedOn`**

IGNORE_CASE_ENUMS

`public static final java.util.Set` **`IGNORE_CASE_ENUMS`**

CIM classes for whose attributes/literals we don't verify upper/lower case and plural.

CIM_VERSION_CLASS_SUFFIX

`public static final java.lang.String` **`CIM_VERSION_CLASS_SUFFIX`**

(continued from last page)

CIM-specific suffix for version class name.
Constant value: **CIMVersion**

ATTR_version

`public static final java.lang.String` **ATTR_version**

Attribute in a top-package or CIM profile version class (and 61850 namespace class).
Constant value: **version**

ATTR_date

`public static final java.lang.String` **ATTR_date**

Attribute in a top-package or CIM profile version class (and 61850 namespace class).
Constant value: **date**

TVN_nsuri

`public static final java.lang.String` **TVN_nsuri**

Tagged value name to specify namespace URI for a UML object.
Constant value: **nsuri**

TVN_nsprefix

`public static final java.lang.String` **TVN_nsprefix**

Tagged value name to specify namespace prefix for a UML object.
Constant value: **nsprefix**

CLASS_abstract

`public static final java.lang.String` **CLASS_abstract**

Qualifier for a class that is abstract.
Constant value: **abstract**

TVN_throws

`public static final java.lang.String` **TVN_throws**

Exceptions in EA must be specified as tagged value with the tag name 'throws'.
Constant value: **throws**

INF_PREFIX

`public static final java.lang.String` **INF_PREFIX**

Prefix for informative sub-packages. This is to avoid name clashes among normative and informative packages (and to make informative sub-packages obvious). Note that this will apply to all packages that start with this prefix (e.g., 'InfWork' as well as 'Informative'.
Constant value: **Inf**

DOC_FORMAT_STRING

`public static final java.lang.String` **DOC_FORMAT_STRING**

(continued from last page)

Format string, including the prefix for packages reserved for diagrams used for the template only, and that should not be printed with the content of the regular model. The subpackage of "ABC" that has name "DocABC" will be matched, and considered as informative package.
Constant value: **`Doc%s`**

DetailedDiagrams

`public static final java.lang.String` **`DetailedDiagrams`**

Reserved name for a package that contains diagrams that may be useful for information but not for printing into any generated spec. Typically contains diagrams that illustrate open CIM issues, or that illustrate classes used in profiles. It is considered as informative.

In addition, because EA takes long to export diagrams, and we never want to reference these diagrams from within a template, they are never exported (i.e., they always have an empty pic).
Constant value: **`DetailedDiagrams`**

org.tanjakostic.jcleancim.model
Class UmlAssociation

```
java.lang.Object
  |-org.tanjakostic.jcleancim.model.AbstractUmlObject
    |-org.tanjakostic.jcleancim.model.UmlAssociation
```

All Implemented Interfaces:
UmlObject

---

public class **UmlAssociation**
extends AbstractUmlObject

UML association, with its two ends (UML classes).

All the information related to the types/UML classes (`getSource()` and `getTarget()`) gets derived from the contained association ends (`getSourceEnd()` and `getTargetEnd()`). Whether association is informative, in contrast, is based on the stereotype (this allows us to identify and report when the due stereotype is missing).

Ownership of association is defined in `OwningWg`, according to TC57 rules.

Associations with their two ends are a bit tricky. To define owner (owning top-level package and its WG), we use the classes at both ends, i.e., methods `getSource()` and `getTarget()`.

Consider example association from combined CIM model, between Location and PowerSystemResource. Location (from IEC61968) is the source and PowerSystemResource (from IEC61970) is the target. Qualified association ends, to display association, are shown so:

[0..*] Location.PowerSystemResources - [0..1] PowerSystemResource.Location

---

Nested Class Summary

class	`UmlAssociation.Data` UmlAssociation.Data
class	`UmlAssociation.Direction` UmlAssociation.Direction

Fields inherited from class `org.tanjakostic.jcleancim.model.AbstractUmlObject`
`CLASS_SEPARATOR`, `NULL_OBJ_NAME`, `PACKAGE_SEPARATOR`

Method Summary

boolean	`areEndVisibilitiesSame()` Returns whether both ends have the same visibility.
`UmlAssociation.Direction`	`getDirection()` **Deprecated.** *use* `getNavigability()` *instead.*
`UmlAssociationEndPair`	`getEndsAsSource(boolean asSource)` Returns the pair of association ends from the perspective of source if `asSource` true (this is what we need when printing model documentation for associations of a `type`, or when reading RDF/OWL properties), from the perspective of target otherwise.

UmlKind	getKind()
static java.util.List	getKinds(Nature nature) Returns all available classifications (kinds) for associations.
Namespace	getNamespace()
Nature	getNature()
UmlAssociation.Direction	getNavigability() Returns the nature of navigability (whether navigable); for direction, use UmlAssociationEnd.getNavigable() for association ends.
OwningWg	getOwner()
java.util.Set	getPredefinedTagNames()
java.lang.String	getQualifiedName()
UmlClass	getSource() Returns UmlClass used as type for the source end.
UmlAssociationEnd	getSourceEnd() Returns source end.
UmlClass	getTarget() Returns UmlClass used as type for the target end.
UmlAssociationEnd	getTargetEnd() Returns target end.
boolean	hasANavigableEnd() Returns whether this association has at least one navigable end.
boolean	involvesWg(OwningWg wg) Returns whether any of two classes of this association involve owner wg.
boolean	isAtLeastOneEndPublic() Returns whether at least one end is public.
boolean	isDirectionMismatchForEnds() Returns whether an association with unspecified navigability has a navigable end.
boolean	isMapping() Returns whether this association is accross model domains.
boolean	isNonPrivate() Returns whether any end is not private.
boolean	isNonPublic() Returns whether neither end is public.
boolean	isWithinSameWg() Returns whether this association is between the classes with the same owner.
java.lang.String	toString()

Methods inherited from class `org.tanjakostic.jcleancim.model.AbstractUmlObject`

`addDeprecAndInf`, `addTaggedValue`, `appendRemainingCustomStereotypes`, `classifyPerScope`, `classifyPerScopePerTag`, `classifyPerTag`, `collectDuplicateDescriptions`, `collectDuplicateNames`, `collectForScope`, `collectNames`, `collectQNames`, `findAllForName`, `findWithSameUuidAndLog`, `getAlias`, `getDescription`, `getHtmlDescription`, `getId`, `getKind`, `getName`, `getNature`, `getOwner`, `getQualifiedName`, `getSince`, `getStereotype`, `getTaggedValues`, `getUnallowedTagNames`, `getUuid`, `getVisibility`, `initFromTags`, `isDeprecated`, `isInformative`, `saveTags`, `toShortString`, `toShortString`, `validateTag`

Methods inherited from class `java.lang.Object`

`clone`, `equals`, `finalize`, `getClass`, `hashCode`, `notify`, `notifyAll`, `toString`, `wait`, `wait`, `wait`

Methods inherited from interface `org.tanjakostic.jcleancim.model.UmlObject`

`addTaggedValue`, `getAlias`, `getDescription`, `getHtmlDescription`, `getId`, `getKind`, `getName`, `getNamespace`, `getNature`, `getOwner`, `getPredefinedTagNames`, `getQualifiedName`, `getSince`, `getStereotype`, `getTaggedValues`, `getUnallowedTagNames`, `getUuid`, `getVisibility`, `isDeprecated`, `isInformative`, `toShortString`

Methods

getKinds

`public static java.util.List` **`getKinds`**`(Nature nature)`

Returns all available classifications (kinds) for associations.

Parameters:
`nature` - ignored in this method

getSourceEnd

`public UmlAssociationEnd` **`getSourceEnd`**`()`

Returns source end.

getTargetEnd

`public UmlAssociationEnd` **`getTargetEnd`**`()`

Returns target end.

getEndsAsSource

`public UmlAssociationEndPair` **`getEndsAsSource`**`(boolean asSource)`

Returns the pair of association ends from the perspective of source if `asSource` true (this is what we need when printing model documentation for associations of a `type`, or when reading RDF/OWL properties), from the perspective of target otherwise.

(continued from last page)

getSource

`public UmlClass getSource()`

Returns `UmlClass` used as type for the source end.

getTarget

`public UmlClass getTarget()`

Returns `UmlClass` used as type for the target end.

isNonPrivate

`public boolean isNonPrivate()`

Returns whether any end is not private.

isNonPublic

`public boolean isNonPublic()`

Returns whether neither end is public.

isAtLeastOneEndPublic

`public boolean isAtLeastOneEndPublic()`

Returns whether at least one end is public.

areEndVisibilitiesSame

`public boolean areEndVisibilitiesSame()`

Returns whether both ends have the same visibility.

getDirection

`public UmlAssociation.Direction getDirection()`

Deprecated. *use* `getNavigability()` *instead.*

Returns the direction (navigability).

getNavigability

`public UmlAssociation.Direction getNavigability()`

Returns the nature of navigability (whether navigable); for direction, use `UmlAssociationEnd.getNavigable()` for association ends.

hasANavigableEnd

`public boolean hasANavigableEnd()`

Returns whether this association has at least one navigable end.

(continued from last page)

isDirectionMismatchForEnds

`public boolean isDirectionMismatchForEnds()`

Returns whether an association with unspecified navigability has a navigable end. This may happen in EA when you draw an association (which gets created as navigable, according to your local EA settings) and then you make it of unspecified direction: EA does not correctly update the formerly navigable end to unspecified.

isMapping

`public boolean isMapping()`

Returns whether this association is accross model domains.

isWithinSameWg

`public boolean isWithinSameWg()`

Returns whether this association is between the classes with the same owner.

involvesWg

`public boolean involvesWg(OwningWg wg)`

Returns whether any of two classes of this association involve owner `wg`.

getOwner

`public OwningWg getOwner()`

getNamespace

`public Namespace getNamespace()`

Returns own namespace initialised from tagged values if not empty. Otherwise, returns the namespace of the source end.

getNature

`public Nature getNature()`

getKind

`public UmlKind getKind()`

getQualifiedName

`public java.lang.String getQualifiedName()`

(continued from last page)

getPredefinedTagNames

```
public java.util.Set getPredefinedTagNames()
```

toString

```
public java.lang.String toString()
```

org.tanjakostic.jcleancim.model
Class UmlAssociation.Direction

```
java.lang.Object
  +-java.lang.Enum
    +-org.tanjakostic.jcleancim.model.UmlAssociation.Direction
```

All Implemented Interfaces:
java.io.Serializable, java.lang.Comparable

---

public static final class **UmlAssociation.Direction**
extends java.lang.Enum

Direction (navigability) of association.

Field Summary

public static final	biDirectional
public static final	directed
public static final	unspecified

Method Summary

static UmlAssociation.Direction	valueOf(java.lang.String name)
static UmlAssociation.Direction[]	values()

Methods inherited from class `java.lang.Enum`

clone, compareTo, equals, finalize, getDeclaringClass, hashCode, name, ordinal, toString, valueOf

Methods inherited from class `java.lang.Object`

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface `java.lang.Comparable`

compareTo

Fields

(continued from last page)

biDirectional

```
public static final org.tanjakostic.jcleancim.model.UmlAssociation.Direction
biDirectional
```

directed

```
public static final org.tanjakostic.jcleancim.model.UmlAssociation.Direction directed
```

unspecified

```
public static final org.tanjakostic.jcleancim.model.UmlAssociation.Direction
unspecified
```

Methods

values

```
public static UmlAssociation.Direction[] values()
```

valueOf

```
public static UmlAssociation.Direction valueOf(java.lang.String name)
```

org.tanjakostic.jcleancim.model
Class UmlAssociation.Data

```
java.lang.Object
   |
   +-org.tanjakostic.jcleancim.model.UmlAssociation.Data
```

public static class **UmlAssociation.Data**
extends java.lang.Object

Data from the UML model repository specific to `UmlAssociation`.

Constructor Summary

public	`Data(UmlAssociation.Direction direction)` Constructor.

Method Summary

static `UmlAssociation.Data`	`empty()` Returns empty instance; sets default direction to `UmlAssociation.Direction.unspecified`.
`UmlAssociation.Direction`	`getDirection()`

Methods inherited from class `java.lang.Object`
`clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait`

Constructors

Data

`public Data(UmlAssociation.Direction direction)`

Constructor.

Parameters:
`direction`

Methods

empty

`public static UmlAssociation.Data empty()`

Returns empty instance; sets default direction to `UmlAssociation.Direction.unspecified`.

(continued from last page)

getDirection

```
public UmlAssociation.Direction getDirection()
```

org.tanjakostic.jcleancim.model
Class UmlAssociationEnd

```
java.lang.Object
  |-org.tanjakostic.jcleancim.model.AbstractUmlObject
    |-org.tanjakostic.jcleancim.model.UmlAssociationEnd
```

All Implemented Interfaces:
UmlObject

---

public class **UmlAssociationEnd**
extends AbstractUmlObject

UML association end.

This class is more of a helper for `UmlAssociation`. We make it however implement `UmlObject` to be able to use utility methods of `AbstractUmlObject`, but do not store any instance in `UmlModel` - association ends are stored in associations only. Note that after creation, several methods will return null before the containinig association gets created with this instance as one of its ends (`org.tanjakostic.jcleancim.model.UmlAssociation(UmlAssociationEnd, UmlAssociationEnd, UmlObjectData, UmlAssociation.Data)`).

The owner of this end is determined as the owner of the type of the other end, and the nature is the nature of the type of this end. Example: For association A (bRole) --- (aRole) B, if this is aRole, its owner is the owner of A (because it is needed by A), and its nature is the nature of B (because its type is B).

Nested Class Summary

class	`UmlAssociationEnd.Data` UmlAssociationEnd.Data
class	`UmlAssociationEnd.Kind` UmlAssociationEnd.Kind
class	`UmlAssociationEnd.Navigable` UmlAssociationEnd.Navigable

Fields inherited from class `org.tanjakostic.jcleancim.model.AbstractUmlObject`
`CLASS_SEPARATOR`, `NULL_OBJ_NAME`, `PACKAGE_SEPARATOR`

Constructor Summary

public	`UmlAssociationEnd(UmlClass type, UmlObjectData objData, UmlAssociationEnd.Data data)` Constructor.

Method Summary

`UmlAssociation`	`getContainingAssociation()` Returns the association to which this end belongs.
`UmlKind`	`getKind()`

static java.util.List	getKinds(Nature nature) Returns all available classifications (kinds) for association ends.
UmlMultiplicity	getMultiplicity()
Namespace	getNamespace()
Nature	getNature()
UmlAssociationEnd.Navigable	getNavigable()
OwningWg	getOwner()
java.util.Set	getPredefinedTagNames()
java.lang.String	getQualifiedName()
UmlClass	getType() Returns UmlClass used as type for this association end.
boolean	isAggregation()
boolean	isAssociation()
boolean	isComposition()
boolean	isDeprecated() This default implementation returns whether the stereotype string of this object contains the string UmlStereotype.DEPRECATED.
boolean	isInformative() This default implementation returns true if objects steretypes include UmlStereotype#INFORMATIVE. If there are additional criteria for deriving informative status, ensure to invoke this default implementation first.
boolean	isNamedWithoutMultiplicity()
boolean	isOther()
boolean	isSource() Returns whether this end is the source end of the containing association.
boolean	isTarget() Returns whether this end is the target end of the containing association.
java.lang.String	toString()

Methods inherited from class org.tanjakostic.jcleancim.model.AbstractUmlObject

addDeprecAndInf, addTaggedValue, appendRemainingCustomStereotypes, classifyPerScope, classifyPerScopePerTag, classifyPerTag, collectDuplicateDescriptions, collectDuplicateNames, collectForScope, collectNames, collectQNames, findAllForName, findWithSameUuidAndLog, getAlias, getDescription, getHtmlDescription, getId, getKind, getName, getNature, getOwner, getQualifiedName, getSince, getStereotype, getTaggedValues, getUnallowedTagNames, getUuid, getVisibility, initFromTags, isDeprecated, isInformative, saveTags, toShortString, toShortString, validateTag

Methods inherited from class `java.lang.Object`

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface `org.tanjakostic.jcleancim.model.UmlObject`

addTaggedValue, getAlias, getDescription, getHtmlDescription, getId, getKind, getName, getNamespace, getNature, getOwner, getPredefinedTagNames, getQualifiedName, getSince, getStereotype, getTaggedValues, getUnallowedTagNames, getUuid, getVisibility, isDeprecated, isInformative, toShortString

Constructors

UmlAssociationEnd

```
public UmlAssociationEnd(UmlClass type,
                         UmlObjectData objData,
                         UmlAssociationEnd.Data data)
```

Constructor. After creating this object, you may want to add tagged values. In every case, the association that will be initialised from two instances of this, has to use `setContainingAssociation(UmlAssociation)` to correctly set reference to itself.

Parameters:

`type` - class used as type for this association end.
`objData` - common data for any `UmlObject`.
`data` - data proper to `UmlAssociationEnd`.

Methods

getKinds

```
public static java.util.List getKinds(Nature nature)
```

Returns all available classifications (kinds) for association ends.

Parameters:

`nature` - ignored in this method

getContainingAssociation

```
public UmlAssociation getContainingAssociation()
```

Returns the association to which this end belongs.

(continued from last page)

getType

```
public UmlClass getType()
```

Returns `UmlClass` used as type for this association end.

isAssociation

```
public boolean isAssociation()
```

isAggregation

```
public boolean isAggregation()
```

isComposition

```
public boolean isComposition()
```

isOther

```
public boolean isOther()
```

getMultiplicity

```
public UmlMultiplicity getMultiplicity()
```

getNavigable

```
public UmlAssociationEnd.Navigable getNavigable()
```

isNamedWithoutMultiplicity

```
public boolean isNamedWithoutMultiplicity()
```

isSource

```
public boolean isSource()
```

Returns whether this end is the source end of the containing association.

isTarget

```
public boolean isTarget()
```

(continued from last page)

Returns whether this end is the target end of the containing association.

getOwner

`public OwningWg getOwner()`

Before two instances of this are used to create an association, returns null.

getNamespace

`public Namespace getNamespace()`

Returns own namespace initialised from tagged values if not empty. Otherwise, returns null if association end has not yet been added to its association, or association's namespace.

getNature

`public Nature getNature()`

isInformative

`public boolean isInformative()`

This default implementation returns true if objects steretypes include UmlStereotype#INFORMATIVE. If there are additional criteria for deriving informative status, ensure to invoke this default implementation first.

Before two instances of this are used to create an association, returns false.

Association end is considered as informative if any of the following is true:

- association end stereotype includes UmlStereotype#INFORMATIVE,
- association end type is informative,
- association end's other end type is informative,
- containing association is informative.

getKind

`public UmlKind getKind()`

getQualifiedName

`public java.lang.String getQualifiedName()`

Returns qualified name of this association end (i.e., the type of the other association end prepended to the name).

(continued from last page)

isDeprecated

```
public boolean isDeprecated()
```

This default implementation returns whether the stereotype string of this object contains the string `UmlStereotype.DEPRECATED`.

Association end is deprecated if its stereotype contains UmlStereotype#DEPRECATED, or if any of these is deprecated: its type, its other end type, or the containing association.

getPredefinedTagNames

```
public java.util.Set getPredefinedTagNames()
```

toString

```
public java.lang.String toString()
```

org.tanjakostic.jcleancim.model
Class UmlAssociationEnd.Kind

```
java.lang.Object
  +-java.lang.Enum
    +-org.tanjakostic.jcleancim.model.UmlAssociationEnd.Kind
```

All Implemented Interfaces:
UmlKind, java.io.Serializable, java.lang.Comparable

---

public static final class **UmlAssociationEnd.Kind**
extends java.lang.Enum
implements java.lang.Comparable, java.io.Serializable, UmlKind

Kind of aggregation for association end.

Field Summary

public static final	AGGREG
public static final	ASSOC
public static final	COMPOS
public static final	OTHER

Method Summary

static UmlAssociationEnd.Kind	findForValue(java.lang.String value) Returns literal with value if found, OTHER instance otherwise.
java.lang.String	getDesc()
java.lang.String	getLabel()
java.lang.String	getTag()
java.lang.String	getValue()
static UmlAssociationEnd.Kind	valueOf(java.lang.String name)
static UmlAssociationEnd.Kind[]	values()

Methods inherited from class java.lang.Enum

clone, compareTo, equals, finalize, getDeclaringClass, hashCode, name, ordinal, toString, valueOf

Methods inherited from class `java.lang.Object`
clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface `java.lang.Comparable`
compareTo

Methods inherited from interface `org.tanjakostic.jcleancim.model.UmlKind`
getDesc, getLabel, getTag, getValue

Fields

COMPOS

`public static final org.tanjakostic.jcleancim.model.UmlAssociationEnd.Kind` **`COMPOS`**

AGGREG

`public static final org.tanjakostic.jcleancim.model.UmlAssociationEnd.Kind` **`AGGREG`**

ASSOC

`public static final org.tanjakostic.jcleancim.model.UmlAssociationEnd.Kind` **`ASSOC`**

OTHER

`public static final org.tanjakostic.jcleancim.model.UmlAssociationEnd.Kind` **`OTHER`**

Methods

values

`public static UmlAssociationEnd.Kind[]` **`values`**`()`

valueOf

`public static UmlAssociationEnd.Kind` **`valueOf`**`(java.lang.String name)`

findForValue

`public static UmlAssociationEnd.Kind findForValue(java.lang.String value)`

Returns literal with `value` if found, `OTHER` instance otherwise.

getValue

`public java.lang.String getValue()`

getLabel

`public java.lang.String getLabel()`

getTag

`public java.lang.String getTag()`

getDesc

`public java.lang.String getDesc()`

org.tanjakostic.jcleancim.model
Class UmlAssociationEnd.Navigable

```
java.lang.Object
  +-java.lang.Enum
    +-org.tanjakostic.jcleancim.model.UmlAssociationEnd.Navigable
```

All Implemented Interfaces:
java.io.Serializable, java.lang.Comparable

---

public static final class **UmlAssociationEnd.Navigable**
extends java.lang.Enum

Navigability of an association end.

Field Summary

public static final	no
public static final	unspecified
public static final	yes

Method Summary

static UmlAssociationEnd.Navigable	valueOf(java.lang.String name)
static UmlAssociationEnd.Navigable[]	values()

Methods inherited from class `java.lang.Enum`

clone, compareTo, equals, finalize, getDeclaringClass, hashCode, name, ordinal, toString, valueOf

Methods inherited from class `java.lang.Object`

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface `java.lang.Comparable`

compareTo

Fields

(continued from last page)

yes

```
public static final org.tanjakostic.jcleancim.model.UmlAssociationEnd.Navigable yes
```

no

```
public static final org.tanjakostic.jcleancim.model.UmlAssociationEnd.Navigable no
```

unspecified

```
public static final org.tanjakostic.jcleancim.model.UmlAssociationEnd.Navigable
unspecified
```

Methods

values

```
public static UmlAssociationEnd.Navigable[] values()
```

valueOf

```
public static UmlAssociationEnd.Navigable valueOf(java.lang.String name)
```

org.tanjakostic.jcleancim.model
Class UmlAssociationEnd.Data

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.model.UmlAssociationEnd.Data
```

public static class **UmlAssociationEnd.Data**
extends java.lang.Object

Data from the UML model repository specific to `UmlAssociationEnd`.

Constructor Summary

public	`Data(UmlAssociationEnd.Kind kind, UmlMultiplicity multiplicity, UmlAssociationEnd.Navigable navigable)` Constructor.

Method Summary

static `UmlAssociationEnd.Data`	`empty()` Returns empty instance; sets default multiplicity to `UmlMultiplicity.ONE`, kind to `UmlAssociationEnd.Kind.ASSOC`, and direction to `UmlAssociationEnd.Navigable.unspecified`.
`UmlAssociationEnd.Kind`	`getKind()`
`UmlMultiplicity`	`getMultiplicity()`
`UmlAssociationEnd.Navigable`	`getNavigable()`

Methods inherited from class `java.lang.Object`

`clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait`

Constructors

Data

```
public Data(UmlAssociationEnd.Kind kind,
            UmlMultiplicity multiplicity,
            UmlAssociationEnd.Navigable navigable)
```

Constructor.

Parameters:
- `kind`
- `multiplicity`
- `navigable`

(continued from last page)

Methods

empty

`public static UmlAssociationEnd.Data empty()`

Returns empty instance; sets default multiplicity to `UmlMultiplicity.ONE`, kind to `UmlAssociationEnd.Kind.ASSOC`, and direction to `UmlAssociationEnd.Navigable.unspecified`.

getKind

`public UmlAssociationEnd.Kind getKind()`

getMultiplicity

`public UmlMultiplicity getMultiplicity()`

getNavigable

`public UmlAssociationEnd.Navigable getNavigable()`

org.tanjakostic.jcleancim.model
Class UmlAssociationEndPair

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.model.UmlAssociationEndPair
```

public class **UmlAssociationEndPair**
extends java.lang.Object

Helper class, used to hold the two association ends of an association, from the perspective of a UmlClass used as type for those ends. Useful for doc generation, as it gives "this" and "other" end of an association so a class can use "other" end to list its roles (with the other end class) in a similar way it lists its own attributes.

Consider association between classes A and B. Their (qualified) association ends names are (A.bRole, B.aRole) from the perspective of A, and (B.aRole, A.bRole) from the perspective of B.

Here example of a couple of inherited association ends for CIM ConductingEquipment:

```
myEnd: [0..*] EquipmentContainer.Equipments, otherEnd: [0..1] Equipment.EquipmentContainer
myEnd: [0..*] PSRType.PowerSystemResources, otherEnd: [0..1] PowerSystemResource.PSRType
myEnd: [0..1] Measurement.PowerSystemResource, otherEnd: [0..*]
PowerSystemResource.Measurements
```

Method Summary

UmlAssociationEnd	getMyEnd()
UmlAssociationEnd	getOtherEnd()
java.lang.String	toString() Example for inherited association ends of ConductingEquipment:

Methods inherited from class java.lang.Object
clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods

getMyEnd

public UmlAssociationEnd **getMyEnd**()

getOtherEnd

public UmlAssociationEnd **getOtherEnd**()

(continued from last page)

toString

`public java.lang.String **toString**()`

Example for inherited association ends of ConductingEquipment:

```
 myEnd: [0..*] EquipmentContainer.Equipments, otherEnd: [0..1]
Equipment.EquipmentContainer
 myEnd: [1..1] OperationalLimitSet.Equipment, otherEnd: [0..*]
Equipment.OperationalLimitSet
 myEnd: [1..1] ContingencyEquipment.Equipment, otherEnd: [0..*]
Equipment.ContingencyEquipment
 myEnd: [0..*] PSRType.PowerSystemResources, otherEnd: [0..1]
PowerSystemResource.PSRType
 myEnd: [0..1] Measurement.PowerSystemResource, otherEnd: [0..*]
PowerSystemResource.Measurements
 myEnd: [1..1] OperatingShare.PowerSystemResource, otherEnd: [0..*]
PowerSystemResource.OperatingShare
 myEnd: [0..*] PsrList.PowerSystemResources, otherEnd: [0..*]
PowerSystemResource.PsrLists
 myEnd: [1..1] OutageSchedule.PowerSystemResource, otherEnd: [0..1]
PowerSystemResource.OutageSchedule
 myEnd: [0..*] ReportingGroup.PowerSystemResource, otherEnd: [0..*]
PowerSystemResource.ReportingGroup
 myEnd: [1..*] ModelingAuthoritySet.IdentifiedObjects, otherEnd: [0..1]
IdentifiedObject.ModelingAuthoritySet
        /*
```

See Also:
`Object.toString()`

org.tanjakostic.jcleancim.model
Class UmlAttribute

```
java.lang.Object
  +-org.tanjakostic.jcleancim.model.AbstractUmlObject
    +-org.tanjakostic.jcleancim.model.UmlAttribute
```

All Implemented Interfaces:
UmlObject

---

public class **UmlAttribute**
extends AbstractUmlObject

UML attribute or enumeration literal.

Implementation note: We distinguish among kinds of attributes by their `UmlAttribute.Kind`, which implements the `UmlKind` interface and internally piggy-backs kinds of `UmlClass`. Knowing the kinds of attributes allows us to do model validation and also to correctly print documentation (and on the fly, calculate detailed statistics).

A cleaner design would be to effectively create subclasses instead of using the above kinds, but it would be overkill for minor differences in functionality per kind.

Nested Class Summary

class	`UmlAttribute.Data` UmlAttribute.Data
class	`UmlAttribute.Kind` UmlAttribute.Kind

Field Summary

public static final	`DO_MAX_LENGTH` Value: **12**

Fields inherited from class `org.tanjakostic.jcleancim.model.AbstractUmlObject`
`CLASS_SEPARATOR`, `NULL_OBJ_NAME`, `PACKAGE_SEPARATOR`

Method Summary

`UmlConstraint`	`addOwnConstraint(UmlObjectData objData, UmlConstraint.Data data)` Creates attribute constraint from arguments, adds it to this attribute and returns the new constraint.
boolean	`displayEmptyValue()` Returns true if this is an enumeration literal whose name needs to be translated as empty string in SCL XML, or simply an attribute with a default value that again needs to be translated as empty string in SCL XML (and in both Word and XML auto-generated docs).
static java.util.Collection	`findAbbreviationLiterals(java.util.Collection attributes)`

static java.util.Collection	findEnumLiterals(java.util.Collection attributes)
static java.util.Collection	findPresenceConditionLiterals(java.util.Collection attributes)
java.util.Collection	getAllSiblings() Returns all (native and inherited) sibling attributes.
java.lang.String	getArrayBounds() Returns formatted string "minId...maxId" created from attribute constraints if existing, empty string otherwise.
java.lang.String	getArrayBoundsWithBrackets()
java.util.Map	getConstraintValues() Returns map of {name, constraint} pairs defined as attribute constraints.
UmlClass	getContainingClass() Returns class containing this attribute.
java.util.List	getDsPresConditions(UmlClass context) (IEC 61850) Returns derived statistics presence conditions.
int	getEaTypeId() See UmlAttribute.Data.getEaTypeId().
java.lang.String	getEaTypeInfo() Returns known (string) info from EA; useful to display in case the type of this attribute in EA model is not a valid UML class, so the model can be corrected.
java.lang.String	getEaTypeName() See UmlAttribute.Data.getEaTypeName().
java.lang.Integer	getInitialValueAsInteger() Returns the initial value if it can be interpreted as integer, null otherwise.
java.lang.String	getInitValue() See UmlAttribute.Data.getInitValue().
java.lang.String	getInitValueWithPotentialOverrideForSCL() (Special handling for IEC61850) Returns empty string as initial value for the case of an attribute that returns true from displayEmptyValue().
UmlKind	getKind()
static java.util.List	getKinds(Nature nature) Returns available classifications (kinds) for attributes.
UmlMultiplicity	getMultiplicity() See UmlAttribute.Data.getMultiplicity().
NameDecomposition	getNameDecomposition() Equivalent to getNameDecomposition(null).
NameDecomposition	getNameDecomposition(java.util.Map sortedAbbrTerms) In case of a data object (attribute on LN, in IEC61850), returns decomposition of the attribute name to abbreviated terms, null otherwise.

Namespace	getNamespace()
Nature	getNature()
java.util.List	getOwnConstraints() Returns constraints defined on this attribute.
OwningWg	getOwner()
java.util.Set	getPredefinedTagNames()
java.util.List	getPresConditions()
java.lang.String	getQualifiedName()
UmlAttribute	getSiblingToMoveAfter() Returns (native or inherited) sibling attribute whose name is defined as value of the tag UML.TAG_moveAfter if found, null otherwise; in case there are multiple sibling attributes with that same name, returns the first one.
UmlClass	getType() Returns UmlClass used as type of this attribute for a non-literal, null otherwise.
ValueRange	getValueRange() Returns value range if specified, null otherwise.
boolean	hasConstValue() Returns whether the attribute has a constant value (for all instances of the class).
boolean	hasDefaultValue() Returns whether the attribute has a default initial value (semantic: value applies to any instance of a class and can be changed later).
boolean	hasInitialValueAsInteger() Returns whether the initial value (when present) was interpreted as integer.
boolean	hasSuperfluousType() See UmlAttribute.Data.isEaTypeSuperfluous().
boolean	hasValueRange() Returns whether the attribute has a range specified.
boolean	isConditional() Returns whether this attribute has presence condition derived from its containing class.
boolean	isConst() See UmlAttribute.Data.isConst().
boolean	isDO() IEC61850
boolean	isInformative()
boolean	isLiteral() CIM and IEC61850

boolean	isMultivalued() Returns whether this attribute is multivalued.
boolean	isOptional() Returns whether the multiplicity is optional.
boolean	isPublic() Returns whether this attribute is public.
boolean	isStatic() See UmlAttribute.Data.isStatic().
java.lang.String	toString()

Methods inherited from class org.tanjakostic.jcleancim.model.AbstractUmlObject

addDeprecAndInf, addTaggedValue, appendRemainingCustomStereotypes, classifyPerScope, classifyPerScopePerTag, classifyPerTag, collectDuplicateDescriptions, collectDuplicateNames, collectForScope, collectNames, collectQNames, findAllForName, findWithSameUuidAndLog, getAlias, getDescription, getHtmlDescription, getId, getKind, getName, getNature, getOwner, getQualifiedName, getSince, getStereotype, getTaggedValues, getUnallowedTagNames, getUuid, getVisibility, initFromTags, isDeprecated, isInformative, saveTags, toShortString, toShortString, validateTag

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.model.UmlObject

addTaggedValue, getAlias, getDescription, getHtmlDescription, getId, getKind, getName, getNamespace, getNature, getOwner, getPredefinedTagNames, getQualifiedName, getSince, getStereotype, getTaggedValues, getUnallowedTagNames, getUuid, getVisibility, isDeprecated, isInformative, toShortString

Fields

DO_MAX_LENGTH

public static final int **DO_MAX_LENGTH**

Constant value: **12**

Methods

getKinds

public static java.util.List **getKinds**(Nature nature)

Returns available classifications (kinds) for attributes.

Parameters:
nature - ignored in this method.

findEnumLiterals

```
public static java.util.Collection findEnumLiterals(java.util.Collection attributes)
```

findAbbreviationLiterals

```
public static java.util.Collection findAbbreviationLiterals(java.util.Collection
attributes)
```

findPresenceConditionLiterals

```
public static java.util.Collection findPresenceConditionLiterals(java.util.Collection
attributes)
```

getContainingClass

```
public UmlClass getContainingClass()
```

Returns class containing this attribute.

getType

```
public UmlClass getType()
```

Returns `UmlClass` used as type of this attribute for a non-literal, null otherwise.

isConst

```
public boolean isConst()
```

See `UmlAttribute.Data.isConst()`.

isStatic

```
public boolean isStatic()
```

See `UmlAttribute.Data.isStatic()`.

getMultiplicity

```
public UmlMultiplicity getMultiplicity()
```

See `UmlAttribute.Data.getMultiplicity()`.

getInitValue

```
public java.lang.String getInitValue()
```

See `UmlAttribute.Data.getInitValue()`.

(continued from last page)

getInitValueWithPotentialOverrideForSCL

`public java.lang.String` **`getInitValueWithPotentialOverrideForSCL`**`()`

(Special handling for IEC61850) Returns empty string as initial value for the case of an attribute that returns `true` from `displayEmptyValue()`. Otherwise, returns `getInitValue()`.

getEaTypeId

`public int` **`getEaTypeId`**`()`

See `UmlAttribute.Data.getEaTypeId()`.

getEaTypeName

`public java.lang.String` **`getEaTypeName`**`()`

See `UmlAttribute.Data.getEaTypeName()`.

getEaTypeInfo

`public java.lang.String` **`getEaTypeInfo`**`()`

Returns known (string) info from EA; useful to display in case the type of this attribute in EA model is not a valid UML class, so the model can be corrected.

hasSuperfluousType

`public boolean` **`hasSuperfluousType`**`()`

See `UmlAttribute.Data.isEaTypeSuperfluous()`.

isOptional

`public boolean` **`isOptional`**`()`

Returns whether the multiplicity is optional.

isMultivalued

`public boolean` **`isMultivalued`**`()`

Returns whether this attribute is multivalued.

isPublic

`public boolean` **`isPublic`**`()`

Returns whether this attribute is public.

isLiteral

`public boolean` **`isLiteral`**`()`

CIM and IEC61850

(continued from last page)

isDO

`public boolean isDO()`

IEC61850

getPresConditions

`public java.util.List getPresConditions()`

isConditional

`public boolean isConditional()`

Returns whether this attribute has presence condition derived from its containing class.

getDsPresConditions

`public java.util.List getDsPresConditions(UmlClass context)`

(IEC 61850) Returns derived statistics presence conditions.

Parameters:

`context` - class for which this attribute is considered; i.e., it could be native to `context` or inherited (from a class other than `context`).

getNameDecomposition

`public NameDecomposition getNameDecomposition()`

Equivalent to `getNameDecomposition(null)`.

getNameDecomposition

`public NameDecomposition getNameDecomposition(java.util.Map sortedAbbrTerms)`

In case of a data object (attribute on LN, in IEC61850), returns decomposition of the attribute name to abbreviated terms, null otherwise.

Parameters:

`sortedAbbrTerms` - (potentially null) abbreviated terms sorted correctly for comparison; if null, this instance needs to find access to those abbreviated terms internally.

addOwnConstraint

```
public UmlConstraint addOwnConstraint(UmlObjectData objData,
        UmlConstraint.Data data)
```

Creates attribute constraint from arguments, adds it to this attribute and returns the new constraint.

getOwnConstraints

`public java.util.List getOwnConstraints()`

Returns constraints defined on this attribute.

(continued from last page)

getArrayBounds

`public java.lang.String getArrayBounds()`

Returns formatted string "minId...maxId" created from attribute constraints if existing, empty string otherwise. This format is useful for doc generation where a multivalued attribute has bounds specified as constraints (IEC 61850-7-3).

getArrayBoundsWithBrackets

`public java.lang.String getArrayBoundsWithBrackets()`

getConstraintValues

`public java.util.Map getConstraintValues()`

Returns map of {name, constraint} pairs defined as attribute constraints.

hasValueRange

`public boolean hasValueRange()`

Returns whether the attribute has a range specified.

getValueRange

`public ValueRange getValueRange()`

Returns value range if specified, null otherwise.

hasConstValue

`public boolean hasConstValue()`

Returns whether the attribute has a constant value (for all instances of the class).

hasDefaultValue

`public boolean hasDefaultValue()`

Returns whether the attribute has a default initial value (semantic: value applies to any instance of a class and can be changed later).

getInitialValueAsInteger

`public java.lang.Integer getInitialValueAsInteger()`

Returns the initial value if it can be interpreted as integer, null otherwise.

hasInitialValueAsInteger

`public boolean hasInitialValueAsInteger()`

Returns whether the initial value (when present) was interpreted as integer.

(continued from last page)

getSiblingToMoveAfter

`public UmlAttribute getSiblingToMoveAfter()`

Returns (native or inherited) sibling attribute whose name is defined as value of the tag `UML.TAG_moveAfter` if found, null otherwise; in case there are multiple sibling attributes with that same name, returns the first one.

displayEmptyValue

`public boolean displayEmptyValue()`

Returns true if this is an enumeration literal whose name needs to be translated as empty string in SCL XML, or simply an attribute with a default value that again needs to be translated as empty string in SCL XML (and in both Word and XML auto-generated docs).

getAllSiblings

`public java.util.Collection getAllSiblings()`

Returns all (native and inherited) sibling attributes.

getOwner

`public OwningWg getOwner()`

getNamespace

`public Namespace getNamespace()`

Returns own namespace initialised from tagged values if not empty. Otherwise, returns the namespace of the containing class.

getNature

`public Nature getNature()`

Returns the nature of containing clas.

isInformative

`public boolean isInformative()`

This default implementation returns true if objects steretypes include UmlStereotype#INFORMATIVE. If there are additional criteria for deriving informative status, ensure to invoke this default implementation first.

getKind

`public UmlKind getKind()`

getQualifiedName

```
public java.lang.String getQualifiedName()
```

getPredefinedTagNames

```
public java.util.Set getPredefinedTagNames()
```

toString

```
public java.lang.String toString()
```

org.tanjakostic.jcleancim.model
Class UmlAttribute.Kind

```
java.lang.Object
  +-java.lang.Enum
    +-org.tanjakostic.jcleancim.model.UmlAttribute.Kind
```

All Implemented Interfaces:
UmlKind, java.io.Serializable, java.lang.Comparable

---

public static final class **UmlAttribute.Kind**
extends java.lang.Enum
implements java.lang.Comparable, java.io.Serializable, UmlKind

Kinds of UML attributes - correspond mainly to the kind of the class used as the attribute's type.

Implementation note: We piggy-back here the kinds defined for `UmlClass` as much as possible. For XML doc generation we need different tags (generic), while preserving detailed description for statistics.

Field Summary

public static final	ABBR_LITERAL
public static final	ATTRIBUTE
public static final	BASIC
public static final	COMP
public static final	COMP_DA
public static final	COMP_FCDA
public static final	COND_LITERAL
public static final	CTS_DO
public static final	DA
public static final	DO
public static final	DT
public static final	ENUM_DA
public static final	ENUM_DO
public static final	ENUM_FCDA

public static final	ENUM61850
public static final	ENUMCIM
public static final	FCDA
public static final	IF
public static final	LITERAL
public static final	PACKED_BASIC
public static final	PACKED_ENUM
public static final	PACKED_ENUM_DA
public static final	PACKED_ENUM_FCDA
public static final	PACKED_LIST_FCDA
public static final	PACKED_LITERAL
public static final	PACKED_PRIM_DA
public static final	PRIM
public static final	PRIM_DA
public static final	SDO
public static final	STRUCTURED
public static final	TRANS_DO

Method Summary

java.lang.String	getDesc()
java.lang.String	getLabel()
java.lang.String	getTag()
static java.util.List	getUmlKinds() Returns all values as UmlKind list.
java.lang.String	getValue()
static UmlAttribute.Kind	valueOf(java.lang.String name)

static UmlAttribute.Kind[]	values()

Methods inherited from class java.lang.Enum
clone, compareTo, equals, finalize, getDeclaringClass, hashCode, name, ordinal, toString, valueOf

Methods inherited from class java.lang.Object
clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface java.lang.Comparable
compareTo

Methods inherited from interface org.tanjakostic.jcleancim.model.UmlKind
getDesc, getLabel, getTag, getValue

Fields

PRIM

public static final org.tanjakostic.jcleancim.model.UmlAttribute.Kind **PRIM**

DT

public static final org.tanjakostic.jcleancim.model.UmlAttribute.Kind **DT**

COMP

public static final org.tanjakostic.jcleancim.model.UmlAttribute.Kind **COMP**

ENUMCIM

public static final org.tanjakostic.jcleancim.model.UmlAttribute.Kind **ENUMCIM**

LITERAL

public static final org.tanjakostic.jcleancim.model.UmlAttribute.Kind **LITERAL**

(continued from last page)

ATTRIBUTE

public static final org.tanjakostic.jcleancim.model.UmlAttribute.Kind **ATTRIBUTE**

ABBR_LITERAL

public static final org.tanjakostic.jcleancim.model.UmlAttribute.Kind **ABBR_LITERAL**

COND_LITERAL

public static final org.tanjakostic.jcleancim.model.UmlAttribute.Kind **COND_LITERAL**

PACKED_LITERAL

public static final org.tanjakostic.jcleancim.model.UmlAttribute.Kind **PACKED_LITERAL**

IF

public static final org.tanjakostic.jcleancim.model.UmlAttribute.Kind **IF**

BASIC

public static final org.tanjakostic.jcleancim.model.UmlAttribute.Kind **BASIC**

STRUCTURED

public static final org.tanjakostic.jcleancim.model.UmlAttribute.Kind **STRUCTURED**

PACKED_BASIC

public static final org.tanjakostic.jcleancim.model.UmlAttribute.Kind **PACKED_BASIC**

ENUM61850

public static final org.tanjakostic.jcleancim.model.UmlAttribute.Kind **ENUM61850**

PACKED_ENUM

public static final org.tanjakostic.jcleancim.model.UmlAttribute.Kind **PACKED_ENUM**

(continued from last page)

PACKED_ENUM_DA

public static final org.tanjakostic.jcleancim.model.UmlAttribute.Kind **PACKED_ENUM_DA**

ENUM_DA

public static final org.tanjakostic.jcleancim.model.UmlAttribute.Kind **ENUM_DA**

PACKED_PRIM_DA

public static final org.tanjakostic.jcleancim.model.UmlAttribute.Kind **PACKED_PRIM_DA**

PRIM_DA

public static final org.tanjakostic.jcleancim.model.UmlAttribute.Kind **PRIM_DA**

COMP_DA

public static final org.tanjakostic.jcleancim.model.UmlAttribute.Kind **COMP_DA**

DA

public static final org.tanjakostic.jcleancim.model.UmlAttribute.Kind **DA**

PACKED_ENUM_FCDA

public static final org.tanjakostic.jcleancim.model.UmlAttribute.Kind **PACKED_ENUM_FCDA**

ENUM_FCDA

public static final org.tanjakostic.jcleancim.model.UmlAttribute.Kind **ENUM_FCDA**

PACKED_LIST_FCDA

public static final org.tanjakostic.jcleancim.model.UmlAttribute.Kind **PACKED_LIST_FCDA**

(continued from last page)

COMP_FCDA

```
public static final org.tanjakostic.jcleancim.model.UmlAttribute.Kind COMP_FCDA
```

FCDA

```
public static final org.tanjakostic.jcleancim.model.UmlAttribute.Kind FCDA
```

SDO

```
public static final org.tanjakostic.jcleancim.model.UmlAttribute.Kind SDO
```

ENUM_DO

```
public static final org.tanjakostic.jcleancim.model.UmlAttribute.Kind ENUM_DO
```

CTS_DO

```
public static final org.tanjakostic.jcleancim.model.UmlAttribute.Kind CTS_DO
```

TRANS_DO

```
public static final org.tanjakostic.jcleancim.model.UmlAttribute.Kind TRANS_DO
```

DO

```
public static final org.tanjakostic.jcleancim.model.UmlAttribute.Kind DO
```

Methods

values

```
public static UmlAttribute.Kind[] values()
```

valueOf

```
public static UmlAttribute.Kind valueOf(java.lang.String name)
```

(continued from last page)

getUmlKinds

```
public static java.util.List getUmlKinds()
```

Returns all values as UmlKind list.

getValue

```
public java.lang.String getValue()
```

getLabel

```
public java.lang.String getLabel()
```

getTag

```
public java.lang.String getTag()
```

getDesc

```
public java.lang.String getDesc()
```

org.tanjakostic.jcleancim.model
Class UmlAttribute.Data

```
java.lang.Object
   |
   +-org.tanjakostic.jcleancim.model.UmlAttribute.Data
```

public static class **UmlAttribute.Data**
extends java.lang.Object

Data from the UML model repository specific to `UmlAttribute`.

Constructor Summary

public	`Data(boolean isConst, boolean isStatic, UmlMultiplicity multiplicity, java.lang.String initValue, int eaTypeId, java.lang.String eaTypeName, boolean isEaTypeSuperfluous)` Constructor.

Method Summary

static `UmlAttribute.Data`	`empty()` Returns empty instance; sets default multiplicity to `UmlMultiplicity.ONE`.
`int`	`getEaTypeId()`
`java.lang.String`	`getEaTypeName()`
`java.lang.String`	`getInitValue()`
`UmlMultiplicity`	`getMultiplicity()`
`boolean`	`isConst()`
`boolean`	`isEaTypeSuperfluous()`
`boolean`	`isStatic()`

Methods inherited from class `java.lang.Object`

`clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait`

Constructors

(continued from last page)

Data

```
public Data(boolean isConst,
            boolean isStatic,
            UmlMultiplicity multiplicity,
            java.lang.String initValue,
            int eaTypeId,
            java.lang.String eaTypeName,
            boolean isEaTypeSuperfluous)
```

Constructor.

Parameters:

`isConst`
`isStatic`
`multiplicity`
`initValue`
`eaTypeId`
`eaTypeName`
`isEaTypeSuperfluous`

Methods

empty

```
public static UmlAttribute.Data empty()
```

Returns empty instance; sets default multiplicity to `UmlMultiplicity.ONE`.

isConst

```
public boolean isConst()
```

isStatic

```
public boolean isStatic()
```

getMultiplicity

```
public UmlMultiplicity getMultiplicity()
```

getInitValue

```
public java.lang.String getInitValue()
```

getEaTypeId

```
public int getEaTypeId()
```

getEaTypeName

```
public java.lang.String getEaTypeName()
```

isEaTypeSuperfluous

```
public boolean isEaTypeSuperfluous()
```

org.tanjakostic.jcleancim.model
Class UmlClass

```
java.lang.Object
  |-org.tanjakostic.jcleancim.model.AbstractUmlObject
    |-org.tanjakostic.jcleancim.model.UmlStructure
      |-org.tanjakostic.jcleancim.model.UmlClass
```

All Implemented Interfaces:
UmlObject

---

public class **UmlClass**
extends UmlStructure

UML class, interface or enumerated type. In addition to UML features specific to classes, it inherits implementation for features common with UML packages from `UmlStructure` (so we avoid code duplication).

Implementation note: We distinguish among kinds of classes by their `UmlClass.CimKind` or `UmlClass.Iec61850Kind`, where both these types implement the common `UmlKind` interface. Knowing the kinds of classes allows us to do model validation and also to correctly print documentation (and on the fly, calculate detailed statistics).

A cleaner design would be to effectively create subclasses instead of using the above kinds, but it would be overkill for minor differences in functionality per kind.

Nested Class Summary

class	`UmlClass.CimKind` UmlClass.CimKind
class	`UmlClass.Data` UmlClass.Data
class	`UmlClass.Iec61850Kind` UmlClass.Iec61850Kind
class	`UmlClass.InheritedKind` UmlClass.InheritedKind

Field Summary

public static final	`SUPER_COMP_CDC` Value: **`ComposedCDC`**
public static final	`SUPER_LN` Value: **`LN`**
public static final	`SUPER_PRIM_CDC` Value: **`PrimitiveCDC`**
public static final	`TAG_FUNCTIONS` Value: **`Functions`**

public static final	TAG_PRESENCE_CONDITIONS Value: **PresenceConditions**

Fields inherited from class org.tanjakostic.jcleancim.model.AbstractUmlObject
CLASS_SEPARATOR, NULL_OBJ_NAME, PACKAGE_SEPARATOR

Constructor Summary

public	UmlClass(UmlPackage containingPackage, UmlObjectData objData, UmlClass.Data data) Creates a class without superclasses; convenient for stereotyped and root classes.
public	UmlClass(UmlPackage containingPackage, java.util.Collection superclasses, UmlObjectData objData, UmlClass.Data data) Creates the instance and adds itself to the containingPackage, and as child to every object from superclasses.

Method Summary

UmlAssociation	addAssociation(UmlAssociationEnd sourceEnd, UmlAssociationEnd targetEnd, UmlObjectData objData, UmlAssociation.Data data) Creates from arguments an association with this as type of sourceEnd, adds it to both types of association ends and to the model, and returns the newly created object.
UmlAttribute	addAttribute(UmlClass type, UmlObjectData objData, UmlAttribute.Data data) Creates from arguments an attribute or enumeration literal, adds it to itself and to the model, populates afferent/efferent collections for this and for type, and returns the newly created object.
UmlConstraint	addConstraint(UmlObjectData objData, UmlConstraint.Data data) Creates from arguments a constraint, adds it to itself, and returns the newly created object.
UmlOperation	addOperation(UmlClass returnType, UmlObjectData objData, UmlOperation.Data data) Creates from arguments an operation, adds it to itself and to the model, populates afferent/efferent collections for this and for non-null returnType, and returns the newly created object.
java.util.Collection	collectDependencyEfferentClasses() Returns all classes that I depend on through an explicit UML dependency in the model.
java.util.Set	findAttributes(java.lang.String attrName) Returns (native) attributes with attrName.
java.util.Set	findAttributes(java.lang.String attrName, UmlClass.InheritedKind inh) Returns attributes with attrName selectively, according to inheritance criterion inh.
java.util.Set	findAttributes(UmlClass attrType) Returns (native) attributes whose type is attrType.
java.util.Set	findAttributes(UmlClass attrType, UmlClass.InheritedKind inh) Returns attributes of type attrType selectively, according to inheritance criterion inh.
java.util.Map	findAttributesPerInitialValue() Returns (native) attributes indexed per their initial value.

java.util.Set	findInitialValuesOrdered() Returns (alphabetically) ordered initial values for (native) attributes; empty list in case there are no initial values.
java.util.List	getAllSuperclassesFlattened(boolean skipInformative) Starting from direct superclasses, returns all the superclasses up to all the roots (no explicit knowledge of multiple inheritance or branching due to it).
java.util.List	getAssociationEndPairs() Returns pairs of association ends, from the perspective of this class (this end vs.
java.util.Collection	getAssociations()
java.util.Collection	getAttributeAfferentClasses() Returns classes that have attributes that use me as their type.
java.util.Collection	getAttributeEfferentClasses() Returns classes that my attributes use as their type.
java.util.Collection	getAttributes() Returns native attributes.
java.lang.String	getCdcId() Initialised from tagged value, applicable to 61850 CDC classes; null if no tag defined.
static java.util.Map	getCimDataTypeMinSpec() Minimum requirement for a valid UmlStereotype.CIMDATATYPE: key is attribute name (AbstractUmlObject.getName()) and value is the kind of its type ((UmlAttribute.getKind())).
java.util.Map	getConstraints()
UmlStructure	getContainer()
UmlPackage	getContainingPackage()
java.lang.String	getIecRef() Initialised from tagged value, applicable to IEC61850-5 classes only; null if no tag defined.
java.lang.String	getIeeeRef() Initialised from tagged value, applicable to IEC61850-5 classes only; null if no tag defined.
java.util.List	getInheritedAssociationEndPairs() Returns inherited pairs of association ends, from the perspective of this class (this end vs.
java.util.Collection	getInheritedAssociations()
java.util.Set	getInheritedAttributes() Returns inherited attributes.
java.util.Set	getInheritedOperations() Returns inherited operations.
java.util.List	getInheritedOtherSideAssociationEnds() Returns inherited association ends with other classes.
UmlKind	getKind()

static java.util.List	getKinds(Nature nature) Returns available classifications (kinds) for classes with nature.
UmlModel	getModel()
Namespace	getNamespace()
Nature	getNature()
java.lang.String	getOldName() Initialised from tagged value, applicable to 61850 7-2 classes; null if no tag defined.
java.util.Collection	getOperationAfferentClasses() Returns classes that have operation parameters that use me as their type.
java.util.Collection	getOperationEfferentClasses() Returns classes that my operation parameters and exceptions use as their type.
java.util.Collection	getOperations() Returns native operations.
java.util.List	getOtherSideAssociationEnds() Returns association ends with other classes.
OwningWg	getOwner()
java.util.Set	getPredefinedTagNames()
java.lang.String	getQualifiedName()
java.lang.String	getRsName() Initialised from tagged value, applicable to IEC61850-5 classes only; null if no tag defined.
java.util.Collection	getSubclasses() Returns direct subclasses of this class.
java.util.Collection	getSuperclasses() Returns direct superclasses of this class.
boolean	hasSuperclass(java.lang.String supName) Returns whether supName is one of superclasses in the inheritance chain.
boolean	inheritsFromStatisticsLN() (61850) Returns whether this class inherits from org.tanjakostic.jcleancim.model.UML#StatisticsLN
boolean	is74LN() IEC61850 - e.g., StatisticsLN or LPHD
boolean	isAbbreviationEnumeration() IEC 61850
boolean	isAbstract()

boolean	isAdmin() (61850) Returns whether this class has stereotype org.tanjakostic.jcleancim.model.UmlStereotype#ADMIN
boolean	isAnyCDC() IEC 61850 (like CDC or SPS)
boolean	isAnyDA() IEC 61850 (like DA or Vector)
boolean	isAnyFCDA() IEC 61850 (like FCDA_ST_dchg or INT32_ST_dchg)
boolean	isAnyLN() IEC 61850
boolean	isAssociationClass()
boolean	isBasic() IEC 61850 (P_*)
boolean	isClass() CIM non-root class without stereotype.
boolean	isCodedEnum() IEC 61850
boolean	isCodedEnumDA() IEC 61850
boolean	isCodedEnumFCDA() IEC 61850 (like PackedEnumFCDA_ST_dchg or DpStatus_ST_dchg)
boolean	isComposedCDC() IEC 61850 (like ComposedCDC or WYE)
boolean	isComposedDA() IEC 61850
boolean	isComposedFCDA() IEC 61850 (like Analog_MX_dchg)
boolean	isCompound() CIM
boolean	isConditionEnumeration() IEC 61850
boolean	isDatatype() CIM
boolean	isEaEnumeration()
boolean	isEaInterface()
boolean	isEaLeafPropSet()

boolean	isEaPersistentPropSet()
boolean	isEaRootPropSet()
boolean	isEnumCDC() IEC 61850 (like ENS and its subtypes, derived from SPS)
boolean	isEnumDA() IEC 61850
boolean	isEnumeratedType() Returns whether this is an enumerated type.
boolean	isEnumeration() CIM and IEC61850 - simple enumeration with no other stereotypes
boolean	isEnumFCDA() IEC 61850 (like EnumDA_ST_dchg or CurveChar_SP_dchg)
boolean	isFCDA() IEC 61850 (FCDA from the meta-model)
boolean	isFrom72() IEC 61850 - returns whether the class is from the model supporting IEC 61850-7-2.
boolean	isFromMetaModel() IEC 61850 - returns whether the class is from the meta-model (package and subpackages).
boolean	isFunction() IEC 61850
boolean	isInformative()
boolean	isInterface() IEC 61850
boolean	isNamespaceClass()
boolean	isNullClass()
boolean	isOrHasSuperclass(java.lang.String name) Returns whether name is this class or one of its superclasses in the inheritance chain.
boolean	isOrInheritsFromStatisticsLN() (61850) Returns whether this class is itself or inherits from org.tanjakostic.jcleancim.model.UML#StatisticsLN
boolean	isOther() IEC 61850
boolean	isPackedList() IEC 61850
boolean	isPackedListDA() IEC 61850

boolean	isPackedListFCDA() IEC 61850 (like Quality_ST_dchg)
boolean	isPrimitive() CIM
boolean	isPrimitiveCDC() IEC 61850 (like PrimitiveCDC or SPS)
boolean	isPrimitiveDA() IEC 61850
boolean	isSelfInherited()
boolean	isStructured() IEC 61850 (S_*); at present, S_Originator only
boolean	isTrackingDerivedCDC() IEC 61850 (like CTSINT32, derived from CTS)
boolean	isTransientCDC() IEC 61850 (like SPCTransient, derived from SPC)
boolean	isUnknown() IEC 61850 - with an unknown stereotype
boolean	isUsableForStatistics() (61850) Returns whether this class or any of its superclasses has stereotype org.tanjakostic.jcleancim.model.UmlStereotype#STATISTICS
boolean	isUsedAsTypeForAttributes()
boolean	isVersionClass()
boolean	isWithOldDatatypeStereotype()
boolean	needsAlias() (61850) Returns whether this class needs alias, for doc generation purposes.
boolean	needsTags() (61850) Returns whether this class needs tagged values, for doc generation purposes.
java.lang.String	toString()

Methods inherited from class org.tanjakostic.jcleancim.model.UmlStructure

addDependency, addDiagram, addSkippedUmlItem, collectDependencyAfferentStructures, collectDependencyEfferentStructures, collectMyAndParentsDependencyEfferentStructures, getContainer, getDependenciesAsSource, getDependenciesAsTarget, getDiagrams, getKind, getModel, getNature, getOwner, getQualifiedName, getSkippedUmlItems, isSelfDependent

Methods inherited from class org.tanjakostic.jcleancim.model.AbstractUmlObject

addDeprecAndInf, addTaggedValue, appendRemainingCustomStereotypes, classifyPerScope, classifyPerScopePerTag, classifyPerTag, collectDuplicateDescriptions, collectDuplicateNames, collectForScope, collectNames, collectQNames, findAllForName, findWithSameUuidAndLog, getAlias, getDescription, getHtmlDescription, getId, getKind, getName, getNature, getOwner, getQualifiedName, getSince, getStereotype, getTaggedValues, getUnallowedTagNames, getUuid, getVisibility, initFromTags, isDeprecated, isInformative, saveTags, toShortString, toShortString, validateTag

Methods inherited from class `java.lang.Object`

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface `org.tanjakostic.jcleancim.model.UmlObject`

addTaggedValue, getAlias, getDescription, getHtmlDescription, getId, getKind, getName, getNamespace, getNature, getOwner, getPredefinedTagNames, getQualifiedName, getSince, getStereotype, getTaggedValues, getUnallowedTagNames, getUuid, getVisibility, isDeprecated, isInformative, toShortString

Fields

SUPER_PRIM_CDC

`public static final java.lang.String` **`SUPER_PRIM_CDC`**

Constant value: **`PrimitiveCDC`**

SUPER_COMP_CDC

`public static final java.lang.String` **`SUPER_COMP_CDC`**

Constant value: **`ComposedCDC`**

SUPER_LN

`public static final java.lang.String` **`SUPER_LN`**

Constant value: **`LN`**

TAG_FUNCTIONS

`public static final java.lang.String` **`TAG_FUNCTIONS`**

Constant value: **`Functions`**

TAG_PRESENCE_CONDITIONS

`public static final java.lang.String` **`TAG_PRESENCE_CONDITIONS`**

Constant value: **`PresenceConditions`**

Constructors

UmlClass

```
public UmlClass(UmlPackage containingPackage,
                UmlObjectData objData,
                UmlClass.Data data)
```

Creates a class without superclasses; convenient for stereotyped and root classes. See `UmlClass(UmlPackage, Collection, UmlObjectData, UmlClass.Data)`.

UmlClass

```
public UmlClass(UmlPackage containingPackage,
                java.util.Collection superclasses,
                UmlObjectData objData,
                UmlClass.Data data)
```

Creates the instance and adds itself to the `containingPackage`, and as child to every object from `superclasses`. After creating this object, you may want to add tagged values, constraints, attributes, associations and operations (as well as other objects - see `org.tanjakostic.jcleancim.model.UmlStructure(UmlObjectData, UmlStructure.Data)`).

Parameters:

`containingPackage` - parent UML package.
`superclasses` - list of superclasses; could be empty but not null.
`objData` - common data for any `UmlObject`.
`data` - data proper to `UmlClass`.

Methods

getCimDataTypeMinSpec

```
public static java.util.Map getCimDataTypeMinSpec()
```

Minimum requirement for a valid `UmlStereotype.CIMDATATYPE`: key is attribute name (`AbstractUmlObject.getName()`) and value is the kind of its type ((`UmlAttribute.getKind()`)).

getKinds

```
public static java.util.List getKinds(Nature nature)
```

Returns available classifications (kinds) for classes with `nature`.

getAllSuperclassesFlattened

```
public java.util.List getAllSuperclassesFlattened(boolean skipInformative)
```

Starting from direct superclasses, returns all the superclasses up to all the roots (no explicit knowledge of multiple inheritance or branching due to it).

getContainingPackage

```
public UmlPackage getContainingPackage()
```

(continued from last page)

isNullClass

```
public boolean isNullClass()
```

isVersionClass

```
public boolean isVersionClass()
```

isNamespaceClass

```
public boolean isNamespaceClass()
```

isAbstract

```
public boolean isAbstract()
```

isWithOldDatatypeStereotype

```
public boolean isWithOldDatatypeStereotype()
```

isEaPersistentPropSet

```
public boolean isEaPersistentPropSet()
```

isEaLeafPropSet

```
public boolean isEaLeafPropSet()
```

isEaRootPropSet

```
public boolean isEaRootPropSet()
```

isEaInterface

```
public boolean isEaInterface()
```

isEaEnumeration

```
public boolean isEaEnumeration()
```

(continued from last page)

isAssociationClass

`public boolean **isAssociationClass**()`

isSelfInherited

`public boolean **isSelfInherited**()`

getRsName

`public java.lang.String **getRsName**()`

Initialised from tagged value, applicable to IEC61850-5 classes only; null if no tag defined.

getIeeeRef

`public java.lang.String **getIeeeRef**()`

Initialised from tagged value, applicable to IEC61850-5 classes only; null if no tag defined.

getIecRef

`public java.lang.String **getIecRef**()`

Initialised from tagged value, applicable to IEC61850-5 classes only; null if no tag defined.

getCdcId

`public java.lang.String **getCdcId**()`

Initialised from tagged value, applicable to 61850 CDC classes; null if no tag defined.

getOldName

`public java.lang.String **getOldName**()`

Initialised from tagged value, applicable to 61850 7-2 classes; null if no tag defined.

needsAlias

`public boolean **needsAlias**()`

(61850) Returns whether this class needs alias, for doc generation purposes.

needsTags

`public boolean **needsTags**()`

(61850) Returns whether this class needs tagged values, for doc generation purposes.

(continued from last page)

isAdmin

`public boolean isAdmin()`

(61850) Returns whether this class has stereotype org.tanjakostic.jcleancim.model.UmlStereotype#ADMIN

isUsableForStatistics

`public boolean isUsableForStatistics()`

(61850) Returns whether this class or any of its superclasses has stereotype org.tanjakostic.jcleancim.model.UmlStereotype#STATISTICS

inheritsFromStatisticsLN

`public boolean inheritsFromStatisticsLN()`

(61850) Returns whether this class inherits from org.tanjakostic.jcleancim.model.UML#StatisticsLN

isOrInheritsFromStatisticsLN

`public boolean isOrInheritsFromStatisticsLN()`

(61850) Returns whether this class is itself or inherits from org.tanjakostic.jcleancim.model.UML#StatisticsLN

isPrimitive

`public boolean isPrimitive()`

CIM

isEnumeration

`public boolean isEnumeration()`

CIM and IEC61850 - simple enumeration with no other stereotypes

isDatatype

`public boolean isDatatype()`

CIM

isCompound

`public boolean isCompound()`

CIM

isClass

`public boolean isClass()`

CIM non-root class without stereotype.

isInterface

`public boolean isInterface()`

(continued from last page)

IEC 61850

isCodedEnum

`public boolean isCodedEnum()`

IEC 61850

isAbbreviationEnumeration

`public boolean isAbbreviationEnumeration()`

IEC 61850

isConditionEnumeration

`public boolean isConditionEnumeration()`

IEC 61850

isPackedList

`public boolean isPackedList()`

IEC 61850

isBasic

`public boolean isBasic()`

IEC 61850 (P_*)

isStructured

`public boolean isStructured()`

IEC 61850 (S_*); at present, S_Originator only

isCodedEnumDA

`public boolean isCodedEnumDA()`

IEC 61850

isEnumDA

`public boolean isEnumDA()`

IEC 61850

isPackedListDA

`public boolean isPackedListDA()`

IEC 61850

(continued from last page)

isPrimitiveDA

`public boolean isPrimitiveDA()`

IEC 61850

isComposedDA

`public boolean isComposedDA()`

IEC 61850

isCodedEnumFCDA

`public boolean isCodedEnumFCDA()`

IEC 61850 (like PackedEnumFCDA_ST_dchg or DpStatus_ST_dchg)

isEnumFCDA

`public boolean isEnumFCDA()`

IEC 61850 (like EnumDA_ST_dchg or CurveChar_SP_dchg)

isPackedListFCDA

`public boolean isPackedListFCDA()`

IEC 61850 (like Quality_ST_dchg)

isComposedFCDA

`public boolean isComposedFCDA()`

IEC 61850 (like Analog_MX_dchg)

isPrimitiveCDC

`public boolean isPrimitiveCDC()`

IEC 61850 (like PrimitiveCDC or SPS)

isEnumCDC

`public boolean isEnumCDC()`

IEC 61850 (like ENS and its subtypes, derived from SPS)

isComposedCDC

`public boolean isComposedCDC()`

IEC 61850 (like ComposedCDC or WYE)

isTransientCDC

`public boolean isTransientCDC()`

(continued from last page)

IEC 61850 (like SPCTransient, derived from SPC)

isTrackingDerivedCDC

`public boolean` **`isTrackingDerivedCDC`**`()`

IEC 61850 (like CTSINT32, derived from CTS)

isAnyDA

`public boolean` **`isAnyDA`**`()`

IEC 61850 (like DA or Vector)

isAnyFCDA

`public boolean` **`isAnyFCDA`**`()`

IEC 61850 (like FCDA_ST_dchg or INT32_ST_dchg)

isFCDA

`public boolean` **`isFCDA`**`()`

IEC 61850 (FCDA from the meta-model)

isAnyCDC

`public boolean` **`isAnyCDC`**`()`

IEC 61850 (like CDC or SPS)

isAnyLN

`public boolean` **`isAnyLN`**`()`

IEC 61850

is74LN

`public boolean` **`is74LN`**`()`

IEC61850 - e.g., StatisticsLN or LPHD

isFunction

`public boolean` **`isFunction`**`()`

IEC 61850

isOther

`public boolean` **`isOther`**`()`

IEC 61850

(continued from last page)

isUnknown

`public boolean isUnknown()`

IEC 61850 - with an unknown stereotype

isUsedAsTypeForAttributes

`public boolean isUsedAsTypeForAttributes()`

isEnumeratedType

`public boolean isEnumeratedType()`

Returns whether this is an enumerated type. For CIM, the result is the same as by `isEnumeration()`, while for IEC 61850, this method returns true also for those enumerations that have additional stereotype.

isFrom72

`public boolean isFrom72()`

IEC 61850 - returns whether the class is from the model supporting IEC 61850-7-2.

isFromMetaModel

`public boolean isFromMetaModel()`

IEC 61850 - returns whether the class is from the meta-model (package and subpackages).

collectDependencyEfferentClasses

`public java.util.Collection collectDependencyEfferentClasses()`

Returns all classes that I depend on through an explicit UML dependency in the model.

getSuperclasses

`public java.util.Collection getSuperclasses()`

Returns direct superclasses of this class.

getSubclasses

`public java.util.Collection getSubclasses()`

Returns direct subclasses of this class.

hasSuperclass

`public boolean hasSuperclass(java.lang.String supName)`

Returns whether `supName` is one of superclasses in the inheritance chain.

isOrHasSuperclass

`public boolean isOrHasSuperclass(java.lang.String name)`

(continued from last page)

Returns whether `name` is this class or one of its superclasses in the inheritance chain.

addOperation

```
public UmlOperation addOperation(UmlClass returnType,
        UmlObjectData objData,
        UmlOperation.Data data)
```

Creates from arguments an operation, adds it to itself and to the model, populates afferent/efferent collections for this and for non-null `returnType`, and returns the newly created object. After that, you may want to add tagged values, exceptions, and parameters to the new operation.

For `returnType` you should provide null in case the operation returns void. If the return type could not be determined from the model repository, provide the "null" class obtained with `UmlModel.getNullClasses()` as argument for `returnType`.

In case the operation with the same UUID has already been added, returns the existing operation immediately.

Parameters:

`returnType` - return type, null if the operation returns void
`objData`
`data`

Throws:

`IllegalArgumentException` - if `returnType` is null but data.kind says it does not return void, or if this and non-null `returnType` are from different models.

getOperations

```
public java.util.Collection getOperations()
```

Returns native operations.

getInheritedOperations

```
public java.util.Set getInheritedOperations()
```

Returns inherited operations.

getOperationAfferentClasses

```
public java.util.Collection getOperationAfferentClasses()
```

Returns classes that have operation parameters that use me as their type.

getOperationEfferentClasses

```
public java.util.Collection getOperationEfferentClasses()
```

Returns classes that my operation parameters and exceptions use as their type.

addConstraint

```
public UmlConstraint addConstraint(UmlObjectData objData,
        UmlConstraint.Data data)
```

Creates from arguments a constraint, adds it to itself, and returns the newly created object. In case the constraint with the same UUID has already been added, returns the existing item immediately. In case the constraint with the same name has already been added, overwrites the old constraint.

(continued from last page)

Parameters:
`objData`
`data`

getConstraints

```
public java.util.Map getConstraints()
```

addAttribute

```
public UmlAttribute addAttribute(UmlClass type,
        UmlObjectData objData,
        UmlAttribute.Data data)
```

Creates from arguments an attribute or enumeration literal, adds it to itself and to the model, populates afferent/efferent collections for this and for `type`, and returns the newly created object. After that, you may want to add tagged values and constraints to the new attribute.

For `type` that cannot be determined from the model repository, provide the "null" class obtained with `UmlModel.getNullClasses()` as argument.

In case the attribute with the same UUID has already been added, returns the existing attribute immediately.

Parameters:
`type` - type of the attribute if it is not a literal, null otherwise.
`objData`
`data`

Throws:
`IllegalArgumentException` - if this and non-null `type` are from different models,or if `type` is null and this not an enumerated type.

getAttributes

```
public java.util.Collection getAttributes()
```

Returns native attributes.

getInheritedAttributes

```
public java.util.Set getInheritedAttributes()
```

Returns inherited attributes.

getAttributeAfferentClasses

```
public java.util.Collection getAttributeAfferentClasses()
```

Returns classes that have attributes that use me as their type.

getAttributeEfferentClasses

```
public java.util.Collection getAttributeEfferentClasses()
```

Returns classes that my attributes use as their type.

(continued from last page)

findAttributes

```
public java.util.Set findAttributes(java.lang.String attrName)
```

Returns (native) attributes with `attrName`.

findAttributes

```
public java.util.Set findAttributes(UmlClass attrType)
```

Returns (native) attributes whose type is `attrType`.

findAttributes

```
public java.util.Set findAttributes(java.lang.String attrName,
        UmlClass.InheritedKind inh)
```

Returns attributes with `attrName` selectively, according to inheritance criterion `inh`.

findAttributes

```
public java.util.Set findAttributes(UmlClass attrType,
        UmlClass.InheritedKind inh)
```

Returns attributes of type `attrType` selectively, according to inheritance criterion `inh`.

findAttributesPerInitialValue

```
public java.util.Map findAttributesPerInitialValue()
```

Returns (native) attributes indexed per their initial value.

findInitialValuesOrdered

```
public java.util.Set findInitialValuesOrdered()
```

Returns (alphabetically) ordered initial values for (native) attributes; empty list in case there are no initial values.

addAssociation

```
public UmlAssociation addAssociation(UmlAssociationEnd sourceEnd,
        UmlAssociationEnd targetEnd,
        UmlObjectData objData,
        UmlAssociation.Data data)
```

Creates from arguments an association with this as type of `sourceEnd`, adds it to both types of association ends and to the model, and returns the newly created object. After that, you may want to add tagged values to the new association.

In case the association with the same UUID has already been added to either type of association end, returns the existing association immediately.

It is the responsibility of the caller to call this method on the source end's type.

Parameters:

`sourceEnd`
`targetEnd`
`objData`
`data`

Throws:

(continued from last page)

`IllegalArgumentException` - if the types of `sourceEnd` and `targetEnd` are from different models, or if this is not the type of `sourceEnd`.

getAssociations

`public java.util.Collection` **`getAssociations`**`()`

getInheritedAssociations

`public java.util.Collection` **`getInheritedAssociations`**`()`

getAssociationEndPairs

`public java.util.List` **`getAssociationEndPairs`**`()`

Returns pairs of association ends, from the perspective of this class (this end vs. other end) - convenient for documentation generation.

See Also:
`UmlAssociationEndPair`

getInheritedAssociationEndPairs

`public java.util.List` **`getInheritedAssociationEndPairs`**`()`

Returns inherited pairs of association ends, from the perspective of this class (this end vs. other end) - convenient for documentation generation.

See Also:
`UmlAssociationEndPair`

getOtherSideAssociationEnds

`public java.util.List` **`getOtherSideAssociationEnds`**`()`

Returns association ends with other classes.

getInheritedOtherSideAssociationEnds

`public java.util.List` **`getInheritedOtherSideAssociationEnds`**`()`

Returns inherited association ends with other classes.

getModel

`public` `UmlModel` **`getModel`**`()`

Returns the model this structure belongs to.

getContainer

`public` `UmlStructure` **`getContainer`**`()`

Returns containing structure, null in case this is the model package.

getOwner

public OwningWg **getOwner**()

getNamespace

public Namespace **getNamespace**()

Returns own namespace initialised from tagged values if not empty. Otherwise, returns the namespace of the containing package.

getNature

public Nature **getNature**()

isInformative

public boolean **isInformative**()

This default implementation returns true if objects steretypes include UmlStereotype#INFORMATIVE. If there are additional criteria for deriving informative status, ensure to invoke this default implementation first.

getKind

public UmlKind **getKind**()

This implementation returns either the CIM of the IEC61850 kind, depending on the nature of the class.

getQualifiedName

public java.lang.String **getQualifiedName**()

getPredefinedTagNames

public java.util.Set **getPredefinedTagNames**()

toString

public java.lang.String **toString**()

org.tanjakostic.jcleancim.model
Class UmlClass.CimKind

```
java.lang.Object
  +-java.lang.Enum
    +-org.tanjakostic.jcleancim.model.UmlClass.CimKind
```

All Implemented Interfaces:
UmlKind, java.io.Serializable, java.lang.Comparable

---

public static final class **UmlClass.CimKind**
extends java.lang.Enum
implements java.lang.Comparable, java.io.Serializable, UmlKind

Kind of the UML class for CIM domain.

Field Summary

public static final	CLASS
public static final	COMP
public static final	DT
public static final	ENUM
public static final	NULL_CIM
public static final	PRIM
public static final	ROOT_CLASS

Method Summary

java.lang.String	getDesc()
java.lang.String	getLabel()
java.lang.String	getTag()
static java.util.List	getUmlKinds() Returns all values as UmlKind list.
java.lang.String	getValue()
static UmlClass.CimKind	valueOf(java.lang.String name)

static UmlClass.CimKind[]	values()

Methods inherited from class `java.lang.Enum`

clone, compareTo, equals, finalize, getDeclaringClass, hashCode, name, ordinal, toString, valueOf

Methods inherited from class `java.lang.Object`

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface `java.lang.Comparable`

compareTo

Methods inherited from interface `org.tanjakostic.jcleancim.model.UmlKind`

getDesc, getLabel, getTag, getValue

Fields

PRIM

public static final org.tanjakostic.jcleancim.model.UmlClass.CimKind **PRIM**

ENUM

public static final org.tanjakostic.jcleancim.model.UmlClass.CimKind **ENUM**

DT

public static final org.tanjakostic.jcleancim.model.UmlClass.CimKind **DT**

COMP

public static final org.tanjakostic.jcleancim.model.UmlClass.CimKind **COMP**

ROOT_CLASS

public static final org.tanjakostic.jcleancim.model.UmlClass.CimKind **ROOT_CLASS**

(continued from last page)

CLASS

`public static final org.tanjakostic.jcleancim.model.UmlClass.CimKind CLASS`

NULL_CIM

`public static final org.tanjakostic.jcleancim.model.UmlClass.CimKind NULL_CIM`

Methods

values

`public static UmlClass.CimKind[] values()`

valueOf

`public static UmlClass.CimKind valueOf(java.lang.String name)`

getUmlKinds

`public static java.util.List getUmlKinds()`

Returns all values as UmlKind list.

getValue

`public java.lang.String getValue()`

getLabel

`public java.lang.String getLabel()`

getTag

`public java.lang.String getTag()`

getDesc

`public java.lang.String getDesc()`

org.tanjakostic.jcleancim.model
Class UmlClass.Iec61850Kind

```
java.lang.Object
  |-java.lang.Enum
    |-org.tanjakostic.jcleancim.model.UmlClass.Iec61850Kind
```

All Implemented Interfaces:
UmlKind, java.io.Serializable, java.lang.Comparable

---

public static final class **UmlClass.Iec61850Kind**
extends java.lang.Enum
implements java.lang.Comparable, java.io.Serializable, UmlKind

Kind of the UML class for IEC 61850 domain.

Field Summary

public static final	ABBR_ENUM
public static final	BASIC
public static final	CODED_ENUM_FCDA
public static final	COMP_CDC
public static final	COMP_DA
public static final	COMP_FCDA
public static final	COND_ENUM
public static final	CTS_CDC
public static final	ENUM
public static final	ENUM_CDC
public static final	ENUM_DA
public static final	ENUM_FCDA
public static final	FCDA
public static final	FUNCTION
public static final	IF

public static final	LN
public static final	NULL_61850
public static final	OTHER_61850
public static final	PACKED_BASIC
public static final	PACKED_ENUM
public static final	PACKED_ENUM_DA
public static final	PACKED_LIST_FCDA
public static final	PACKED_PRIM_DA
public static final	PRIM_CDC
public static final	PRIM_DA
public static final	STRUCTURED 6.6.3.17 Originator (S_Originator unknown 61850)
public static final	TRANS_CDC
public static final	UNKNOWN_61850

Method Summary

java.lang.String	getDesc()
java.lang.String	getLabel()
java.lang.String	getTag()
static java.util.List	getUmlKinds() Returns all values as UmlKind list.
java.lang.String	getValue()
static UmlClass.Iec61850Kind	valueOf(java.lang.String name)
static UmlClass.Iec61850Kind[]	values()

Methods inherited from class `java.lang.Enum`

clone, compareTo, equals, finalize, getDeclaringClass, hashCode, name, ordinal, toString, valueOf

Methods inherited from class `java.lang.Object`
`clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait`

Methods inherited from interface `java.lang.Comparable`
`compareTo`

Methods inherited from interface `org.tanjakostic.jcleancim.model.UmlKind`
`getDesc`, `getLabel`, `getTag`, `getValue`

Fields

IF

`public static final org.tanjakostic.jcleancim.model.UmlClass.Iec61850Kind` **`IF`**

BASIC

`public static final org.tanjakostic.jcleancim.model.UmlClass.Iec61850Kind` **`BASIC`**

STRUCTURED

`public static final org.tanjakostic.jcleancim.model.UmlClass.Iec61850Kind` **`STRUCTURED`**

6.6.3.17 Originator (S_Originator unknown 61850)

PACKED_BASIC

`public static final org.tanjakostic.jcleancim.model.UmlClass.Iec61850Kind` **`PACKED_BASIC`**

ENUM

`public static final org.tanjakostic.jcleancim.model.UmlClass.Iec61850Kind` **`ENUM`**

PACKED_ENUM

`public static final org.tanjakostic.jcleancim.model.UmlClass.Iec61850Kind` **`PACKED_ENUM`**

ABBR_ENUM

`public static final org.tanjakostic.jcleancim.model.UmlClass.Iec61850Kind` **`ABBR_ENUM`**

COND_ENUM

public static final org.tanjakostic.jcleancim.model.UmlClass.Iec61850Kind **COND_ENUM**

PACKED_ENUM_DA

public static final org.tanjakostic.jcleancim.model.UmlClass.Iec61850Kind **PACKED_ENUM_DA**

ENUM_DA

public static final org.tanjakostic.jcleancim.model.UmlClass.Iec61850Kind **ENUM_DA**

PACKED_PRIM_DA

public static final org.tanjakostic.jcleancim.model.UmlClass.Iec61850Kind **PACKED_PRIM_DA**

PRIM_DA

public static final org.tanjakostic.jcleancim.model.UmlClass.Iec61850Kind **PRIM_DA**

COMP_DA

public static final org.tanjakostic.jcleancim.model.UmlClass.Iec61850Kind **COMP_DA**

CODED_ENUM_FCDA

public static final org.tanjakostic.jcleancim.model.UmlClass.Iec61850Kind **CODED_ENUM_FCDA**

ENUM_FCDA

public static final org.tanjakostic.jcleancim.model.UmlClass.Iec61850Kind **ENUM_FCDA**

PACKED_LIST_FCDA

public static final org.tanjakostic.jcleancim.model.UmlClass.Iec61850Kind **PACKED_LIST_FCDA**

COMP_FCDA

public static final org.tanjakostic.jcleancim.model.UmlClass.Iec61850Kind **COMP_FCDA**

FCDA

public static final org.tanjakostic.jcleancim.model.UmlClass.Iec61850Kind **FCDA**

CTS_CDC

public static final org.tanjakostic.jcleancim.model.UmlClass.Iec61850Kind **CTS_CDC**

ENUM_CDC

public static final org.tanjakostic.jcleancim.model.UmlClass.Iec61850Kind **ENUM_CDC**

TRANS_CDC

public static final org.tanjakostic.jcleancim.model.UmlClass.Iec61850Kind **TRANS_CDC**

PRIM_CDC

public static final org.tanjakostic.jcleancim.model.UmlClass.Iec61850Kind **PRIM_CDC**

COMP_CDC

public static final org.tanjakostic.jcleancim.model.UmlClass.Iec61850Kind **COMP_CDC**

LN

public static final org.tanjakostic.jcleancim.model.UmlClass.Iec61850Kind **LN**

FUNCTION

public static final org.tanjakostic.jcleancim.model.UmlClass.Iec61850Kind **FUNCTION**

OTHER_61850

public static final org.tanjakostic.jcleancim.model.UmlClass.Iec61850Kind **OTHER_61850**

(continued from last page)

NULL_61850

```
public static final org.tanjakostic.jcleancim.model.UmlClass.Iec61850Kind NULL_61850
```

UNKNOWN_61850

```
public static final org.tanjakostic.jcleancim.model.UmlClass.Iec61850Kind
UNKNOWN_61850
```

Methods

values

```
public static UmlClass.Iec61850Kind[] values()
```

valueOf

```
public static UmlClass.Iec61850Kind valueOf(java.lang.String name)
```

getUmlKinds

```
public static java.util.List getUmlKinds()
```

Returns all values as UmlKind list.

getValue

```
public java.lang.String getValue()
```

getLabel

```
public java.lang.String getLabel()
```

getTag

```
public java.lang.String getTag()
```

getDesc

```
public java.lang.String getDesc()
```

org.tanjakostic.jcleancim.model
Class UmlClass.InheritedKind

```
java.lang.Object
  +-java.lang.Enum
    +-org.tanjakostic.jcleancim.model.UmlClass.InheritedKind
```

All Implemented Interfaces:
java.io.Serializable, java.lang.Comparable

public static final class **UmlClass.InheritedKind**
extends java.lang.Enum

Used in queries for attributes, association ends and operations.

Field Summary

public static final	all
public static final	inherited
public static final	own

Method Summary

static UmlClass.InheritedKind	valueOf(java.lang.String name)
static UmlClass.InheritedKind[]	values()

Methods inherited from class `java.lang.Enum`

clone, compareTo, equals, finalize, getDeclaringClass, hashCode, name, ordinal, toString, valueOf

Methods inherited from class `java.lang.Object`

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface `java.lang.Comparable`

compareTo

Fields

(continued from last page)

own

```
public static final org.tanjakostic.jcleancim.model.UmlClass.InheritedKind own
```

inherited

```
public static final org.tanjakostic.jcleancim.model.UmlClass.InheritedKind inherited
```

all

```
public static final org.tanjakostic.jcleancim.model.UmlClass.InheritedKind all
```

Methods

values

```
public static UmlClass.InheritedKind[] values()
```

valueOf

```
public static UmlClass.InheritedKind valueOf(java.lang.String name)
```

org.tanjakostic.jcleancim.model
Class UmlClass.Data

```
java.lang.Object
  +-org.tanjakostic.jcleancim.model.UmlStructure.Data
    +-org.tanjakostic.jcleancim.model.UmlClass.Data
```

public static class **UmlClass.Data**
extends UmlStructure.Data

Data from the UML model repository specific to `UmlClass`.

Constructor Summary

public	`Data(boolean selfDependent, boolean isAbstract, boolean eaPersistentPropSet, boolean eaLeafPropSet, boolean eaRootPropSet, boolean isEaInterface, boolean associationClass, boolean selfInherited, boolean isEaEnumeration)` Constructor.
public	`Data(UmlStructure.Data data, boolean isAbstract, boolean eaPersistentPropSet, boolean eaLeafPropSet, boolean eaRootPropSet, boolean isEaInterface, boolean associationClass, boolean selfInherited, boolean isEaEnumeration)` Constructor.

Method Summary

static UmlClass.Data	`empty()` Returns an empty instance.
boolean	`isAbstract()`
boolean	`isAssociationClass()`
boolean	`isEaEnumeration()`
boolean	`isEaInterface()`
boolean	`isEaLeafPropSet()`
boolean	`isEaPersistentPropSet()`
boolean	`isEaRootPropSet()`
boolean	`isSelfInherited()`

Methods inherited from class `org.tanjakostic.jcleancim.model.UmlStructure.Data`

`empty`, `isSelfDependent`

Methods inherited from class `java.lang.Object`
`clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait`

Constructors

Data

```
public Data(boolean selfDependent,
            boolean isAbstract,
            boolean eaPersistentPropSet,
            boolean eaLeafPropSet,
            boolean eaRootPropSet,
            boolean isEaInterface,
            boolean associationClass,
            boolean selfInherited,
            boolean isEaEnumeration)
```

Constructor.

Parameters:

`selfDependent`
`isAbstract`
`eaPersistentPropSet`
`eaLeafPropSet`
`eaRootPropSet`
`isEaInterface`
`associationClass`
`selfInherited`
`isEaEnumeration`

Data

```
public Data(UmlStructure.Data data,
            boolean isAbstract,
            boolean eaPersistentPropSet,
            boolean eaLeafPropSet,
            boolean eaRootPropSet,
            boolean isEaInterface,
            boolean associationClass,
            boolean selfInherited,
            boolean isEaEnumeration)
```

Constructor.

Methods

empty

```
public static UmlClass.Data empty()
```

Returns an empty instance.

isAbstract

```
public boolean isAbstract()
```

isEaPersistentPropSet

public boolean **isEaPersistentPropSet**()

isEaLeafPropSet

public boolean **isEaLeafPropSet**()

isEaRootPropSet

public boolean **isEaRootPropSet**()

isEaInterface

public boolean **isEaInterface**()

isAssociationClass

public boolean **isAssociationClass**()

isSelfInherited

public boolean **isSelfInherited**()

isEaEnumeration

public boolean **isEaEnumeration**()

org.tanjakostic.jcleancim.model
Class UmlConstraint

```
java.lang.Object
  |-org.tanjakostic.jcleancim.model.AbstractUmlObject
    |-org.tanjakostic.jcleancim.model.UmlConstraint
```

All Implemented Interfaces:
UmlObject

---

public class **UmlConstraint**
extends AbstractUmlObject

UML constraint.

Initially, we have designed it to allow handling of complex IEC61850 presence conditions (through class constraints; for recognised formats, see `UmlConstraint.Kind`) and IEC61850 array min/max index (through attribute constraint).

Then we retrofitted the implementation to be able to use "vanilla" constraints on classes (without any special processing). For CIM domains, just ignore specials and use normal UmlObject methods on this class that you normally use for other UmlObject-s.

Design note: We could have had two subclasses, but it would have been an overkill at this point in time.

Nested Class Summary

class	`UmlConstraint.Data` UmlConstraint.Data
class	`UmlConstraint.Kind` UmlConstraint.Kind

Field Summary

public static final	`SEPARATOR` (61850) Used to separate attribute names from the condition text for class constraints. Value: **:**

Fields inherited from class `org.tanjakostic.jcleancim.model.AbstractUmlObject`
`CLASS_SEPARATOR`, `NULL_OBJ_NAME`, `PACKAGE_SEPARATOR`

Method Summary

java.util.List	`getAttrNames()` Returns list of class attribute names if this is a class constraint, empty list otherwise.
java.lang.String	`getCondition()` Returns value for condition if this is an attribute constraint, or description of presence condition if this is a class constraint.
`UmlAttribute`	`getContainingAttribute()` Returns containing attribute if this is an attribute constraint, null otherwise.

UmlClass	getContainingClass() Returns containing class if this is a class constraint, null otherwise.
UmlKind	getKind()
Namespace	getNamespace()
Nature	getNature()
OwningWg	getOwner()
java.util.Set	getPredefinedTagNames()
PresenceCondition	getPresenceCondition() Returns presence condition deduced from this IEC61850 class constraint; null for other model nature classes and for attribute constraint.
java.lang.String	getQualifiedName()
boolean	isInformative() This default implementation returns true if objects steretypes include UmlStereotype#INFORMATIVE. If there are additional criteria for deriving informative status, ensure to invoke this default implementation first.
boolean	isSupportsTags() Returns whether tagged values are supported.
java.lang.String	toString()
void	validateTag(java.lang.String name, java.lang.String value)

Methods inherited from class org.tanjakostic.jcleancim.model.AbstractUmlObject

addDeprecAndInf, addTaggedValue, appendRemainingCustomStereotypes, classifyPerScope, classifyPerScopePerTag, classifyPerTag, collectDuplicateDescriptions, collectDuplicateNames, collectForScope, collectNames, collectQNames, findAllForName, findWithSameUuidAndLog, getAlias, getDescription, getHtmlDescription, getId, getKind, getName, getNature, getOwner, getQualifiedName, getSince, getStereotype, getTaggedValues, getUnallowedTagNames, getUuid, getVisibility, initFromTags, isDeprecated, isInformative, saveTags, toShortString, toShortString, validateTag

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.model.UmlObject

addTaggedValue, getAlias, getDescription, getHtmlDescription, getId, getKind, getName, getNamespace, getNature, getOwner, getPredefinedTagNames, getQualifiedName, getSince, getStereotype, getTaggedValues, getUnallowedTagNames, getUuid, getVisibility, isDeprecated, isInformative, toShortString

(continued from last page)

Fields

SEPARATOR

`public static final java.lang.String` **`SEPARATOR`**

(61850) Used to separate attribute names from the condition text for class constraints.
Constant value: **`:`**

Methods

getContainingClass

`public` `UmlClass` **`getContainingClass`**`()`

Returns containing class if this is a class constraint, null otherwise.

getAttrNames

`public java.util.List` **`getAttrNames`**`()`

Returns list of class attribute names if this is a class constraint, empty list otherwise.

getPresenceCondition

`public` `PresenceCondition` **`getPresenceCondition`**`()`

Returns presence condition deduced from this IEC61850 class constraint; null for other model nature classes and for attribute constraint.

getCondition

`public java.lang.String` **`getCondition`**`()`

Returns value for condition if this is an attribute constraint, or description of presence condition if this is a class constraint.

getContainingAttribute

`public` `UmlAttribute` **`getContainingAttribute`**`()`

Returns containing attribute if this is an attribute constraint, null otherwise.

isSupportsTags

`public boolean` **`isSupportsTags`**`()`

Returns whether tagged values are supported.

getOwner

`public` `OwningWg` **`getOwner`**`()`

getNamespace

`public` `Namespace` **`getNamespace`**`()`

(continued from last page)

getNature

```
public Nature getNature()
```

isInformative

```
public boolean isInformative()
```

This default implementation returns true if objects steretypes include UmlStereotype#INFORMATIVE. If there are additional criteria for deriving informative status, ensure to invoke this default implementation first.

Returns whether its container is informative, ignoring any potential stereotype (these cannot be stored in the model).

getKind

```
public UmlKind getKind()
```

getQualifiedName

```
public java.lang.String getQualifiedName()
```

validateTag

```
protected void validateTag(java.lang.String name,
         java.lang.String value)
```

Subclasses should overwride this method in case some validation about the tagged value is needed before adding it. This default implementation is a no-op.

getPredefinedTagNames

```
public java.util.Set getPredefinedTagNames()
```

Always returns empty set.

toString

```
public java.lang.String toString()
```

org.tanjakostic.jcleancim.model
Class UmlConstraint.Kind

```
java.lang.Object
  |-java.lang.Enum
    |-org.tanjakostic.jcleancim.model.UmlConstraint.Kind
```

All Implemented Interfaces:
UmlKind, java.io.Serializable, java.lang.Comparable

---

public static final class **UmlConstraint.Kind**
extends java.lang.Enum
implements java.lang.Comparable, java.io.Serializable, UmlKind

Kind of constraint.

Field Summary

public static final	ATTR_MIN_MAX We currently use attribute constraints in IEC61850-7-3 for attributes that are arrays, to store their min and max index, to be able to print "ARRAY min...max OF XYZ".
public static final	CLASS We currently use class constraints in IEC61850-7-4 and IEC61850-7-3 UML for presence conditions of attributes that are not simply M or O.

Method Summary

java.lang.String	getDesc()
java.lang.String	getLabel()
java.lang.String	getTag()
java.lang.String	getValue()
static UmlConstraint.Kind	valueOf(java.lang.String name)
static UmlConstraint.Kind[]	values()

Methods inherited from class `java.lang.Enum`

clone, compareTo, equals, finalize, getDeclaringClass, hashCode, name, ordinal, toString, valueOf

Methods inherited from class `java.lang.Object`

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface `java.lang.Comparable`
`compareTo`

Methods inherited from interface `org.tanjakostic.jcleancim.model.UmlKind`
`getDesc`, `getLabel`, `getTag`, `getValue`

Fields

CLASS

`public static final org.tanjakostic.jcleancim.model.UmlConstraint.Kind` **`CLASS`**

We currently use class constraints in IEC61850-7-4 and IEC61850-7-3 UML for presence conditions of attributes that are not simply M or O. These conditions usually involve multiple attributes of a class, so we place constraint on a class. The expected format for doc of the constraint is:

```
<comma>[: optional free text]
```

ATTR_MIN_MAX

`public static final org.tanjakostic.jcleancim.model.UmlConstraint.Kind` **`ATTR_MIN_MAX`**

We currently use attribute constraints in IEC61850-7-3 for attributes that are arrays, to store their min and max index, to be able to print "ARRAY min...max OF XYZ".

Methods

values

`public static UmlConstraint.Kind[]` **`values`**`()`

valueOf

`public static UmlConstraint.Kind` **`valueOf`**`(java.lang.String name)`

getValue

`public java.lang.String` **`getValue`**`()`

getLabel

`public java.lang.String` **`getLabel`**`()`

(continued from last page)

getTag

```
public java.lang.String getTag()
```

getDesc

```
public java.lang.String getDesc()
```

org.tanjakostic.jcleancim.model
Class UmlConstraint.Data

```
java.lang.Object
   |
   +-org.tanjakostic.jcleancim.model.UmlConstraint.Data
```

public static class **UmlConstraint.Data**
extends java.lang.Object

Data from the UML model repository specific to `UmlConstraint`.

Constructor Summary

public	`Data(java.util.List attrNames, java.lang.String condition, boolean supportsTags)` Constructor.

Method Summary

`java.util.List`	`getAttrNames()`
`java.lang.String`	`getCondition()`
`boolean`	`isSupportsTags()`

Methods inherited from class `java.lang.Object`
`clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait`

Constructors

Data

```
public Data(java.util.List attrNames,
            java.lang.String condition,
            boolean supportsTags)
```

Constructor.

Parameters:
`attrNames`
`condition` - text in the notes
`supportsTags`

Methods

getAttrNames

```
public java.util.List getAttrNames()
```

(continued from last page)

getCondition

```
public java.lang.String getCondition()
```

isSupportsTags

```
public boolean isSupportsTags()
```

org.tanjakostic.jcleancim.model
Class UmlDependency

```
java.lang.Object
  +-org.tanjakostic.jcleancim.model.AbstractUmlObject
    +-org.tanjakostic.jcleancim.model.UmlDependency
```

All Implemented Interfaces:
UmlObject

---

public class **UmlDependency**
extends AbstractUmlObject

Explicit (hand-drawn) UML dependency between either two structures (packages or classes).

Design note: We could have had two subclasses, but it would have been an overkill at this point in time.

Nested Class Summary

class	UmlDependency.Data UmlDependency.Data
class	UmlDependency.Kind UmlDependency.Kind

Fields inherited from class org.tanjakostic.jcleancim.model.AbstractUmlObject
CLASS_SEPARATOR, NULL_OBJ_NAME, PACKAGE_SEPARATOR

Method Summary

UmlKind	getKind()
static java.util.List	getKinds(Nature nature) Returns all available classifications (kinds) for dependencies.
Namespace	getNamespace()
Nature	getNature()
OwningWg	getOwner()
java.util.Set	getPredefinedTagNames()
java.lang.String	getQualifiedName()
UmlStructure	getSource() Returns source.
UmlStructure	getTarget() Returns target.

boolean	isForPackage() Returns true if this is dependency between packages, false if it is between classes.
boolean	isInformative() This default implementation returns true if objects steretypes include UmlStereotype#INFORMATIVE. If there are additional criteria for deriving informative status, ensure to invoke this default implementation first.
boolean	isWithinSameWg() Returns whether source and target have the same owner.
java.lang.String	toString()

Methods inherited from class org.tanjakostic.jcleancim.model.AbstractUmlObject

addDeprecAndInf, addTaggedValue, appendRemainingCustomStereotypes, classifyPerScope, classifyPerScopePerTag, classifyPerTag, collectDuplicateDescriptions, collectDuplicateNames, collectForScope, collectNames, collectQNames, findAllForName, findWithSameUuidAndLog, getAlias, getDescription, getHtmlDescription, getId, getKind, getName, getNature, getOwner, getQualifiedName, getSince, getStereotype, getTaggedValues, getUnallowedTagNames, getUuid, getVisibility, initFromTags, isDeprecated, isInformative, saveTags, toShortString, toShortString, validateTag

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.model.UmlObject

addTaggedValue, getAlias, getDescription, getHtmlDescription, getId, getKind, getName, getNamespace, getNature, getOwner, getPredefinedTagNames, getQualifiedName, getSince, getStereotype, getTaggedValues, getUnallowedTagNames, getUuid, getVisibility, isDeprecated, isInformative, toShortString

Methods

getKinds

public static java.util.List **getKinds**(Nature nature)

Returns all available classifications (kinds) for dependencies.

Parameters:
nature - ignored in this method

getSource

public UmlStructure **getSource**()

Returns source.

getTarget

public UmlStructure **getTarget**()

(continued from last page)

Returns target.

isForPackage

`public boolean isForPackage()`

Returns true if this is dependency between packages, false if it is between classes.

isWithinSameWg

`public boolean isWithinSameWg()`

Returns whether source and target have the same owner.

getOwner

`public OwningWg getOwner()`

getNamespace

`public Namespace getNamespace()`

getNature

`public Nature getNature()`

isInformative

`public boolean isInformative()`

This default implementation returns true if objects steretypes include UmlStereotype#INFORMATIVE. If there are additional criteria for deriving informative status, ensure to invoke this default implementation first.

Dependency is never informative, and this method always returns false.

getKind

`public UmlKind getKind()`

getQualifiedName

`public java.lang.String getQualifiedName()`

getPredefinedTagNames

`public java.util.Set getPredefinedTagNames()`

(continued from last page)

Always returns empty set.

toString

```
public java.lang.String toString()
```

org.tanjakostic.jcleancim.model
Class UmlDependency.Kind

```
java.lang.Object
  |-java.lang.Enum
    |-org.tanjakostic.jcleancim.model.UmlDependency.Kind
```

All Implemented Interfaces:
UmlKind, java.io.Serializable, java.lang.Comparable

---

public static final class **UmlDependency.Kind**
extends java.lang.Enum
implements java.lang.Comparable, java.io.Serializable, UmlKind

Kinds of dependencies.

Field Summary

public static final	CLASS
public static final	PACKAGE

Method Summary

java.lang.String	getDesc()
java.lang.String	getLabel()
java.lang.String	getTag()
java.lang.String	getValue()
static UmlDependency.Kind	valueOf(java.lang.String name)
static UmlDependency.Kind[]	values()

Methods inherited from class `java.lang.Enum`

clone, compareTo, equals, finalize, getDeclaringClass, hashCode, name, ordinal, toString, valueOf

Methods inherited from class `java.lang.Object`

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface `java.lang.Comparable`

compareTo

Methods inherited from interface org.tanjakostic.jcleancim.model.UmlKind
getDesc, getLabel, getTag, getValue

Fields

PACKAGE

```
public static final org.tanjakostic.jcleancim.model.UmlDependency.Kind PACKAGE
```

CLASS

```
public static final org.tanjakostic.jcleancim.model.UmlDependency.Kind CLASS
```

Methods

values

```
public static UmlDependency.Kind[] values()
```

valueOf

```
public static UmlDependency.Kind valueOf(java.lang.String name)
```

getValue

```
public java.lang.String getValue()
```

getLabel

```
public java.lang.String getLabel()
```

getTag

```
public java.lang.String getTag()
```

getDesc

```
public java.lang.String getDesc()
```

org.tanjakostic.jcleancim.model
Class UmlDependency.Data

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.model.UmlDependency.Data
```

public static class **UmlDependency.Data**
extends java.lang.Object

Data from the UML model repository specific to `UmlDependency`.

Constructor Summary

public	`Data()` Constructor.

Method Summary

static `UmlDependency.Data`	`empty()` Returns an empty instance.

Methods inherited from class `java.lang.Object`
`clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait`

Constructors

Data

`public` **`Data`**`()`

Constructor.

Methods

empty

`public static UmlDependency.Data` **`empty`**`()`

Returns an empty instance.

org.tanjakostic.jcleancim.model
Class UmlDiagram

```
java.lang.Object
  |-org.tanjakostic.jcleancim.model.AbstractUmlObject
    |-org.tanjakostic.jcleancim.model.UmlDiagram
```

All Implemented Interfaces:
UmlObject

---

public class **UmlDiagram**
extends AbstractUmlObject

Diagram from the UML model, assigned to either a class or a package.

Design note: We could have had two subclasses, but it would have been an overkill at this point in time.

Nested Class Summary

class	UmlDiagram.Data UmlDiagram.Data
class	UmlDiagram.Kind UmlDiagram.Kind

Fields inherited from class org.tanjakostic.jcleancim.model.AbstractUmlObject
CLASS_SEPARATOR, NULL_OBJ_NAME, PACKAGE_SEPARATOR

Method Summary

UmlStructure	getContainer() Returns the containing structure (class or package).
UmlKind	getKind()
static java.util.List	getKinds(Nature nature) Returns all available classifications (kinds) for diagrams.
Namespace	getNamespace()
Nature	getNature()
OwningWg	getOwner()
java.io.File	getPic() Returns the file where the image has been stored; in case the application failed or did not need to store the real image, the file is the default and isBlankPic() returns true.
java.util.Set	getPredefinedTagNames()

java.lang.String	getQualifiedName()
boolean	isBlankPic() Returns true in case the application failed or did not need to store the real image.
boolean	isForPackage() Returns whether the containing structure for this diagram is a package.
boolean	isInformative()
boolean	isPortrait() Returns whether the page format is portrait.
boolean	isSupportsTags() Returns whether tagged values are allowed.
java.lang.String	toString()
void	validateTag(java.lang.String name, java.lang.String value)

Methods inherited from class org.tanjakostic.jcleancim.model.AbstractUmlObject

addDeprecAndInf, addTaggedValue, appendRemainingCustomStereotypes, classifyPerScope, classifyPerScopePerTag, classifyPerTag, collectDuplicateDescriptions, collectDuplicateNames, collectForScope, collectNames, collectQNames, findAllForName, findWithSameUuidAndLog, getAlias, getDescription, getHtmlDescription, getId, getKind, getName, getNature, getOwner, getQualifiedName, getSince, getStereotype, getTaggedValues, getUnallowedTagNames, getUuid, getVisibility, initFromTags, isDeprecated, isInformative, saveTags, toShortString, toShortString, validateTag

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.model.UmlObject

addTaggedValue, getAlias, getDescription, getHtmlDescription, getId, getKind, getName, getNamespace, getNature, getOwner, getPredefinedTagNames, getQualifiedName, getSince, getStereotype, getTaggedValues, getUnallowedTagNames, getUuid, getVisibility, isDeprecated, isInformative, toShortString

Methods

getKinds

public static java.util.List **getKinds**(Nature nature)

Returns all available classifications (kinds) for diagrams.

Parameters:
nature - ignored in this method

(continued from last page)

getContainer

`public UmlStructure getContainer()`

Returns the containing structure (class or package).

isForPackage

`public boolean isForPackage()`

Returns whether the containing structure for this diagram is a package.

getPic

`public java.io.File getPic()`

Returns the file where the image has been stored; in case the application failed or did not need to store the real image, the file is the default and `isBlankPic()` returns true.

isBlankPic

`public boolean isBlankPic()`

Returns true in case the application failed or did not need to store the real image. We allow for this condition in order to be able to run the rest of application, even without the diagrams.

isPortrait

`public boolean isPortrait()`

Returns whether the page format is portrait.

isSupportsTags

`public boolean isSupportsTags()`

Returns whether tagged values are allowed.

getOwner

`public OwningWg getOwner()`

getNamespace

`public Namespace getNamespace()`

isInformative

`public boolean isInformative()`

This default implementation returns true if objects steretypes include UmlStereotype#INFORMATIVE. If there are additional criteria for deriving informative status, ensure to invoke this default implementation first.

getNature

```
public Nature getNature()
```

getKind

```
public UmlKind getKind()
```

getQualifiedName

```
public java.lang.String getQualifiedName()
```

validateTag

```
protected void validateTag(java.lang.String name,
         java.lang.String value)
```

Subclasses should overwride this method in case some validation about the tagged value is needed before adding it. This default implementation is a no-op.

getPredefinedTagNames

```
public java.util.Set getPredefinedTagNames()
```

Always returns empty set.

toString

```
public java.lang.String toString()
```

org.tanjakostic.jcleancim.model
Class UmlDiagram.Kind

```
java.lang.Object
  +-java.lang.Enum
    +-org.tanjakostic.jcleancim.model.UmlDiagram.Kind
```

All Implemented Interfaces:
UmlKind, java.io.Serializable, java.lang.Comparable

---

public static final class **UmlDiagram.Kind**
extends java.lang.Enum
implements java.lang.Comparable, java.io.Serializable, UmlKind

Kind of diagram, as given by EA.

Field Summary

public static final	ACTIVITY
public static final	ANALYSIS
public static final	COMPONENT
public static final	CUSTOM
public static final	DEPLOYMENT
public static final	LOGICAL
public static final	OBJECT
public static final	OTHER
public static final	PACKAGE
public static final	SEQUENCE
public static final	STATECHART
public static final	USE_CASE

Method Summary

static UmlDiagram.Kind	findForValue(java.lang.String value) Returns literal with value if found, OTHER instance otherwise.

java.lang.String	getDesc()
java.lang.String	getLabel()
java.lang.String	getTag()
java.lang.String	getValue()
static UmlDiagram.Kind	valueOf(java.lang.String name)
static UmlDiagram.Kind[]	values()

Methods inherited from class java.lang.Enum
clone, compareTo, equals, finalize, getDeclaringClass, hashCode, name, ordinal, toString, valueOf

Methods inherited from class java.lang.Object
clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface java.lang.Comparable
compareTo

Methods inherited from interface org.tanjakostic.jcleancim.model.UmlKind
getDesc, getLabel, getTag, getValue

Fields

ACTIVITY

public static final org.tanjakostic.jcleancim.model.UmlDiagram.Kind **ACTIVITY**

ANALYSIS

public static final org.tanjakostic.jcleancim.model.UmlDiagram.Kind **ANALYSIS**

COMPONENT

public static final org.tanjakostic.jcleancim.model.UmlDiagram.Kind **COMPONENT**

(continued from last page)

CUSTOM

`public static final org.tanjakostic.jcleancim.model.UmlDiagram.Kind` **`CUSTOM`**

DEPLOYMENT

`public static final org.tanjakostic.jcleancim.model.UmlDiagram.Kind` **`DEPLOYMENT`**

LOGICAL

`public static final org.tanjakostic.jcleancim.model.UmlDiagram.Kind` **`LOGICAL`**

SEQUENCE

`public static final org.tanjakostic.jcleancim.model.UmlDiagram.Kind` **`SEQUENCE`**

STATECHART

`public static final org.tanjakostic.jcleancim.model.UmlDiagram.Kind` **`STATECHART`**

USE_CASE

`public static final org.tanjakostic.jcleancim.model.UmlDiagram.Kind` **`USE_CASE`**

PACKAGE

`public static final org.tanjakostic.jcleancim.model.UmlDiagram.Kind` **`PACKAGE`**

OBJECT

`public static final org.tanjakostic.jcleancim.model.UmlDiagram.Kind` **`OBJECT`**

OTHER

`public static final org.tanjakostic.jcleancim.model.UmlDiagram.Kind` **`OTHER`**

Methods

(continued from last page)

values

```
public static UmlDiagram.Kind[] values()
```

valueOf

```
public static UmlDiagram.Kind valueOf(java.lang.String name)
```

findForValue

```
public static UmlDiagram.Kind findForValue(java.lang.String value)
```

Returns literal with `value` if found, `OTHER` instance otherwise.

getValue

```
public java.lang.String getValue()
```

getLabel

```
public java.lang.String getLabel()
```

getTag

```
public java.lang.String getTag()
```

getDesc

```
public java.lang.String getDesc()
```

org.tanjakostic.jcleancim.model
Class UmlDiagram.Data

```
java.lang.Object
   |
   +-org.tanjakostic.jcleancim.model.UmlDiagram.Data
```

public static class **UmlDiagram.Data**
extends java.lang.Object

Data from the UML model repository specific to `UmlDiagram`.

Constructor Summary

public	`Data(UmlDiagram.Kind kind, boolean portrait, boolean supportsTags)` Constructor.

Method Summary

static `UmlDiagram.Data`	`empty()` Returns an empty instance; sets default kind to `UmlDiagram.Kind.CUSTOM`.
`UmlDiagram.Kind`	`getKind()`
boolean	`isPortrait()`
boolean	`isSupportsTags()`

Methods inherited from class `java.lang.Object`
`clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait`

Constructors

Data

```
public Data(UmlDiagram.Kind kind,
            boolean portrait,
            boolean supportsTags)
```

Constructor.

Parameters:
`kind`
`portrait`
`supportsTags`

Methods

(continued from last page)

empty

```
public static UmlDiagram.Data empty()
```

Returns an empty instance; sets default kind to `UmlDiagram.Kind.CUSTOM`.

getKind

```
public UmlDiagram.Kind getKind()
```

isPortrait

```
public boolean isPortrait()
```

isSupportsTags

```
public boolean isSupportsTags()
```

org.tanjakostic.jcleancim.model
Interface UmlKind

All Known Implementing Classes:
Kind, Kind, Iec61850Kind, CimKind, Kind, Kind, Kind, ReturnKind, Kind, Kind, Kind

public interface **UmlKind**
extends

Interface intended to be implemented by various *Kind enumerations, to allow for uniform processing for any `UmlObject`.

In some instances, we have to store predefined names (like EA stereotypes), so we define custom constructors that take that value for an additional field. In other instances, we cannot define an enumeration literal with the name of java reserved words (like interface, package or class), but would like to print things a bit differently. For some UML objects, we also need to print machine-processable tokens as XML tags. The methods below allow for customised strings per enumeration literal for different purposes.

Use the toString() method to obtain the name of enumeration literal itself.

Method Summary

abstract java.lang.String	`getDesc()` Returns the description, typically used for statistics printing.
abstract java.lang.String	`getLabel()` Returns the label to be used for human-readable documentation.
abstract java.lang.String	`getTag()` Returns the tag to be used for machine-processable documentation, typically used for statistics printing.
abstract java.lang.String	`getValue()` Returns the value set by the custom constructor.

Methods

getValue

`public abstract java.lang.String getValue()`

Returns the value set by the custom constructor. E.g., for ENUM("enumaration", "enumaration" "Enumeration"), returns first argument "enumaration".

getLabel

`public abstract java.lang.String getLabel()`

Returns the label to be used for human-readable documentation. E.g., ENUM("enumaration", "enumaration" "Enumeration"), returns second argument "enumaration".

getTag

`public abstract java.lang.String getTag()`

(continued from last page)

Returns the tag to be used for machine-processable documentation, typically used for statistics printing. E.g., ENUM("enumaration", "enumaration" "Enumeration") returns third argument "Enumeration".

getDesc

```
public abstract java.lang.String getDesc()
```

Returns the description, typically used for statistics printing. PRIM("Primitive", "primitive" "primitive class"), returns "primitive class".

org.tanjakostic.jcleancim.model
Class UmlModel

```
java.lang.Object
   |
   +-org.tanjakostic.jcleancim.model.UmlModel
```

public class **UmlModel**
extends java.lang.Object

TODO: Add link to test model built with the API.

Under the root in a repository, there may be several models. This class stores them all as `UmlPackage.Kind.MODEL` packages - they are the entry point into the full UML content. In the standard IEC TC57 UML we have three such models:

- TC57CIM, with `Nature.CIM` (default),
- TC57CIMProfiles, with `CIM_PROFILE` (must be specified in configuration), and,
- IEC61850Domain, with `Nature.IEC61850` (must be specified in configuration).

A model package contains one or more `UmlPackage.Kind.TOP` packages, each of them assigned to an owner `OwningWg`. Owner is typically an IEC TC57 working group that manages the part of the model under the `UmlPackage.Kind.TOP` package.

If a model package is found that contains top-level packages of unknown names, that model will get the owner `OwningWg.OTHER_CIM` (default) or `OwningWg.OTHER_IEC61850` (if model name specified in configuration as `Nature.IEC61850`). This allows to combine the custom extensions with the standard models.

The current implementation of `OwningWg` defines the "known" top-package names and IEC TC57 working groups and holds the rules about allowed dependencies. That class does not care about models, but rather about top-level packages (each owned by a WG). This also provides for flexibility when you develop non-standard extensions.

Implementation note 1: The internal maps in this class are used purely to store model elements present in the UML repository, for quick access to elements per UML type where bulk data is needed. The package-private modifiers, such as `addClass(UmlClass)` , perform no consistency checks at all and should not be used - respect the restrictions given in their doc! In contrast, the in-memory objects, once after they get inter-linked with their own accessors (e.g., `UmlPackage.addClass(UmlClass)`), can be navigated "naturally" with their own getters ((e.g., `UmlPackage.getClasses()`)). The entry point for that navigation are model packages, obtained through `getModelPackages()`.

Implementation note 2: This class also creates internal "null" model packages and classes (one per nature) to provide for valid in-memory objects for types of attributes and operations that may be invalid in the repository (and for which we don't have any information, but need to create an instance of `UmlClass`. The treatment of these "null" elements is different on purpose: we don't want to, e.g., validate them or to print their documentation - but we have to be able to do this with the in-memory objects whose type `UmlClass` may be invalid in the original UML repository - so, they are never included in maps/collections that contain the in-memory contect from the real UML model repository.

Constructor Summary

public	`UmlModel(Config cfg)` Constructor.

Method Summary

void	`crossCheck(UmlModel profilesModel)`
java.util.Collection	`findAssociations(java.util.EnumSet wgs, java.util.EnumSet kinds, boolean includeNormative, boolean includeInformative)` Returns all associations involving classes from owners `wg`.

java.util.List	findAttributes(java.lang.String packageName, boolean includeLiterals, boolean includeNonLiterals) Returns all attributes in and under the package packageName according to the given filters (note: setting both includeLiterals and includeNonLiterals to true returns all attributes).
java.util.List	findAttributes(java.lang.String packageName, boolean includeLiterals, boolean includeNonLiterals, boolean namesOrdered) Same as findAttributes(String, boolean, boolean), but allows for ordering per name to be specified.
java.util.Collection	findAttributesWithConstraints() Returns attributes that have any kind of constraint (own and by class).
java.util.Map	findAttributesWithDuplicates(java.lang.String packageName, boolean includeLiterals, boolean includeNonLiterals, boolean namesOrdered) Returns all attributes per name in and under the package packageName, grouped by their name and according to the given filters (note: setting both includeLiterals and includeNonLiterals to true returns all attributes).
java.util.Collection	findCimNoncimAssociations() Returns all associations that are mappings between models of different natures.
java.util.Set	findClasses(java.util.EnumSet wgs, java.util.EnumSet cimKinds, java.util.EnumSet iec61850Kinds, boolean includeNormative, boolean includeInformative) Returns all classes from owners wg.
java.util.Set	findClasses(java.lang.String name) Returns all classes with name matching name.
java.util.Collection	findClassesWithConstraints() Returns all classes that have constraints.
java.util.Collection	findDiagrams(java.lang.String containerName, java.lang.String name, boolean includeOnPackage, boolean includeOnClass) Returns all diagrams whose container name matches containerName and with name name, with the specified options applied.
java.util.Collection	findDOAttributes() Returns attributes on logical nodes (not in meta-model).
java.util.Collection	findMultivaluedAttributes() Returns multi-valued attributes.
java.util.Collection	findPackages(java.util.List names) TODO: Refactor to use AbstractUmlObject.findAllForName().
java.util.Map	findPresenceConditionLiterals() (IEC61850) Returns all literals that represent presence conditions.
java.util.Map	getAbbreviatedTermsSortedPerDecreasingLength() (IEC61850) Returns all abbreviated terms sorted by decreasing length; handles duplicate definitions by appending all of them per term.
java.util.Map	getAbbreviationLiterals() (IEC61850) Returns all abbreviation literal indexed by name.
java.util.Collection	getAssociations() Returns all associations in this model.

java.util.Collection	getAttributes() Returns all attributes in this model.
Config	getCfg() Returns configuration.
java.util.Collection	getClasses() Returns all classes in this model.
java.util.Collection	getDependencies() Returns all explicit (hand-drawn) dependencies in this model.
java.util.Collection	getDiagrams() Returns all diagrams in this model.
java.lang.String	getModelNamesWithNature() Returns comma-separated pairs {modelPackageName nature}.
java.util.Collection	getModelPackages() Returns all model packages.
java.util.Collection	getModelPackages(java.util.EnumSet natures) Returns model packages of specified nature.
java.util.Collection	getNamespaceInfos() Returns non-null namespace informations for all packages.
java.util.Collection	getNamespacePackages(java.util.EnumSet wgs) Returns packages that have namespace info for specified owners.
java.util.Map	getNullClasses() Returns special, "null" classes per nature.
java.util.Map	getNullModelPackages() Returns special, "null" model packages per nature; they are created by default to hold special, "null" classes that may be needed when the model repository allows for invalid or bad definition of types for attributes and operation parameters.
java.util.Collection	getOperations() Returns all operations in this model.
java.util.Collection	getPackages() Returns all packages in this model.
java.util.Map	getTags()
java.util.Collection	getTopPackages(java.util.EnumSet wgs) Returns top packages for specified owners.
java.lang.String	getUuid() Returns model UUID.
java.util.Collection	getVersionInfos(java.util.EnumSet wgs) Returns version informations for top-level packages of specified owners.
java.lang.String	toString()

Methods inherited from class `java.lang.Object`

```
clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait,
wait
```

Constructors

UmlModel

`public UmlModel(Config cfg)`

Constructor.

Methods

getCfg

`public Config getCfg()`

Returns configuration.

getUuid

`public java.lang.String getUuid()`

Returns model UUID.

getModelPackages

`public java.util.Collection getModelPackages()`

Returns all model packages.

getNullModelPackages

`public java.util.Map getNullModelPackages()`

Returns special, "null" model packages per nature; they are created by default to hold special, "null" classes that may be needed when the model repository allows for invalid or bad definition of types for attributes and operation parameters. These "null" model packages are not included in the model through regular packages accessors, but through this method only.

getNullClasses

`public java.util.Map getNullClasses()`

Returns special, "null" classes per nature.

See Also:
`getNullModelPackages()`

crossCheck

`public void crossCheck(UmlModel profilesModel)`

(continued from last page)

Parameters:
`profilesModel`

getPackages

`public java.util.Collection` **`getPackages`**`()`

Returns all packages in this model.

findPackages

`public java.util.Collection` **`findPackages`**`(java.util.List names)`

TODO: Refactor to use AbstractUmlObject.findAllForName().

Returns potentially empty list of all packages whose name matches one of `names`.

getClasses

`public java.util.Collection` **`getClasses`**`()`

Returns all classes in this model.

findClasses

`public java.util.Set` **`findClasses`**`(java.lang.String name)`

Returns all classes with name matching `name`.

findClasses

```
public java.util.Set findClasses(java.util.EnumSet wgs,
         java.util.EnumSet cimKinds,
         java.util.EnumSet iec61850Kinds,
         boolean includeNormative,
         boolean includeInformative)
```

Returns all classes from owners `wg`.

Parameters:
`wgs` - one or more owners.
`cimKinds` - one or more CIM class kinds.
`iec61850Kinds` - one or more IEC61850 class kinds.
`includeNormative` - whether to include normative classes.
`includeInformative` - whether to include informative classes.

findClassesWithConstraints

`public java.util.Collection` **`findClassesWithConstraints`**`()`

Returns all classes that have constraints.

getAbbreviatedTermsSortedPerDecreasingLength

`public java.util.Map` **`getAbbreviatedTermsSortedPerDecreasingLength`**`()`

(IEC61850) Returns all abbreviated terms sorted by decreasing length; handles duplicate definitions by appending all of them per term.

getAttributes

```
public java.util.Collection getAttributes()
```

Returns all attributes in this model.

getAbbreviationLiterals

```
public java.util.Map getAbbreviationLiterals()
```

(IEC61850) Returns all abbreviation literal indexed by name.

findPresenceConditionLiterals

```
public java.util.Map findPresenceConditionLiterals()
```

(IEC61850) Returns all literals that represent presence conditions.

findAttributes

```
public java.util.List findAttributes(java.lang.String packageName,
         boolean includeLiterals,
         boolean includeNonLiterals)
```

Returns all attributes in and under the package `packageName` according to the given filters (note: setting both `includeLiterals` and `includeNonLiterals` to true returns all attributes).

Parameters:

`packageName` - name of package to start from.
`includeLiterals` - whether to include enumeration literals.
`includeNonLiterals` - whether to include non-literals.

findAttributesWithDuplicates

```
public java.util.Map findAttributesWithDuplicates(java.lang.String packageName,
         boolean includeLiterals,
         boolean includeNonLiterals,
         boolean namesOrdered)
```

Returns all attributes per name in and under the package `packageName`, grouped by their name and according to the given filters (note: setting both `includeLiterals` and `includeNonLiterals` to true returns all attributes). The names may be ordered.

Parameters:

`packageName` - name of package to start from.
`includeLiterals` - whether to include enumeration literals.
`includeNonLiterals` - whether to include non-literals.
`namesOrdered` - whether to perform name ordering.

findAttributes

```
public java.util.List findAttributes(java.lang.String packageName,
         boolean includeLiterals,
         boolean includeNonLiterals,
         boolean namesOrdered)
```

Same as `findAttributes(String, boolean, boolean)`, but allows for ordering per name to be specified.

Parameters:

(continued from last page)

`packageName` - name of package to start from.
`includeLiterals` - whether to include enumeration literals.
`includeNonLiterals` - whether to include non-literals.
`namesOrdered` - whether to perform name ordering.

findAttributesWithConstraints

```
public java.util.Collection findAttributesWithConstraints()
```

Returns attributes that have any kind of constraint (own and by class).

findMultivaluedAttributes

```
public java.util.Collection findMultivaluedAttributes()
```

Returns multi-valued attributes.

findDOAttributes

```
public java.util.Collection findDOAttributes()
```

Returns attributes on logical nodes (not in meta-model).

getOperations

```
public java.util.Collection getOperations()
```

Returns all operations in this model.

getAssociations

```
public java.util.Collection getAssociations()
```

Returns all associations in this model.

findAssociations

```
public java.util.Collection findAssociations(java.util.EnumSet wgs,
         java.util.EnumSet kinds,
         boolean includeNormative,
         boolean includeInformative)
```

Returns all associations involving classes from owners `wg`.

Parameters:

`wgs` - one or more owners.
`kinds` - one or more association kinds.
`includeNormative` - whether to include normative associations.
`includeInformative` - whether to include informative associations.

findCimNoncimAssociations

```
public java.util.Collection findCimNoncimAssociations()
```

Returns all associations that are mappings between models of different natures.

(continued from last page)

getDependencies

```
public java.util.Collection getDependencies()
```

Returns all explicit (hand-drawn) dependencies in this model.

getDiagrams

```
public java.util.Collection getDiagrams()
```

Returns all diagrams in this model.

findDiagrams

```
public java.util.Collection findDiagrams(java.lang.String containerName,
         java.lang.String name,
         boolean includeOnPackage,
         boolean includeOnClass)
```

Returns all diagrams whose container name matches `containerName` and with name `name`, with the specified options applied.

Parameters:

`containerName` - name of the diagram's container.
`name` - name of the diagram.
`includeOnPackage` - includes diagrams defined on packages.
`includeOnClass` - includes diagrams defined on classes.

getTags

```
public java.util.Map getTags()
```

getModelNamesWithNature

```
public java.lang.String getModelNamesWithNature()
```

Returns comma-separated pairs {modelPackageName nature}.

getModelPackages

```
public java.util.Collection getModelPackages(java.util.EnumSet natures)
```

Returns model packages of specified nature.

getTopPackages

```
public java.util.Collection getTopPackages(java.util.EnumSet wgs)
```

Returns top packages for specified owners.

getNamespacePackages

```
public java.util.Collection getNamespacePackages(java.util.EnumSet wgs)
```

Returns packages that have namespace info for specified owners. TODO: tests

(continued from last page)

getVersionInfos

```
public java.util.Collection getVersionInfos(java.util.EnumSet wgs)
```

Returns version informations for top-level packages of specified owners.

getNamespaceInfos

```
public java.util.Collection getNamespaceInfos()
```

Returns non-null namespace informations for all packages. FIXME: tests

toString

```
public java.lang.String toString()
```

org.tanjakostic.jcleancim.model
Class UmlMultiplicity

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.model.UmlMultiplicity
```

public class **UmlMultiplicity**
extends java.lang.Object

We should use only 4 multiplicities in EA: [1], [1..*], [0..1] and [0..*]. Use `parseBounds(String, String)` to obtain one of those standard ones, or whatever is defined as custom. For validation purposes, use `isCustom()` to identify custom ones that should be fixed in the model.

Nested Class Summary

class	`UmlMultiplicity.Kind` UmlMultiplicity.Kind

Field Summary

public static final	`EMPTY`
public static final	`ONE`
public static final	`ONE_TO_MANY`
public static final	`OPT_MANY`
public static final	`OPT_ONE`

Method Summary

java.lang.String	`getBounds()`
java.lang.String	`getLower()`
java.lang.String	`getUpper()`
boolean	`isCustom()` Returns whether this is a custom instance returned from `parseBounds(String, String)` that should be fixed in the EA model.
boolean	`isMultivalue()` Returns true if the upper bound is not empty and different than 1.
boolean	`isOptional()` Returns true if the lower bound is 0 or empty.
static `UmlMultiplicity`	`parseBounds(java.lang.String lower, java.lang.String upper)` Returns multiplicity object from lower and upper bounds

static UmlMultiplicity	parseFromString(java.lang.String mult) Returns multiplicity object from formatted string "lower..upper".
java.lang.String	toString()

Methods inherited from class java.lang.Object
clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Fields

ONE

public static final org.tanjakostic.jcleancim.model.UmlMultiplicity **ONE**

ONE_TO_MANY

public static final org.tanjakostic.jcleancim.model.UmlMultiplicity **ONE_TO_MANY**

OPT_ONE

public static final org.tanjakostic.jcleancim.model.UmlMultiplicity **OPT_ONE**

OPT_MANY

public static final org.tanjakostic.jcleancim.model.UmlMultiplicity **OPT_MANY**

EMPTY

public static final org.tanjakostic.jcleancim.model.UmlMultiplicity **EMPTY**

Methods

getLower

public java.lang.String **getLower**()

getUpper

public java.lang.String **getUpper**()

isCustom

```
public boolean isCustom()
```

Returns whether this is a custom instance returned from `parseBounds(String, String)` that should be fixed in the EA model.

isOptional

```
public boolean isOptional()
```

Returns true if the lower bound is 0 or empty.

isMultivalue

```
public boolean isMultivalue()
```

Returns true if the upper bound is not empty and different than 1.

parseBounds

```
public static UmlMultiplicity parseBounds(java.lang.String lower,
         java.lang.String upper)
```

Returns multiplicity object from lower and upper bounds

parseFromString

```
public static UmlMultiplicity parseFromString(java.lang.String mult)
```

Returns multiplicity object from formatted string "lower..upper".

getBounds

```
public java.lang.String getBounds()
```

toString

```
public java.lang.String toString()
```

org.tanjakostic.jcleancim.model
Class UmlMultiplicity.Kind

```
java.lang.Object
  +-java.lang.Enum
    +-org.tanjakostic.jcleancim.model.UmlMultiplicity.Kind
```

All Implemented Interfaces:
java.io.Serializable, java.lang.Comparable

public static final class **UmlMultiplicity.Kind**
extends java.lang.Enum

Facilitates handling of IEC61850 class constraints (to ignore presence condition literals which must be printed in the documentation, but are not actually used as constraints, rather deduced from multiplicity of attribute).

Field Summary

public static final	M Mandatory.
public static final	O Optional.

Method Summary

static boolean	isMember(java.lang.String value)
static UmlMultiplicity.Kind	valueOf(java.lang.String name)
static UmlMultiplicity.Kind[]	values()

Methods inherited from class `java.lang.Enum`

clone, compareTo, equals, finalize, getDeclaringClass, hashCode, name, ordinal, toString, valueOf

Methods inherited from class `java.lang.Object`

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface `java.lang.Comparable`

compareTo

Fields

(continued from last page)

M

```
public static final org.tanjakostic.jcleancim.model.UmlMultiplicity.Kind M
```

Mandatory.

O

```
public static final org.tanjakostic.jcleancim.model.UmlMultiplicity.Kind O
```

Optional.

Methods

values

```
public static UmlMultiplicity.Kind[] values()
```

valueOf

```
public static UmlMultiplicity.Kind valueOf(java.lang.String name)
```

isMember

```
public static boolean isMember(java.lang.String value)
```

org.tanjakostic.jcleancim.model
Interface UmlObject

All Known Implementing Classes:
AbstractUmlObject

public interface **UmlObject**
extends

Data common to all UML objects.

Methods never return a null value, but rather empty string or empty collection.

Method Summary

abstract java.lang.String	addTaggedValue(java.lang.String name, java.lang.String value) Adds the UML tagged value (name, value pair) to this UML object, as defined in the UML model.
abstract java.lang.String	getAlias() Returns the alias of this UML object, as defined in the UML model, empty string if not defined.
abstract TextDescription	getDescription() Returns the raw text description for this UML object, as defined in the UML model.
abstract TextDescription	getHtmlDescription() Returns the formatted description for this UML object, as defined in the UML model.
abstract java.lang.Integer	getId() Returns the local ID of this UML object.
abstract UmlKind	getKind() Returns kind of this UML object, as assigned by the application.
abstract java.lang.String	getName() Returns the name of this UML object, as defined in the UML model.
abstract Namespace	getNamespace() Returns the namespace of this UML object, based on tagged values in the UML model and when missing, calculated by the application.
abstract Nature	getNature() Returns the nature of this UML object, which determines the validation rules to apply, and sometimes document generation formats.
abstract OwningWg	getOwner() Returns the IEC working group owning this UML object, as calculated by the application based on the UML model structure.
abstract java.util.Set	getPredefinedTagNames() Returns allowed tag names, as expected to be found in the UML model.
abstract java.lang.String	getQualifiedName() Returns the name of this UML object combined with some container-related information (e.g., packageName.className, or containingPackageName.packageName).

abstract java.lang.String	getSince() Returns the version of the model in which this UML object has been introduced, as defined in the UML model.
abstract UmlStereotype	getStereotype() Returns the stereotype of this UML object, as defined in the UML model.
abstract java.util.Map	getTaggedValues() Returns all the tagged values of this UML object, as defined in the UML model.
abstract java.util.Set	getUnallowedTagNames() Returns actual tag names defined for this object, but not found in getPredefinedTagNames().
abstract java.lang.String	getUuid() Returns the universally unique ID of this UML object.
abstract UmlVisibility	getVisibility() Returns visibility of this UML object, as defined in the UML model.
abstract boolean	isDeprecated() Returns whether this UML object is deprecated; this may be defined directly on this object with the stereotype UmlStereotype.DEPRECATED, or derived (for instance, for association ends of an association).
abstract boolean	isInformative() Returns whether this UML object is informative (and thus should be ignored when generating official IEC documents).
abstract java.lang.String	toShortString(boolean includeId, boolean isNameQualified) Returns the context as string, for logging purposes:

Methods

getId

```
public abstract java.lang.Integer getId()
```

Returns the local ID of this UML object.

In case of EA, this is an integer, assigned per EA type of tables (i.e., it's a counter) that cannot be changed and is a real persistent identifier for the given scope, so the application should be using this number to ensure uniqueness. However, some EA objects do not have this identifier at all, and the application is free to assign any number, assuming that that kind of UML object will not be cached on its own (but rather accessed from its container).

getUuid

```
public abstract java.lang.String getUuid()
```

Returns the universally unique ID of this UML object.

In case of EA, this is the UUID used for manipulating diagram objects (in contrast to everythin else) and when manipulating XMI, but it can be stripped off on model import/export so it cannot be considered as persistent. In case there is not UUID assigned within the model, the application should assign a valid UUID.

getSince

```
public abstract java.lang.String getSince()
```

(continued from last page)

Returns the version of the model in which this UML object has been introduced, as defined in the UML model.

getOwner

`public abstract OwningWg getOwner()`

Returns the IEC working group owning this UML object, as calculated by the application based on the UML model structure.

getNamespace

`public abstract Namespace getNamespace()`

Returns the namespace of this UML object, based on tagged values in the UML model and when missing, calculated by the application.

getNature

`public abstract Nature getNature()`

Returns the nature of this UML object, which determines the validation rules to apply, and sometimes document generation formats.

isInformative

`public abstract boolean isInformative()`

Returns whether this UML object is informative (and thus should be ignored when generating official IEC documents).

getVisibility

`public abstract UmlVisibility getVisibility()`

Returns visibility of this UML object, as defined in the UML model.

getKind

`public abstract UmlKind getKind()`

Returns kind of this UML object, as assigned by the application.

getName

`public abstract java.lang.String getName()`

Returns the name of this UML object, as defined in the UML model.

getAlias

`public abstract java.lang.String getAlias()`

Returns the alias of this UML object, as defined in the UML model, empty string if not defined. Typically used for a "pretty print" name of an UML object, as required for documentation generation.

getQualifiedName

`public abstract java.lang.String getQualifiedName()`

(continued from last page)

Returns the name of this UML object combined with some container-related information (e.g., packageName.className, or containingPackageName.packageName). This is meant to be used for displaying purposes, to facilitate locating UML objects.

toShortString

```
public abstract java.lang.String toShortString(boolean includeId,
          boolean isNameQualified)
```

Returns the context as string, for logging purposes:

```
owner nature [inf] [visibility] [qualifier] kind [stereotype] [q]name;
```

Parameters:

`includeId` - whether to print ID

`isNameQualified` - whether to print qualified name

getDescription

```
public abstract TextDescription getDescription()
```

Returns the raw text description for this UML object, as defined in the UML model. For formatted description, use `getHtmlDescription()`.

getHtmlDescription

```
public abstract TextDescription getHtmlDescription()
```

Returns the formatted description for this UML object, as defined in the UML model. For raw text description, use `getDescription()`.

getStereotype

```
public abstract UmlStereotype getStereotype()
```

Returns the stereotype of this UML object, as defined in the UML model.

isDeprecated

```
public abstract boolean isDeprecated()
```

Returns whether this UML object is deprecated; this may be defined directly on this object with the stereotype `UmlStereotype.DEPRECATED`, or derived (for instance, for association ends of an association).

getPredefinedTagNames

```
public abstract java.util.Set getPredefinedTagNames()
```

Returns allowed tag names, as expected to be found in the UML model.

getUnallowedTagNames

```
public abstract java.util.Set getUnallowedTagNames()
```

(continued from last page)

Returns actual tag names defined for this object, but not found in `getPredefinedTagNames()`.

addTaggedValue

```
public abstract java.lang.String addTaggedValue(java.lang.String name,
         java.lang.String value)
  throws InvalidTagException
```

Adds the UML tagged value (`name`, `value` pair) to this UML object, as defined in the UML model.

Parameters:

`name` - tag name.
`value` - tag value.

Returns:

null if the `name` is a new tag, otherwise old value for `name` thas has been overwritten with `value`.

Throws:

`InvalidTagException` - if either `name` or `value` is invalid.

getTaggedValues

```
public abstract java.util.Map getTaggedValues()
```

Returns all the tagged values of this UML object, as defined in the UML model.

org.tanjakostic.jcleancim.model
Class UmlObjectData

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.model.UmlObjectData
```

public class **UmlObjectData**
extends java.lang.Object

Simple data structure that allows us to instantiate a subset of data of any `UmlObject` that can be initialised simply without any validation logic. This facilitates creation from both real UML model (with builders), and from within the API (for testing without real UML model).

TODO: see whether we'll need the attribute since (for IEC 61850).

Implementation note: We have considered having this class inherit from `AbstractUmlObject` and then have concrete classes (such as, e.g., `UmlPackage`) inherit form this. However, we have discarded this option because it would force us to do lots of checks in the code to avoid NPEs. We also considered making the data containers (such as `UmlPackage.Data`) inherit from this one, but then the creation of immutable objects (i.e., using purely ctor params) would become extremely cumbersome. So, we prefer using this type in composition, which is better modular and better testable.

Constructor Summary

public	`UmlObjectData(java.lang.String name, java.lang.String alias, TextDescription txtDoc, TextDescription htmlDoc)` Constructor useful when collecting model content for documentation.
public	`UmlObjectData(java.lang.Integer id, java.lang.String uuid, java.lang.String name, java.lang.String alias, UmlStereotype stereotype, java.lang.String eaVisibility, TextDescription txtDoc, TextDescription htmlDoc)` Constructor; accepts null arguments and initialises them with default values, so that all the getters return non-null values.
public	`UmlObjectData(UmlObject o)` Copy constructor.

Method Summary

java.lang.String	`getAlias()`
TextDescription	`getHtmlDescription()`
java.lang.Integer	`getId()`
java.lang.String	`getName()`
java.lang.String	`getSince()`
UmlStereotype	`getStereotype()`
TextDescription	`getTxtDescription()`

java.lang.String	getUuid()
UmlVisibility	getVisibility()
java.lang.String	toString()

Methods inherited from class java.lang.Object
clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Constructors

UmlObjectData

```
public UmlObjectData(java.lang.String name,
                     java.lang.String alias,
                     TextDescription txtDoc,
                     TextDescription htmlDoc)
```

Constructor useful when collecting model content for documentation.

UmlObjectData

```
public UmlObjectData(java.lang.Integer id,
                     java.lang.String uuid,
                     java.lang.String name,
                     java.lang.String alias,
                     UmlStereotype stereotype,
                     java.lang.String eaVisibility,
                     TextDescription txtDoc,
                     TextDescription htmlDoc)
```

Constructor; accepts null arguments and initialises them with default values, so that all the getters return non-null values. In case an argument is a null object, stores empty string except for `eaVisibility` (sets it to `UmlVisibility.PUBLIC`) and for `uuid` (generates random UUID). In such a way, all the getters return non-null values.

Parameters:

`id` - if null, sets it to auto-generated sequence number.
`uuid` - if null, sets it to an UUID generated from combination of id and name (note that this is not guaranteed to produce a unique UUID, but we need repeatable values for comparisons in tests and for debugging).
`name` - if null, sets it to empty string.
`alias` - if null, sets it to empty string.
`stereotype` - if null, sets it to empty stereotype.
`eaVisibility` - if null, sets it to `UmlVisibility.PUBLIC`.
`txtDoc` - if null, sets it to empty string.
`htmlDoc` - if null, sets it to empty string.

UmlObjectData

```
public UmlObjectData(UmlObject o)
```

Copy constructor.

Methods

(continued from last page)

getId

public java.lang.Integer **getId**()

getUuid

public java.lang.String **getUuid**()

getSince

public java.lang.String **getSince**()

getName

public java.lang.String **getName**()

getAlias

public java.lang.String **getAlias**()

getStereotype

public UmlStereotype **getStereotype**()

getVisibility

public UmlVisibility **getVisibility**()

getTxtDescription

public TextDescription **getTxtDescription**()

getHtmlDescription

public TextDescription **getHtmlDescription**()

toString

public java.lang.String **toString**()

(continued from last page)

org.tanjakostic.jcleancim.model
Class UmlOperation

```
java.lang.Object
  +-org.tanjakostic.jcleancim.model.AbstractUmlObject
    +-org.tanjakostic.jcleancim.model.UmlOperation
```

All Implemented Interfaces:
UmlObject

public class **UmlOperation**
extends AbstractUmlObject

UML operation.

Nested Class Summary

class	UmlOperation.Data UmlOperation.Data
class	UmlOperation.ReturnKind UmlOperation.ReturnKind

Fields inherited from class org.tanjakostic.jcleancim.model.AbstractUmlObject
CLASS_SEPARATOR, NULL_OBJ_NAME, PACKAGE_SEPARATOR

Method Summary

UmlClass	addException(UmlClass exc) Adds non-null class exc to this operation, and returns the same object.
UmlParameter	addParameter(UmlParameter par) Adds non-null parameter par to this operation, and returns the same object.
UmlClass	getContainingClass()
java.lang.String	getEaExceptionTypeInfo(int i)
int	getEaReturnTypeId()
java.lang.String	getEaReturnTypeInfo() Returns known (string) info from EA; useful to display in case the return type of this operation in EA model is not a valid UML class, so the model can be corrected.
java.lang.String	getEaReturnTypeName()
java.util.Set	getEfferentClasses() Returns all classes that I use as type for return value, parameter or exception.
java.util.List	getExceptions() Returns all exceptions declared for this operation.

java.lang.String	getExceptionsSignature() Returns potentially empty string containing comma-separated list of exceptions that follow the 'throw' statement in operation signature.
UmlKind	getKind()
static java.util.List	getKinds(Nature nature) Returns all available classifications (kinds) for operations.
Namespace	getNamespace()
Nature	getNature()
OwningWg	getOwner()
java.util.List	getParameters() Returns all parameters of this operation.
java.util.Set	getPredefinedTagNames()
java.lang.String	getQualifiedName()
UmlClass	getReturnType() Returns return type of this operation, null if kind is UmlOperation.ReturnKind.OP_RET_VOID.
java.lang.String	getSignature() E.g.
boolean	isAbstract()
boolean	isFinal()
boolean	isInformative()
boolean	isStatic()
boolean	isVoidReturned()
java.lang.String	toString()

Methods inherited from class org.tanjakostic.jcleancim.model.AbstractUmlObject

addDeprecAndInf, addTaggedValue, appendRemainingCustomStereotypes, classifyPerScope, classifyPerScopePerTag, classifyPerTag, collectDuplicateDescriptions, collectDuplicateNames, collectForScope, collectNames, collectQNames, findAllForName, findWithSameUuidAndLog, getAlias, getDescription, getHtmlDescription, getId, getKind, getName, getNature, getOwner, getQualifiedName, getSince, getStereotype, getTaggedValues, getUnallowedTagNames, getUuid, getVisibility, initFromTags, isDeprecated, isInformative, saveTags, toShortString, toShortString, validateTag

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.model.UmlObject

addTaggedValue, getAlias, getDescription, getHtmlDescription, getId, getKind, getName, getNamespace, getNature, getOwner, getPredefinedTagNames, getQualifiedName, getSince, getStereotype, getTaggedValues, getUnallowedTagNames, getUuid, getVisibility, isDeprecated, isInformative, toShortString

Methods

getKinds

public static java.util.List **getKinds**(Nature nature)

Returns all available classifications (kinds) for operations.

Parameters:
nature - ignored in this method

getContainingClass

public UmlClass **getContainingClass**()

isAbstract

public boolean **isAbstract**()

isStatic

public boolean **isStatic**()

isFinal

public boolean **isFinal**()

isVoidReturned

public boolean **isVoidReturned**()

getEaReturnTypeId

public int **getEaReturnTypeId**()

(continued from last page)

getEaReturnTypeName

`public java.lang.String getEaReturnTypeName()`

getEaReturnTypeInfo

`public java.lang.String getEaReturnTypeInfo()`

Returns known (string) info from EA; useful to display in case the return type of this operation in EA model is not a valid UML class, so the model can be corrected.

getEaExceptionTypeInfo

`public java.lang.String getEaExceptionTypeInfo(int i)`

getReturnType

`public UmlClass getReturnType()`

Returns return type of this operation, null if kind is `UmlOperation.ReturnKind.OP_RET_VOID`.

addParameter

`public UmlParameter addParameter(UmlParameter par)`

Adds non-null parameter `par` to this operation, and returns the same object.

getParameters

`public java.util.List getParameters()`

Returns all parameters of this operation.

addException

`public UmlClass addException(UmlClass exc)`

Adds non-null class `exc` to this operation, and returns the same object.

getExceptions

`public java.util.List getExceptions()`

Returns all exceptions declared for this operation.

getEfferentClasses

`public java.util.Set getEfferentClasses()`

Returns all classes that I use as type for return value, parameter or exception.

(continued from last page)

getExceptionsSignature

`public java.lang.String getExceptionsSignature()`

Returns potentially empty string containing comma-separated list of exceptions that follow the 'throw' statement in operation signature.

getSignature

`public java.lang.String getSignature()`

E.g. "abstract public static RetType[] foo(C1 arg1, C2[] arg2) throws SomeExc, OthExc".

getOwner

`public OwningWg getOwner()`

getNamespace

`public Namespace getNamespace()`

Returns own namespace initialised from tagged values if not empty. Otherwise, returns the namespace of the containing class.

getNature

`public Nature getNature()`

isInformative

`public boolean isInformative()`

This default implementation returns true if objects steretypes include UmlStereotype#INFORMATIVE. If there are additional criteria for deriving informative status, ensure to invoke this default implementation first.

getKind

`public UmlKind getKind()`

getQualifiedName

`public java.lang.String getQualifiedName()`

getPredefinedTagNames

`public java.util.Set getPredefinedTagNames()`

(continued from last page)

toString

```
public java.lang.String toString()
```

org.tanjakostic.jcleancim.model
Class UmlOperation.ReturnKind

```
java.lang.Object
  +-java.lang.Enum
    +-org.tanjakostic.jcleancim.model.UmlOperation.ReturnKind
```

All Implemented Interfaces:
UmlKind, java.io.Serializable, java.lang.Comparable

---

public static final class **UmlOperation.ReturnKind**
extends java.lang.Enum
implements java.lang.Comparable, java.io.Serializable, UmlKind

---

Field Summary

public static final	OP_RET_ARRAY
public static final	OP_RET_SIMPLE
public static final	OP_RET_VOID

Method Summary

java.lang.String	getDesc()
java.lang.String	getLabel()
java.lang.String	getTag()
java.lang.String	getValue()
static UmlOperation.ReturnKind	valueOf(java.lang.String name)
static UmlOperation.ReturnKind[]	values()

Methods inherited from class `java.lang.Enum`

clone, compareTo, equals, finalize, getDeclaringClass, hashCode, name, ordinal, toString, valueOf

Methods inherited from class `java.lang.Object`

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface `java.lang.Comparable`
`compareTo`

Methods inherited from interface `org.tanjakostic.jcleancim.model.UmlKind`
`getDesc`, `getLabel`, `getTag`, `getValue`

Fields

OP_RET_VOID

```
public static final org.tanjakostic.jcleancim.model.UmlOperation.ReturnKind
OP_RET_VOID
```

OP_RET_ARRAY

```
public static final org.tanjakostic.jcleancim.model.UmlOperation.ReturnKind
OP_RET_ARRAY
```

OP_RET_SIMPLE

```
public static final org.tanjakostic.jcleancim.model.UmlOperation.ReturnKind
OP_RET_SIMPLE
```

Methods

values

```
public static UmlOperation.ReturnKind[] values()
```

valueOf

```
public static UmlOperation.ReturnKind valueOf(java.lang.String name)
```

getValue

```
public java.lang.String getValue()
```

getLabel

```
public java.lang.String getLabel()
```

getTag

```
public java.lang.String getTag()
```

getDesc

```
public java.lang.String getDesc()
```

org.tanjakostic.jcleancim.model
Class UmlOperation.Data

```
java.lang.Object
   |
   +-org.tanjakostic.jcleancim.model.UmlOperation.Data
```

public static class **UmlOperation.Data**
extends java.lang.Object

Data from the UML model repository specific to `UmlOperation`.

Constructor Summary

public	`Data(UmlOperation.ReturnKind kind, boolean isAbstract, boolean isStatic, boolean isFinal, int eaReturnTypeId, java.lang.String eaReturnTypeName, java.util.List eaExceptionTypeInfo)` Constructor.

Method Summary

static `UmlOperation.Data`	`empty()` Returns empty instance; sets default return kind to ReturnKind#OP_RET_VOID.
`java.util.List`	`getEaExceptionTypeInfo()`
`int`	`getEaReturnTypeId()`
`java.lang.String`	`getEaReturnTypeName()`
`UmlOperation.ReturnKind`	`getKind()`
`boolean`	`isAbstract()`
`boolean`	`isFinal()`
`boolean`	`isStatic()`

Methods inherited from class `java.lang.Object`

`clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait`

Constructors

(continued from last page)

Data

```
public Data(UmlOperation.ReturnKind kind,
            boolean isAbstract,
            boolean isStatic,
            boolean isFinal,
            int eaReturnTypeId,
            java.lang.String eaReturnTypeName,
            java.util.List eaExceptionTypeInfo)
```

Constructor.

Parameters:

`kind`
`isAbstract`
`isStatic`
`isFinal`
`eaReturnTypeId`
`eaReturnTypeName`
`eaExceptionTypeInfo`

Methods

empty

```
public static UmlOperation.Data empty()
```

Returns empty instance; sets default return kind to ReturnKind#OP_RET_VOID.

getKind

```
public UmlOperation.ReturnKind getKind()
```

isAbstract

```
public boolean isAbstract()
```

isStatic

```
public boolean isStatic()
```

isFinal

```
public boolean isFinal()
```

getEaReturnTypeId

```
public int getEaReturnTypeId()
```

getEaReturnTypeName

```
public java.lang.String getEaReturnTypeName()
```

getEaExceptionTypeInfo

```
public java.util.List getEaExceptionTypeInfo()
```

org.tanjakostic.jcleancim.model
Class UmlPackage

```
java.lang.Object
  |-org.tanjakostic.jcleancim.model.AbstractUmlObject
    |-org.tanjakostic.jcleancim.model.UmlStructure
      |-org.tanjakostic.jcleancim.model.UmlPackage
```

All Implemented Interfaces:
UmlObject

---

public class **UmlPackage**
extends UmlStructure

UML package and its sub-packages hold the content of the model. In addition to UML features specific to packages, it inherits implementation for features common with UML classes from `UmlStructure` (so we avoid code duplication).

Implementation note: We distinguish among hierarchy levels of packages by their `UmlPackage.Kind`, to ensure we can manage the actual UML model repositories in a loosely coupled way. This also allows us to filter (opt in/out) parts of the in-memory model for e.g. validation, statistics or document generation.

A cleaner design would be to effectively create subclasses instead of using the above kinds, but it would be overkill for minor differences in functionality per kind.

Nested Class Summary

class	`UmlPackage.Data` UmlPackage.Data
class	`UmlPackage.Kind` UmlPackage.Kind

Fields inherited from class `org.tanjakostic.jcleancim.model.AbstractUmlObject`
`CLASS_SEPARATOR`, `NULL_OBJ_NAME`, `PACKAGE_SEPARATOR`

Constructor Summary

public	`UmlPackage(UmlModel model, UmlObjectData objData, UmlPackage.Data data)` Creates a model package (e.g.
public	`UmlPackage(UmlPackage containingPackage, UmlObjectData objData, UmlPackage.Data data)` Creates a top-level or a regular package, and adds itself to `containingPackage`.

Method Summary

static UmlPackage	`basic(UmlModel model, java.lang.String name)` Constructs minimal model package - useful for creation from profiles and testing.
static UmlPackage	`basic(UmlPackage containingPackage, java.lang.String name)` Constructs minimal non-model package - useful for creation from profiles and testing.

static UmlPackage	basic(UmlPackage containingPackage, java.lang.String name, java.lang.String stereotype) Constructs minimal non-model package with stereotype(s) - useful for creation from profiles and testing.
java.util.Collection	collectDependencyEfferentPackages() Returns all classes that I depend on through an explicit UML dependency in the model.
java.util.Set	getChildPackages() Returns all direct sub-packages.
java.util.Set	getChildPackages(java.lang.String name) Returns sub-packages with name.
java.util.Set	getClasses() Returns all classes in this package.
UmlStructure	getContainer() Returns containing structure, null in case this is the model package.
UmlPackage	getContainingPackage() Returns the containing package, null for UmlPackage.Kind.MODEL.
int	getDepth() Returns the depth of this package, relative to the top-level package (i.e., for model package returns -1, for top-level package returns 0, and for all other packages returns positive offset relative to top-level package).
UmlKind	getKind()
static java.util.List	getKinds(Nature nature) Returns all available classifications (kinds) for packages.
UmlModel	getModel() Returns the model this structure belongs to.
Namespace	getNamespace()
NamespaceInfo	getNamespaceInfo() (lazy loaded) Returns the namespace information in case the relevant namespace class (for IEC61850) or version class (for CIM) is defined in the package, null otherwise.
Nature	getNature()
OwningWg	getOwner()
java.util.Set	getPredefinedTagNames()
java.lang.String	getQualifiedName()
VersionInfo	getVersionInfo() (lazy loaded) Returns the version information in case the relevant version class is defined in the package, null otherwise.
boolean	isInformative() This default implementation returns true if objects steretypes include UmlStereotype#INFORMATIVE. If there are additional criteria for deriving informative status, ensure to invoke this default implementation first.

boolean	isInOrUnderPackage(java.lang.String packageName) Returns whether this package or anywhere below it (recursively) there is a package called packageName.
boolean	isTop() Returns whether this package is a top package (owned by a WG).
boolean	isUnderPackage(java.lang.String packageName) Returns whether anywhere below this package (recursively) there is a package called packageName.
void	orderClasses(java.util.List uuids) Orders classes in the order given in uuids.
boolean	shouldExportDiagrams()
java.lang.String	toString()

Methods inherited from class org.tanjakostic.jcleancim.model.UmlStructure

addDependency, addDiagram, addSkippedUmlItem, collectDependencyAfferentStructures, collectDependencyEfferentStructures, collectMyAndParentsDependencyEfferentStructures, getContainer, getDependenciesAsSource, getDependenciesAsTarget, getDiagrams, getKind, getModel, getNature, getOwner, getQualifiedName, getSkippedUmlItems, isSelfDependent

Methods inherited from class org.tanjakostic.jcleancim.model.AbstractUmlObject

addDeprecAndInf, addTaggedValue, appendRemainingCustomStereotypes, classifyPerScope, classifyPerScopePerTag, classifyPerTag, collectDuplicateDescriptions, collectDuplicateNames, collectForScope, collectNames, collectQNames, findAllForName, findWithSameUuidAndLog, getAlias, getDescription, getHtmlDescription, getId, getKind, getName, getNature, getOwner, getQualifiedName, getSince, getStereotype, getTaggedValues, getUnallowedTagNames, getUuid, getVisibility, initFromTags, isDeprecated, isInformative, saveTags, toShortString, toShortString, validateTag

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.model.UmlObject

addTaggedValue, getAlias, getDescription, getHtmlDescription, getId, getKind, getName, getNamespace, getNature, getOwner, getPredefinedTagNames, getQualifiedName, getSince, getStereotype, getTaggedValues, getUnallowedTagNames, getUuid, getVisibility, isDeprecated, isInformative, toShortString

Constructors

UmlPackage

```
public UmlPackage(UmlModel model,
                  UmlObjectData objData,
                  UmlPackage.Data data)
```

(continued from last page)

Creates a model package (e.g. TC57CIM, TC57CIMProfiles and IEC61850Domain), and adds itself to `model`. After creating this object, you may want to add tagged values, classes and child packages (as well as other objects - see `org.tanjakostic.jcleancim.model.UmlStructure(UmlObjectData, UmlStructure.Data)`).

Parameters:

`model` - parent UML model
`objData` - common data for every `UmlObject`s.
`data` - data proper to `UmlPackage`.

UmlPackage

```
public UmlPackage(UmlPackage containingPackage,
                  UmlObjectData objData,
                  UmlPackage.Data data)
```

Creates a top-level or a regular package, and adds itself to `containingPackage`. After creating this object, you may want to add tagged values, classes and child packages (as well as other objects - see `org.tanjakostic.jcleancim.model.UmlStructure(UmlObjectData, UmlStructure.Data)`).

Parameters:

`containingPackage` - parent UML package
`objData` - common data for every `UmlObject`.
`data` - data proper to `UmlPackage`.

Methods

getKinds

```
public static java.util.List getKinds(Nature nature)
```

Returns all available classifications (kinds) for packages.

Parameters:

`nature` - ignored in this method

basic

```
public static UmlPackage basic(UmlModel model,
         java.lang.String name)
```

Constructs minimal model package - useful for creation from profiles and testing.

basic

```
public static UmlPackage basic(UmlPackage containingPackage,
         java.lang.String name)
```

Constructs minimal non-model package - useful for creation from profiles and testing.

basic

```
public static UmlPackage basic(UmlPackage containingPackage,
         java.lang.String name,
         java.lang.String stereotype)
```

Constructs minimal non-model package with stereotype(s) - useful for creation from profiles and testing.

shouldExportDiagrams

```
public boolean shouldExportDiagrams()
```

(continued from last page)

getContainingPackage

`public UmlPackage getContainingPackage()`

Returns the containing package, null for `UmlPackage.Kind.MODEL`.

isTop

`public boolean isTop()`

Returns whether this package is a top package (owned by a WG).

getDepth

`public int getDepth()`

Returns the depth of this package, relative to the top-level package (i.e., for model package returns -1, for top-level package returns 0, and for all other packages returns positive offset relative to top-level package). Used for having proper headings depth for document generation (and optionally, for indentation in logs). For model package returns -1, for top-level package 0, and for all other packages

getClasses

`public java.util.Set getClasses()`

Returns all classes in this package.

orderClasses

`public void orderClasses(java.util.List uuids)`

Orders classes in the order given in `uuids`.

collectDependencyEfferentPackages

`public java.util.Collection collectDependencyEfferentPackages()`

Returns all classes that I depend on through an explicit UML dependency in the model.

getVersionInfo

`public VersionInfo getVersionInfo()`

(lazy loaded) Returns the version information in case the relevant version class is defined in the package, null otherwise. Logs error if more than one version class found and retains only one.

getNamespaceInfo

`public NamespaceInfo getNamespaceInfo()`

(lazy loaded) Returns the namespace information in case the relevant namespace class (for IEC61850) or version class (for CIM) is defined in the package, null otherwise. Logs error if more than one namespace class found and retains only one.

getChildPackages

`public java.util.Set getChildPackages()`

(continued from last page)

Returns all direct sub-packages.

getChildPackages

`public java.util.Set getChildPackages(java.lang.String name)`

Returns sub-packages with `name`. Normally, sub-packages should have name unique within the containing package. However some tools allow this anomaly and we need to support that kind of result (with a set returned).

isUnderPackage

`public boolean isUnderPackage(java.lang.String packageName)`

Returns whether anywhere below this package (recursively) there is a package called `packageName`. To include this package in the search, use `isInOrUnderPackage(String)`.

isInOrUnderPackage

`public boolean isInOrUnderPackage(java.lang.String packageName)`

Returns whether this package or anywhere below it (recursively) there is a package called `packageName`. To exclude this package from the search, use `isUnderPackage(String)`.

getModel

`public UmlModel getModel()`

Returns the model this structure belongs to.

Returns the model this package and all its recursive contents belongs to.

getContainer

`public UmlStructure getContainer()`

Returns containing structure, null in case this is the model package.

See `getContainingPackage()`.

getOwner

`public OwningWg getOwner()`

getNamespace

`public Namespace getNamespace()`

Returns own namespace initialised from tagged values if not empty. Otherwise, returns the namespace of the containing package, or empty namespace otherwise.

getNature

`public Nature getNature()`

isInformative

```
public boolean isInformative()
```

This default implementation returns true if objects steretypes include UmlStereotype#INFORMATIVE. If there are additional criteria for deriving informative status, ensure to invoke this default implementation first.

Model package is never informative.

Any other package is considered as informative if any of the following is true:

- package stereotype includes UmlStereotype#INFORMATIVE,
- package name starts with `UML.INF_PREFIX`,
- package name is `UML.DetailedDiagrams`,
- any parent package (in the chain) is informative.

getKind

```
public UmlKind getKind()
```

getQualifiedName

```
public java.lang.String getQualifiedName()
```

getPredefinedTagNames

```
public java.util.Set getPredefinedTagNames()
```

toString

```
public java.lang.String toString()
```

org.tanjakostic.jcleancim.model
Class UmlPackage.Kind

```
java.lang.Object
  +-java.lang.Enum
    +-org.tanjakostic.jcleancim.model.UmlPackage.Kind
```

All Implemented Interfaces:
UmlKind, java.io.Serializable, java.lang.Comparable

---

public static final class **UmlPackage.Kind**
extends java.lang.Enum
implements java.lang.Comparable, java.io.Serializable, UmlKind

Kind of UML package, reflecting hierarchical package containment and common properties for the model content below a given level.

Field Summary

public static final	MODEL Direct child of a root in UML repository (level 2); we have CIM and IEC 61850 model(s).
public static final	NULL_MODEL This is like MODEL (level 2), but reserved for use by UmlModel.
public static final	PACKAGE Any direct or deep child of the TOP (level 4+).
public static final	TOP Direct child of the MODEL (level 3).

Method Summary

java.lang.String	getDesc()
java.lang.String	getLabel()
java.lang.String	getTag()
java.lang.String	getValue()
static UmlPackage.Kind	valueOf(java.lang.String name)
static UmlPackage.Kind[]	values()

Methods inherited from class java.lang.Enum

clone, compareTo, equals, finalize, getDeclaringClass, hashCode, name, ordinal, toString, valueOf

Methods inherited from class `java.lang.Object`
`clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait`

Methods inherited from interface `java.lang.Comparable`
`compareTo`

Methods inherited from interface `org.tanjakostic.jcleancim.model.UmlKind`
`getDesc`, `getLabel`, `getTag`, `getValue`

Fields

MODEL

`public static final org.tanjakostic.jcleancim.model.UmlPackage.Kind` **`MODEL`**

Direct child of a root in UML repository (level 2); we have CIM and IEC 61850 model(s).

NULL_MODEL

`public static final org.tanjakostic.jcleancim.model.UmlPackage.Kind` **`NULL_MODEL`**

This is like `MODEL` (level 2), but reserved for use by `UmlModel`.

TOP

`public static final org.tanjakostic.jcleancim.model.UmlPackage.Kind` **`TOP`**

Direct child of the `MODEL` (level 3). We have WGs as owners of these.

PACKAGE

`public static final org.tanjakostic.jcleancim.model.UmlPackage.Kind` **`PACKAGE`**

Any direct or deep child of the `TOP` (level 4+).

Methods

values

`public static UmlPackage.Kind[]` **`values`**`()`

valueOf

`public static UmlPackage.Kind` **`valueOf`**`(java.lang.String name)`

(continued from last page)

getValue

```
public java.lang.String getValue()
```

getLabel

```
public java.lang.String getLabel()
```

getTag

```
public java.lang.String getTag()
```

getDesc

```
public java.lang.String getDesc()
```

org.tanjakostic.jcleancim.model
Class UmlPackage.Data

```
java.lang.Object
  +-org.tanjakostic.jcleancim.model.UmlStructure.Data
    +-org.tanjakostic.jcleancim.model.UmlPackage.Data
```

public static class **UmlPackage.Data**
extends UmlStructure.Data

Data from the UML model repository specific to `UmlPackage`.

Constructor Summary

public	`Data(UmlStructure.Data data)` Constructor.

Method Summary

static `UmlPackage.Data`	`empty()` Returns an empty instance.

Methods inherited from class `org.tanjakostic.jcleancim.model.UmlStructure.Data`

`empty`, `isSelfDependent`

Methods inherited from class `java.lang.Object`

`clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait`

Constructors

Data

`public` **`Data`**`(UmlStructure.Data data)`

Constructor.

Methods

empty

`public static UmlPackage.Data` **`empty`**`()`

Returns an empty instance.

org.tanjakostic.jcleancim.model
Class UmlParameter

```
java.lang.Object
  +-org.tanjakostic.jcleancim.model.AbstractUmlObject
    +-org.tanjakostic.jcleancim.model.UmlParameter
```

All Implemented Interfaces:
UmlObject

---

public class **UmlParameter**
extends AbstractUmlObject

Operation parameter.

Nested Class Summary

class	UmlParameter.Data UmlParameter.Data
class	UmlParameter.Kind UmlParameter.Kind

Fields inherited from class org.tanjakostic.jcleancim.model.AbstractUmlObject
CLASS_SEPARATOR, NULL_OBJ_NAME, PACKAGE_SEPARATOR

Constructor Summary

public	UmlParameter(UmlClass type, UmlObjectData objData, UmlParameter.Data data) Constructor.

Method Summary

UmlOperation	getContainingOperation()
java.lang.String	getEaTypeInfo() Returns known (string) info from EA; useful to display in case the type of this parameter in EA model is not a valid UML class, so the model can be corrected.
UmlKind	getKind()
Namespace	getNamespace()
Nature	getNature()
OwningWg	getOwner()
java.util.Set	getPredefinedTagNames()

java.lang.String	getQualifiedName()
java.lang.String	getSignature() E.g., "MyClass[] myArg".
UmlClass	getType()
boolean	isArray()
boolean	isInformative() This default implementation returns true if objects steretypes include UmlStereotype#INFORMATIVE. If there are additional criteria for deriving informative status, ensure to invoke this default implementation first.
java.lang.String	toString()

Methods inherited from class org.tanjakostic.jcleancim.model.AbstractUmlObject

addDeprecAndInf, addTaggedValue, appendRemainingCustomStereotypes, classifyPerScope, classifyPerScopePerTag, classifyPerTag, collectDuplicateDescriptions, collectDuplicateNames, collectForScope, collectNames, collectQNames, findAllForName, findWithSameUuidAndLog, getAlias, getDescription, getHtmlDescription, getId, getKind, getName, getNature, getOwner, getQualifiedName, getSince, getStereotype, getTaggedValues, getUnallowedTagNames, getUuid, getVisibility, initFromTags, isDeprecated, isInformative, saveTags, toShortString, toShortString, validateTag

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.model.UmlObject

addTaggedValue, getAlias, getDescription, getHtmlDescription, getId, getKind, getName, getNamespace, getNature, getOwner, getPredefinedTagNames, getQualifiedName, getSince, getStereotype, getTaggedValues, getUnallowedTagNames, getUuid, getVisibility, isDeprecated, isInformative, toShortString

Constructors

UmlParameter

```
public UmlParameter(UmlClass type,
                    UmlObjectData objData,
                    UmlParameter.Data data)
```

Constructor. After creating this object, you may want to add tagged values. In every case, the operation that will get an instances of this, has to use `setContainingOperation(UmlOperation)` to correctly set reference.

Parameters:

type
objData
data

Throws:

(continued from last page)

`NullPointerException` - if any argument is null.

Methods

getContainingOperation

`public UmlOperation getContainingOperation()`

getEaTypeInfo

`public java.lang.String getEaTypeInfo()`

Returns known (string) info from EA; useful to display in case the type of this parameter in EA model is not a valid UML class, so the model can be corrected.

isArray

`public boolean isArray()`

getType

`public UmlClass getType()`

getSignature

`public java.lang.String getSignature()`

E.g., "MyClass[] myArg".

getOwner

`public OwningWg getOwner()`

Returns owner, null if parameter has not yet been added to its operation.

getNamespace

`public Namespace getNamespace()`

Returns own namespace initialised from tagged values if not empty. Otherwise, returns null if parameter has not yet been added to its operation, or operation's namespace.

getNature

`public Nature getNature()`

Returns nature, null if parameter has not yet been added to its operation.

isInformative

```
public boolean isInformative()
```

This default implementation returns true if objects steretypes include UmlStereotype#INFORMATIVE. If there are additional criteria for deriving informative status, ensure to invoke this default implementation first.

Returns whether this is informative, false if parameter has not yet been added to its operation.

getKind

```
public UmlKind getKind()
```

getQualifiedName

```
public java.lang.String getQualifiedName()
```

getPredefinedTagNames

```
public java.util.Set getPredefinedTagNames()
```

toString

```
public java.lang.String toString()
```

org.tanjakostic.jcleancim.model
Class UmlParameter.Kind

```
java.lang.Object
  +-java.lang.Enum
    +-org.tanjakostic.jcleancim.model.UmlParameter.Kind
```

All Implemented Interfaces:
UmlKind, java.io.Serializable, java.lang.Comparable

---

public static final class **UmlParameter.Kind**
extends java.lang.Enum
implements java.lang.Comparable, java.io.Serializable, UmlKind

---

Field Summary

public static final	ARRAY
public static final	SIMPLE

Method Summary

java.lang.String	getDesc()
java.lang.String	getLabel()
java.lang.String	getTag()
java.lang.String	getValue()
static UmlParameter.Kind	valueOf(java.lang.String name)
static UmlParameter.Kind[]	values()

Methods inherited from class `java.lang.Enum`

clone, compareTo, equals, finalize, getDeclaringClass, hashCode, name, ordinal, toString, valueOf

Methods inherited from class `java.lang.Object`

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface `java.lang.Comparable`

compareTo

Methods inherited from interface org.tanjakostic.jcleancim.model.UmlKind
getDesc, getLabel, getTag, getValue

Fields

ARRAY

public static final org.tanjakostic.jcleancim.model.UmlParameter.Kind **ARRAY**

SIMPLE

public static final org.tanjakostic.jcleancim.model.UmlParameter.Kind **SIMPLE**

Methods

values

public static UmlParameter.Kind[] **values**()

valueOf

public static UmlParameter.Kind **valueOf**(java.lang.String name)

getValue

public java.lang.String **getValue**()

getLabel

public java.lang.String **getLabel**()

getTag

public java.lang.String **getTag**()

getDesc

public java.lang.String **getDesc**()

org.tanjakostic.jcleancim.model
Class UmlParameter.Data

```
java.lang.Object
   |
   +-org.tanjakostic.jcleancim.model.UmlParameter.Data
```

public static class **UmlParameter.Data**
extends java.lang.Object

Data from the UML model repository specific to `UmlParameter`.

Constructor Summary

public	`Data(UmlKind kind, java.lang.String eaTypeInfo)` Constructor.

Method Summary

`java.lang.String`	`getEaTypeInfo()`
`UmlKind`	`getKind()`

Methods inherited from class `java.lang.Object`
`clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait`

Constructors

Data

```
public Data(UmlKind kind,
            java.lang.String eaTypeInfo)
```

Constructor.

Parameters:
`kind`
`eaTypeInfo`

Methods

getKind

```
public UmlKind getKind()
```

(continued from last page)

getEaTypeInfo

```
public java.lang.String getEaTypeInfo()
```

org.tanjakostic.jcleancim.model
Class UmlSkipped

```
java.lang.Object
  +-org.tanjakostic.jcleancim.model.AbstractUmlObject
    +-org.tanjakostic.jcleancim.model.UmlSkipped
```

All Implemented Interfaces:
UmlObject

---

public class **UmlSkipped**
extends AbstractUmlObject

UML element or connector that we ignore, but track for validation purposes.

Design note: We could have had four subclasses, but it would have been an overkill at this point in time.

Nested Class Summary

class	UmlSkipped.Data UmlSkipped.Data
class	UmlSkipped.Kind UmlSkipped.Kind

Field Summary

public static final	EA_BOUNDARY Value: **Boundary**
public static final	EA_NOTE Value: **Note**
public static final	EA_NOTE_LINK Value: **NoteLink**
public static final	EA_PROCESS Value: **Process**
public static final	EA_STATE Value: **State**
public static final	EA_STATE_MACHINE Value: **StateMachine**
public static final	EA_STATE_NODE Value: **StateNode**

public static final	EA_TEXT Value: **Text**

Fields inherited from class org.tanjakostic.jcleancim.model.AbstractUmlObject

CLASS_SEPARATOR, NULL_OBJ_NAME, PACKAGE_SEPARATOR

Method Summary

UmlStructure	getContainer() Returns the container of this skipped element, or the source side if this is a connector.
UmlKind	getKind()
Namespace	getNamespace()
Nature	getNature()
java.lang.String	getOtherEndName() Returns name of the other end if this skipped element is a connector, null otherwise.
OwningWg	getOwner()
java.util.Set	getPredefinedTagNames()
java.lang.String	getQualifiedName()
boolean	isConnector() Returns true if this skipped element is some kind of connector, otherwise it's an element.
boolean	isForPackage() Returns true if this skipped item is related to package, false if related to class.
boolean	isInformative()
java.lang.String	toString()

Methods inherited from class org.tanjakostic.jcleancim.model.AbstractUmlObject

addDeprecAndInf, addTaggedValue, appendRemainingCustomStereotypes, classifyPerScope, classifyPerScopePerTag, classifyPerTag, collectDuplicateDescriptions, collectDuplicateNames, collectForScope, collectNames, collectQNames, findAllForName, findWithSameUuidAndLog, getAlias, getDescription, getHtmlDescription, getId, getKind, getName, getNature, getOwner, getQualifiedName, getSince, getStereotype, getTaggedValues, getUnallowedTagNames, getUuid, getVisibility, initFromTags, isDeprecated, isInformative, saveTags, toShortString, toShortString, validateTag

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.model.UmlObject

addTaggedValue, getAlias, getDescription, getHtmlDescription, getId, getKind, getName, getNamespace, getNature, getOwner, getPredefinedTagNames, getQualifiedName, getSince, getStereotype, getTaggedValues, getUnallowedTagNames, getUuid, getVisibility, isDeprecated, isInformative, toShortString

Fields

EA_STATE_MACHINE

public static final java.lang.String **EA_STATE_MACHINE**

Constant value: **StateMachine**

EA_NOTE

public static final java.lang.String **EA_NOTE**

Constant value: **Note**

EA_TEXT

public static final java.lang.String **EA_TEXT**

Constant value: **Text**

EA_BOUNDARY

public static final java.lang.String **EA_BOUNDARY**

Constant value: **Boundary**

EA_STATE

public static final java.lang.String **EA_STATE**

Constant value: **State**

EA_STATE_NODE

public static final java.lang.String **EA_STATE_NODE**

Constant value: **StateNode**

EA_PROCESS

public static final java.lang.String **EA_PROCESS**

Constant value: **Process**

EA_NOTE_LINK

`public static final java.lang.String EA_NOTE_LINK`

Constant value: **NoteLink**

Methods

getContainer

`public UmlStructure getContainer()`

Returns the container of this skipped element, or the source side if this is a connector.

isForPackage

`public boolean isForPackage()`

Returns true if this skipped item is related to package, false if related to class.

isConnector

`public boolean isConnector()`

Returns true if this skipped element is some kind of connector, otherwise it's an element.

getOtherEndName

`public java.lang.String getOtherEndName()`

Returns name of the other end if this skipped element is a connector, null otherwise.

getOwner

`public OwningWg getOwner()`

getNamespace

`public Namespace getNamespace()`

getNature

`public Nature getNature()`

isInformative

`public boolean isInformative()`

(continued from last page)

This default implementation returns true if objects steretypes include UmlStereotype#INFORMATIVE. If there are additional criteria for deriving informative status, ensure to invoke this default implementation first.

getKind

```
public UmlKind getKind()
```

getQualifiedName

```
public java.lang.String getQualifiedName()
```

getPredefinedTagNames

```
public java.util.Set getPredefinedTagNames()
```

Always returns empty set.

toString

```
public java.lang.String toString()
```

org.tanjakostic.jcleancim.model
Class UmlSkipped.Kind

```
java.lang.Object
  +-java.lang.Enum
    +-org.tanjakostic.jcleancim.model.UmlSkipped.Kind
```

All Implemented Interfaces:
UmlKind, java.io.Serializable, java.lang.Comparable

---

public static final class **UmlSkipped.Kind**
extends java.lang.Enum
implements java.lang.Comparable, java.io.Serializable, UmlKind

Kind of EA elements and connectors that may be found in the model, but are just skipped.

Field Summary

public static final	BOUNDARY
public static final	NOTE
public static final	NOTE_LINK
public static final	OTHER
public static final	PROCESS
public static final	STATE
public static final	STATE_MACHINE
public static final	STATE_NODE
public static final	TEXT

Method Summary

static UmlSkipped.Kind	findForValue(java.lang.String value) Returns literal with value if found, OTHER instance otherwise.
java.lang.String	getDesc()
java.lang.String	getLabel()
java.lang.String	getTag()

java.lang.String	getValue()
static UmlSkipped.Kind	valueOf(java.lang.String name)
static UmlSkipped.Kind[]	values()

Methods inherited from class java.lang.Enum
clone, compareTo, equals, finalize, getDeclaringClass, hashCode, name, ordinal, toString, valueOf

Methods inherited from class java.lang.Object
clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface java.lang.Comparable
compareTo

Methods inherited from interface org.tanjakostic.jcleancim.model.UmlKind
getDesc, getLabel, getTag, getValue

Fields

STATE_MACHINE

public static final org.tanjakostic.jcleancim.model.UmlSkipped.Kind **STATE_MACHINE**

NOTE

public static final org.tanjakostic.jcleancim.model.UmlSkipped.Kind **NOTE**

TEXT

public static final org.tanjakostic.jcleancim.model.UmlSkipped.Kind **TEXT**

BOUNDARY

public static final org.tanjakostic.jcleancim.model.UmlSkipped.Kind **BOUNDARY**

STATE

public static final org.tanjakostic.jcleancim.model.UmlSkipped.Kind **STATE**

(continued from last page)

STATE_NODE

```
public static final org.tanjakostic.jcleancim.model.UmlSkipped.Kind STATE_NODE
```

NOTE_LINK

```
public static final org.tanjakostic.jcleancim.model.UmlSkipped.Kind NOTE_LINK
```

PROCESS

```
public static final org.tanjakostic.jcleancim.model.UmlSkipped.Kind PROCESS
```

OTHER

```
public static final org.tanjakostic.jcleancim.model.UmlSkipped.Kind OTHER
```

Methods

values

```
public static UmlSkipped.Kind[] values()
```

valueOf

```
public static UmlSkipped.Kind valueOf(java.lang.String name)
```

getValue

```
public java.lang.String getValue()
```

getLabel

```
public java.lang.String getLabel()
```

getTag

```
public java.lang.String getTag()
```

getDesc

```
public java.lang.String getDesc()
```

findForValue

```
public static UmlSkipped.Kind findForValue(java.lang.String value)
```

Returns literal with `value` if found, `OTHER` instance otherwise.

org.tanjakostic.jcleancim.model
Class UmlSkipped.Data

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.model.UmlSkipped.Data
```

public static class **UmlSkipped.Data**
extends java.lang.Object

Data from the UML model repository specific to `UmlSkipped`.

Constructor Summary

public	`Data(UmlSkipped.Kind kind, boolean isConnector, java.lang.String otherEndName)` Constructor.

Method Summary

static `UmlSkipped.Data`	`empty(boolean isConnector)` Returns empty instance; for connector, sets default kind to `UmlSkipped.Kind.NOTE_LINK`, and for element to `UmlSkipped.Kind.NOTE`.
`UmlSkipped.Kind`	`getKind()`
`java.lang.String`	`getOtherEndName()`
boolean	`isConnector()`

Methods inherited from class `java.lang.Object`
`clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait`

Constructors

Data

```
public Data(UmlSkipped.Kind kind,
            boolean isConnector,
            java.lang.String otherEndName)
```

Constructor.

Parameters:
`kind`
`isConnector`
`otherEndName` - ignored if `isConnector` is false.

Methods

(continued from last page)

empty

```
public static UmlSkipped.Data empty(boolean isConnector)
```

Returns empty instance; for connector, sets default kind to `UmlSkipped.Kind.NOTE_LINK`, and for element to `UmlSkipped.Kind.NOTE`.

isConnector

```
public boolean isConnector()
```

getKind

```
public UmlSkipped.Kind getKind()
```

getOtherEndName

```
public java.lang.String getOtherEndName()
```

org.tanjakostic.jcleancim.model
Class UmlStereotype

```
java.lang.Object
   |
   +-org.tanjakostic.jcleancim.model.UmlStereotype
```

public class **UmlStereotype**
extends java.lang.Object

UML stereotype.

It is essentially a set of comma-separated string values. To avoid the application to do string parsing and adding e.g. "<<" and ">>" around the stereotypes, this simple class does it in one place.

FIXME: Improve initialisation and add modifiers ...

Field Summary

public static final	ABBR Used for enumerations that represent abbreviations. Value: **abbr**
public static final	ADMIN Used for some abstract LN classes (61850-7-4) to tag that presence conditions of its data objects do not change in the context of derived statistic instance. Value: **admin**
public static final	BASIC Value: **basic**
public static final	CIMDATATYPE Value: **CIMDatatype**
public static final	COMPOUND Value: **Compound**
public static final	COND Used for enumerations that represent presence conditions (modelled as class constraints). Value: **cond**
public static final	DEPRECATED UML stereotype for deprecated UML objects of any kind. Value: **deprecated**
public static final	EA_INTERFACE Value: **interface**
public static final	ENUM UML stereotype for enumeration literals. Value: **enum**

public static final	ENUMERATION UML stereotype for enumeration class. Value: **enumeration**
public static final	EVENT Used for operations. Value: **event**
public static final	IMPORT UML stereotype for import dependency between UML packages. Value: **import**
public static final	INFORMATIVE UML stereotype for non-normative UML objects of any kind. Value: **informative**
public static final	ISBASEDON CIM stereotype for UML profiles, applicable to UML dependency. Value: **IsBasedOn**
public static final	OLD_DATATYPE Value: **Datatype**
public static final	PACKED Value: **packed**
public static final	PRIMITIVE Value: **Primitive**
public static final	STATISTICS Used for CDCs (61850-7-3) allowed for use as type in DOs of derived statistics LNs; CDCs without this stereotype are forbidden for use in derived statistics context. Value: **statistics**
public static final	STRUCTURED Value: **structured**

Constructor Summary

public	UmlStereotype(java.lang.String[] tokens) Constructor.

Method Summary

boolean	contains(java.lang.String token) Returns whether this instance contains token).
boolean	containsAnyOf(java.util.Collection tokens) Returns whether this instance contains any token from token).
static java.util.Map	getAssociationBuiltIns()
static java.util.Map	getAssociationEndBuiltIns()

static java.util.Map	getAttributeBuiltIns()
static java.util.Map	getClassBuiltIns()
static java.util.Map	getDependencyBuiltIns()
static java.util.Map	getDiagramBuiltIns()
static java.util.Map	getOperationBuiltIns()
static java.util.Map	getOperationParameterBuiltIns()
static java.util.Map	getPackageBuiltIns()
java.util.Set	getTokensOtherThan(java.util.Set tokens) Returns all the tokens of this instance "minus" those contained in tokens.
boolean	isEmpty() Returns whether this instance is empty (has no tokens).
boolean	memberOf(java.util.Set tokens) Returns whether all the tokens of this instance are contained in tokens.
java.lang.String	toString()
java.lang.String	value() Returns comma-separated list of stereotype tokens.

Methods inherited from class `java.lang.Object`
clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Fields

ENUMERATION

`public static final java.lang.String` **`ENUMERATION`**

UML stereotype for enumeration class.
Constant value: **`enumeration`**

ENUM

`public static final java.lang.String` **`ENUM`**

UML stereotype for enumeration literals.
Constant value: **`enum`**

DEPRECATED

`public static final java.lang.String` **`DEPRECATED`**

(continued from last page)

UML stereotype for deprecated UML objects of any kind.
Constant value: **deprecated**

INFORMATIVE

`public static final java.lang.String` **INFORMATIVE**

UML stereotype for non-normative UML objects of any kind.
Constant value: **informative**

IMPORT

`public static final java.lang.String` **IMPORT**

UML stereotype for import dependency between UML packages.
Constant value: **import**

PRIMITIVE

`public static final java.lang.String` **PRIMITIVE**

Constant value: **Primitive**

OLD_DATATYPE

`public static final java.lang.String` **OLD_DATATYPE**

Constant value: **Datatype**

CIMDATATYPE

`public static final java.lang.String` **CIMDATATYPE**

Constant value: **CIMDatatype**

COMPOUND

`public static final java.lang.String` **COMPOUND**

Constant value: **Compound**

ISBASEDON

`public static final java.lang.String` **ISBASEDON**

CIM stereotype for UML profiles, applicable to UML dependency.
Constant value: **IsBasedOn**

EA_INTERFACE

`public static final java.lang.String` **EA_INTERFACE**

Constant value: **interface**

(continued from last page)

PACKED

`public static final java.lang.String` **`PACKED`**

Constant value: **`packed`**

BASIC

`public static final java.lang.String` **`BASIC`**

Constant value: **`basic`**

STRUCTURED

`public static final java.lang.String` **`STRUCTURED`**

Constant value: **`structured`**

ADMIN

`public static final java.lang.String` **`ADMIN`**

Used for some abstract LN classes (61850-7-4) to tag that presence conditions of its data objects do not change in the context of derived statistic instance. Not inheritable (i.e., should not be printed in subclasses).
Constant value: **`admin`**

STATISTICS

`public static final java.lang.String` **`STATISTICS`**

Used for CDCs (61850-7-3) allowed for use as type in DOs of derived statistics LNs; CDCs without this stereotype are forbidden for use in derived statistics context. It is inheritable (i.e., should be printed in subclasses).
Constant value: **`statistics`**

COND

`public static final java.lang.String` **`COND`**

Used for enumerations that represent presence conditions (modelled as class constraints).
Constant value: **`cond`**

ABBR

`public static final java.lang.String` **`ABBR`**

Used for enumerations that represent abbreviations.
Constant value: **`abbr`**

EVENT

`public static final java.lang.String` **`EVENT`**

Used for operations.
Constant value: **`event`**

Constructors

(continued from last page)

UmlStereotype

```
public UmlStereotype(java.lang.String[] tokens)
```

Constructor.

Parameters:

`tokens` - (optional) desired number of individual non-null stereotype tokens; empty tokens are skipped. A token that contains comma-separated items will be split and each of those items will be kept as a stereotype token.

Methods

getPackageBuiltIns

```
public static java.util.Map getPackageBuiltIns()
```

getClassBuiltIns

```
public static java.util.Map getClassBuiltIns()
```

getAttributeBuiltIns

```
public static java.util.Map getAttributeBuiltIns()
```

getAssociationBuiltIns

```
public static java.util.Map getAssociationBuiltIns()
```

getAssociationEndBuiltIns

```
public static java.util.Map getAssociationEndBuiltIns()
```

getDependencyBuiltIns

```
public static java.util.Map getDependencyBuiltIns()
```

getDiagramBuiltIns

```
public static java.util.Map getDiagramBuiltIns()
```

getOperationBuiltIns

```
public static java.util.Map getOperationBuiltIns()
```

(continued from last page)

getOperationParameterBuiltIns

`public static java.util.Map getOperationParameterBuiltIns()`

isEmpty

`public boolean isEmpty()`

Returns whether this instance is empty (has no tokens).

contains

`public boolean contains(java.lang.String token)`

Returns whether this instance contains `token`).

containsAnyOf

`public boolean containsAnyOf(java.util.Collection tokens)`

Returns whether this instance contains any token from `token`).

memberOf

`public boolean memberOf(java.util.Set tokens)`

Returns whether all the tokens of this instance are contained in `tokens`.

getTokensOtherThan

`public java.util.Set getTokensOtherThan(java.util.Set tokens)`

Returns all the tokens of this instance "minus" those contained in `tokens`.

value

`public java.lang.String value()`

Returns comma-separated list of stereotype tokens.

toString

`public java.lang.String toString()`

Ensloses `value()` into "<<" and ">>".

org.tanjakostic.jcleancim.model
Class UmlStructure

```
java.lang.Object
  |-org.tanjakostic.jcleancim.model.AbstractUmlObject
    |-org.tanjakostic.jcleancim.model.UmlStructure
```

All Implemented Interfaces:
UmlObject

Direct Known Subclasses:
UmlClass, UmlPackage

---

public abstract class **UmlStructure**
extends AbstractUmlObject

Common implementation for collections contained by packages and classes.

Nested Class Summary

class	UmlStructure.Data UmlStructure.Data

Fields inherited from class org.tanjakostic.jcleancim.model.AbstractUmlObject
CLASS_SEPARATOR, NULL_OBJ_NAME, PACKAGE_SEPARATOR

Constructor Summary

protected	UmlStructure(UmlObjectData objData, UmlStructure.Data data) Creates common parts of structures.

Method Summary

UmlDependency	addDependency(UmlStructure target, UmlObjectData objData, UmlDependency.Data data) Creates from arguments an explicit (hand-drawn) UML dependency, adds it to itself as source, to target as target, and to the model, and returns the newly created object.
UmlDiagram	addDiagram(java.io.File pic, UmlObjectData objData, UmlDiagram.Data data) Creates from arguments a diagram, adds it to itself and to the model, and returns the newly created object.
UmlSkipped	addSkippedUmlItem(UmlObjectData objData, UmlSkipped.Data data) Creates from arguments a skipped UML item, adds it to itself, and returns the newly created object.
java.util.Collection	collectDependencyAfferentStructures() Returns all structures that depend on me through an explicit UML dependency in the model.
java.util.Collection	collectDependencyEfferentStructures() Returns all structures that I *directly* depend on through an explicit UML dependency in the model.

java.util.Collection	collectMyAndParentsDependencyEfferentStructures() Returns all structures that I and my containers *recursively* depend on through an explicit UML dependency in the model; starting from my direct dependencies, then following my container's dependencies and so on).
abstract UmlStructure	getContainer() Returns containing structure, null in case this is the model package.
java.util.Set	getDependenciesAsSource() Returns all explicit (hand-drawn) UML dependencies where I am source.
java.util.Set	getDependenciesAsTarget() Returns all explicit (hand-drawn) UML dependencies where I am target.
java.util.Set	getDiagrams() Returns all diagrams in this structure.
abstract UmlKind	getKind()
abstract UmlModel	getModel() Returns the model this structure belongs to.
abstract Nature	getNature()
abstract OwningWg	getOwner()
abstract java.lang.String	getQualifiedName()
java.util.Set	getSkippedUmlItems() Returns all skipped UML items: elements within or connectors with this structure.
boolean	isSelfDependent()

Methods inherited from class org.tanjakostic.jcleancim.model.AbstractUmlObject

addDeprecAndInf, addTaggedValue, appendRemainingCustomStereotypes, classifyPerScope, classifyPerScopePerTag, classifyPerTag, collectDuplicateDescriptions, collectDuplicateNames, collectForScope, collectNames, collectQNames, findAllForName, findWithSameUuidAndLog, getAlias, getDescription, getHtmlDescription, getId, getKind, getName, getNature, getOwner, getQualifiedName, getSince, getStereotype, getTaggedValues, getUnallowedTagNames, getUuid, getVisibility, initFromTags, isDeprecated, isInformative, saveTags, toShortString, toShortString, validateTag

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.model.UmlObject

addTaggedValue, getAlias, getDescription, getHtmlDescription, getId, getKind, getName, getNamespace, getNature, getOwner, getPredefinedTagNames, getQualifiedName, getSince, getStereotype, getTaggedValues, getUnallowedTagNames, getUuid, getVisibility, isDeprecated, isInformative, toShortString

Constructors

UmlStructure

```
protected UmlStructure(UmlObjectData objData,
                       UmlStructure.Data data)
```

Creates common parts of structures. After creating an instance of a concrete subtype, you may want to add skipped items, explicit (hand-drawn) UML dependencies and diagrams.

Parameters:

`data`

Methods

addSkippedUmlItem

```
public final UmlSkipped addSkippedUmlItem(UmlObjectData objData,
         UmlSkipped.Data data)
```

Creates from arguments a skipped UML item, adds it to itself, and returns the newly created object. In case the item with the same UUID has already been added, returns the existing item immediately.

---

getSkippedUmlItems

```
public final java.util.Set getSkippedUmlItems()
```

Returns all skipped UML items: elements within or connectors with this structure.

---

addDependency

```
public final UmlDependency addDependency(UmlStructure target,
         UmlObjectData objData,
         UmlDependency.Data data)
```

Creates from arguments an explicit (hand-drawn) UML dependency, adds it to itself as `source`, to `target` as target, and to the model, and returns the newly created object. In case the dependency with the same UUID has already been added, returns the existing dependency immediately. It is the responsibility of the caller to call this method on the source structure, otherwise the behaviour is undefined.

Parameters:

`target` - must be of the same type as this.

`objData` - common UML data for the new dependency.

`data` - data proper to new dependency.

Throws:

`IllegalArgumentException` - if this and `target` are from different models, or if this and `target` are the same object, or if the types of this and `target` differ.

---

getDependenciesAsSource

```
public final java.util.Set getDependenciesAsSource()
```

Returns all explicit (hand-drawn) UML dependencies where I am source.

---

getDependenciesAsTarget

```
public final java.util.Set getDependenciesAsTarget()
```

(continued from last page)

Returns all explicit (hand-drawn) UML dependencies where I am target.

isSelfDependent

```
public final boolean isSelfDependent()
```

See Also:
UmlStructure.Data.isSelfDependent()

collectDependencyEfferentStructures

```
public final java.util.Collection collectDependencyEfferentStructures()
```

Returns all structures that I *directly* depend on through an explicit UML dependency in the model. For exhaustive list, use collectMyAndParentsDependencyEfferentStructures().

collectMyAndParentsDependencyEfferentStructures

```
public final java.util.Collection collectMyAndParentsDependencyEfferentStructures()
```

Returns all structures that I and my containers *recursively* depend on through an explicit UML dependency in the model; starting from my direct dependencies, then following my container's dependencies and so on). We stop as soon as a cycle is detected.

For simple list of my direct dependencies, use collectDependencyEfferentStructures().

collectDependencyAfferentStructures

```
public final java.util.Collection collectDependencyAfferentStructures()
```

Returns all structures that depend on me through an explicit UML dependency in the model.

addDiagram

```
public final UmlDiagram addDiagram(java.io.File pic,
         UmlObjectData objData,
         UmlDiagram.Data data)
```

Creates from arguments a diagram, adds it to itself and to the model, and returns the newly created object. In case the diagram with the same UUID has already been added, returns the existing diagram immediately.

getDiagrams

```
public final java.util.Set getDiagrams()
```

Returns all diagrams in this structure.

getModel

```
public abstract UmlModel getModel()
```

Returns the model this structure belongs to.

getContainer

```
public abstract UmlStructure getContainer()
```

Returns containing structure, null in case this is the model package.

getOwner

```
public abstract OwningWg getOwner()
```

getNature

```
public abstract Nature getNature()
```

getKind

```
public abstract UmlKind getKind()
```

getQualifiedName

```
public abstract java.lang.String getQualifiedName()
```

org.tanjakostic.jcleancim.model
Class UmlStructure.Data

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.model.UmlStructure.Data
```

Direct Known Subclasses:
Data, Data

public static class **UmlStructure.Data**
extends java.lang.Object

Data from the UML model repository specific to `UmlStructure`.

Constructor Summary

public	`Data(boolean selfDependent)` Constructor.

Method Summary

static `UmlStructure.Data`	`empty()` Returns an empty instance.
boolean	`isSelfDependent()` Returns whether the repository contains an explicit (hand-drawn) UML self-dependency; these are not included in the in-memory model, but only reported through validation.

Methods inherited from class `java.lang.Object`
`clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait`

Constructors

Data

`public Data(boolean selfDependent)`

Constructor.

Methods

empty

`public static UmlStructure.Data empty()`

Returns an empty instance.

(continued from last page)

isSelfDependent

```
public final boolean isSelfDependent()
```

Returns whether the repository contains an explicit (hand-drawn) UML self-dependency; these are not included in the in-memory model, but only reported through validation.

org.tanjakostic.jcleancim.model
Class UmlVisibility

```
java.lang.Object
  +-java.lang.Enum
    +-org.tanjakostic.jcleancim.model.UmlVisibility
```

All Implemented Interfaces:
java.io.Serializable, java.lang.Comparable

public final class **UmlVisibility**
extends java.lang.Enum

"Translates" EA visibility strings to lower case for classes, attributes, operations and association ends.

Field Summary

public static final	PACKAGE
public static final	PRIVATE
public static final	PROTECTED
public static final	PUBLIC

Method Summary

java.lang.String	toString()
static UmlVisibility	valueOf(java.lang.String name)
static UmlVisibility[]	values()

Methods inherited from class `java.lang.Enum`

clone, compareTo, equals, finalize, getDeclaringClass, hashCode, name, ordinal, toString, valueOf

Methods inherited from class `java.lang.Object`

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface `java.lang.Comparable`

compareTo

(continued from last page)

Fields

PRIVATE

```
public static final org.tanjakostic.jcleancim.model.UmlVisibility PRIVATE
```

PACKAGE

```
public static final org.tanjakostic.jcleancim.model.UmlVisibility PACKAGE
```

PROTECTED

```
public static final org.tanjakostic.jcleancim.model.UmlVisibility PROTECTED
```

PUBLIC

```
public static final org.tanjakostic.jcleancim.model.UmlVisibility PUBLIC
```

Methods

values

```
public static UmlVisibility[] values()
```

valueOf

```
public static UmlVisibility valueOf(java.lang.String name)
```

toString

```
public java.lang.String toString()
```

Returns the literal value transformed to lower case.

org.tanjakostic.jcleancim.model
Class ValueRange

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.model.ValueRange
```

public class **ValueRange**
extends java.lang.Object

WG10 CDC and DA attributes specify sometimes allowed ranges in the initial value.

Field Summary

public static final	RANGE_TOKEN Value: **...**

Constructor Summary

public	ValueRange(java.lang.String min, java.lang.String max) Constructor; at least one of two arguments must be non-null, non-empty.
public	ValueRange(java.lang.String initialValue)

Method Summary

static boolean	isValidRangeFormat(java.lang.String initialValue) Returns whether the intial value in UML repository has required format (i.e., whether it includes '#RANGE_TOKEN' with at least one of min and max).
java.lang.String	max() Returns (potentially null) maximum value.
java.lang.String	min() Returns (potentially null) minimum value.
java.lang.String	toString()

Methods inherited from class `java.lang.Object`
clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Fields

RANGE_TOKEN

public static final java.lang.String **RANGE_TOKEN**

Constant value: **...**

Constructors

ValueRange

```
public ValueRange(java.lang.String min,
                  java.lang.String max)
```

Constructor; at least one of two arguments must be non-null, non-empty.

ValueRange

```
public ValueRange(java.lang.String initialValue)
```

Methods

isValidRangeFormat

```
public static boolean isValidRangeFormat(java.lang.String initialValue)
```

Returns whether the intial value in UML repository has required format (i.e., whether it includes '#RANGE_TOKEN' with at least one of min and max).

min

```
public java.lang.String min()
```

Returns (potentially null) minimum value.

max

```
public java.lang.String max()
```

Returns (potentially null) maximum value.

toString

```
public java.lang.String toString()
```

org.tanjakostic.jcleancim.model
Class VersionInfo

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.model.VersionInfo
```

public class **VersionInfo**
extends java.lang.Object

Content of version class, expected to be found in top packages.

Constructor Summary

public	VersionInfo(UmlClass versionClass) Constructor.
public	VersionInfo(java.lang.String version, java.lang.String date)

Method Summary

java.lang.String	getDate()
static java.lang.String	getExpectedVersionClassName(Nature nature, java.lang.String name) Returns the expected name for the version class, as per IEC TC57 UMl models rules.
java.lang.String	getVersion()
java.lang.String	toString()

Methods inherited from class `java.lang.Object`
clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Constructors

VersionInfo

public **VersionInfo**(UmlClass versionClass)

Constructor. Logs error if the retained version class has unexpected attributes.

VersionInfo

public **VersionInfo**(java.lang.String version,
java.lang.String date)

(continued from last page)

Methods

getExpectedVersionClassName

```
public static java.lang.String getExpectedVersionClassName(Nature nature,
         java.lang.String name)
```

Returns the expected name for the version class, as per IEC TC57 UMl models rules.

getVersion

```
public java.lang.String getVersion()
```

getDate

```
public java.lang.String getDate()
```

toString

```
public java.lang.String toString()
```

Package
org.tanjakostic.jcleancim.statistics

Classes responsible for tracking and reporting model statistics.

Main classes are:

- `ModelStats` - restricts the scope as given in properties and calculates statistics (counts) of different kinds of elements in the model.
- `CrossPackageStats` - collects and logs actual dependencies on two levels: among top level packages (i.e. between different package owners), and among packages within the same top level package (i.e., within the same owner).

TODO:

- Design of `CrossPackageStats` is currently very quick&dirty, but at least allows to log dependencies in general. Please, propose how would you like to better structure the output, or how to make it shorter and easier to scan.
- Do we need to save this in some structured format? Which?

org.tanjakostic.jcleancim.statistics
Class Counter

```
java.lang.Object
   |
   +-org.tanjakostic.jcleancim.statistics.Counter
```

public class **Counter**
extends java.lang.Object

Data structure to hold counts of model elements.

Constructor Summary

public	Counter()

Methods inherited from class `java.lang.Object`
`clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait`

Constructors

Counter

public **Counter**()

org.tanjakostic.jcleancim.statistics
Class CrossPackageStats

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.statistics.CrossPackageStats
```

public class **CrossPackageStats**
extends java.lang.Object

Provides methods to get and log links among packages having different owners (WGs), as well as links among packages within the same owner. Actual dependencies are calculated based on inheritance, associations, dependencies (drawn by hand in the model), attribute types, and arguments and exceptions in operations.

Constructor Summary

public	CrossPackageStats(UmlModel model) Constructor.

Method Summary

void	logStats() Logs statistics on links among packages.
java.lang.String	toString()

Methods inherited from class `java.lang.Object`

`clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait`

Constructors

CrossPackageStats

`public` **`CrossPackageStats`**`(UmlModel model)`

Constructor. Calculates and stores all dependencies.

Parameters:
`model`

Methods

logStats

`public void` **`logStats`**`()`

Logs statistics on links among packages.

(continued from last page)

toString

```
public java.lang.String toString()
```

org.tanjakostic.jcleancim.statistics
Class ModelStats

```
java.lang.Object
   |
   +-org.tanjakostic.jcleancim.statistics.ModelStats
```

public class **ModelStats**
extends java.lang.Object

Statistics (counts) of different kinds of elements in the model per nature and per owner.

Constructor Summary

public	ModelStats(UmlModel model) Constructor.

Method Summary

int	getAssociationCount()
int	getAttributeCount()
int	getClassCount()
int	getDependencyCount()
int	getDiagramCount()
UmlModel	getModel()
int	getOperationCount()
int	getPackageCount()
java.util.Map	getScopedAssociations()
java.util.Map	getScopedAttributes()
java.util.Map	getScopedClasses()
java.util.Map	getScopedDependencies()
java.util.Map	getScopedDiagrams()
java.util.Map	getScopedOperations()
java.util.Map	getScopedPackages()

java.util.Map	getScopedTags()
java.util.Map	getStatsPerNature()
int	getTagNamesCount()
void	logAbbreviatedTermUsage(org.apache.log4j.Level level) (IEC61850) For every abbreviated term, logs DOs using it; opposite to logDONameDecomposition(Level).
void	logAggregationsWithWgClasses(org.apache.log4j.Level level, OwningWg wg)
void	logAttributesWithConstraints(org.apache.log4j.Level level) Note: Attribute constraints can be derived from class or own.
void	logCimNoncimAssociations(org.apache.log4j.Level level)
void	logClasses(org.apache.log4j.Level level)
void	logClassesWithAttributeConstraints(org.apache.log4j.Level level)
void	logDONameDecomposition(org.apache.log4j.Level level) (IEC61850) For every DO, logs abbreviations used; opposite to logAbbreviatedTermUsage(Level).
void	logMultivaluedAttributes(org.apache.log4j.Level level)
void	logNamespaceInfos(org.apache.log4j.Level level)
void	logNormativeAssociationsWithWgClasses(org.apache.log4j.Level level, OwningWg wg)
void	logNormativeClasses(org.apache.log4j.Level level, java.util.EnumSet wgs)
void	logOperations(org.apache.log4j.Level level)
void	logPackages(org.apache.log4j.Level level)
void	logStats()
void	logTaggedValues(org.apache.log4j.Level level) Using tagged values as keys, logs detailed list of referencing objects, except for those tags that are configured to be ignored.
void	logVersionInfos(org.apache.log4j.Level level)
java.lang.String	toHtml()
java.lang.String	toString()

Methods inherited from class `java.lang.Object`

```
clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait,
wait
```

Constructors

ModelStats

```
public ModelStats(UmlModel model)
```

Constructor.

Parameters:
`model`

Methods

getScopedPackages

```
public java.util.Map getScopedPackages()
```

getPackageCount

```
public int getPackageCount()
```

getScopedClasses

```
public java.util.Map getScopedClasses()
```

getClassCount

```
public int getClassCount()
```

getScopedAttributes

```
public java.util.Map getScopedAttributes()
```

getAttributeCount

```
public int getAttributeCount()
```

(continued from last page)

getScopedAssociations

```
public java.util.Map getScopedAssociations()
```

getAssociationCount

```
public int getAssociationCount()
```

getScopedOperations

```
public java.util.Map getScopedOperations()
```

getOperationCount

```
public int getOperationCount()
```

getScopedDependencies

```
public java.util.Map getScopedDependencies()
```

getDependencyCount

```
public int getDependencyCount()
```

getScopedDiagrams

```
public java.util.Map getScopedDiagrams()
```

getDiagramCount

```
public int getDiagramCount()
```

getScopedTags

```
public java.util.Map getScopedTags()
```

getTagNamesCount

```
public int getTagNamesCount()
```

(continued from last page)

getStatsPerNature

```
public java.util.Map getStatsPerNature()
```

getModel

```
public UmlModel getModel()
```

logStats

```
public void logStats()
```

toHtml

```
public java.lang.String toHtml()
```

toString

```
public java.lang.String toString()
```

logPackages

```
public void logPackages(org.apache.log4j.Level level)
```

logClasses

```
public void logClasses(org.apache.log4j.Level level)
```

logOperations

```
public void logOperations(org.apache.log4j.Level level)
```

logNormativeClasses

```
public void logNormativeClasses(org.apache.log4j.Level level,
         java.util.EnumSet wgs)
```

(continued from last page)

logNormativeAssociationsWithWgClasses

```
public void logNormativeAssociationsWithWgClasses(org.apache.log4j.Level level,
        OwningWg wg)
```

logAggregationsWithWgClasses

```
public void logAggregationsWithWgClasses(org.apache.log4j.Level level,
        OwningWg wg)
```

logCimNoncimAssociations

```
public void logCimNoncimAssociations(org.apache.log4j.Level level)
```

logClassesWithAttributeConstraints

```
public void logClassesWithAttributeConstraints(org.apache.log4j.Level level)
```

logAttributesWithConstraints

```
public void logAttributesWithConstraints(org.apache.log4j.Level level)
```

Note: Attribute constraints can be derived from class or own.

logMultivaluedAttributes

```
public void logMultivaluedAttributes(org.apache.log4j.Level level)
```

logNamespaceInfos

```
public void logNamespaceInfos(org.apache.log4j.Level level)
```

logVersionInfos

```
public void logVersionInfos(org.apache.log4j.Level level)
```

logTaggedValues

```
public void logTaggedValues(org.apache.log4j.Level level)
```

Using tagged values as keys, logs detailed list of referencing objects, except for those tags that are configured to be ignored.

(continued from last page)

logDONameDecomposition

`public void **logDONameDecomposition**(org.apache.log4j.Level level)`

(IEC61850) For every DO, logs abbreviations used; opposite to `logAbbreviatedTermUsage(Level)`.

logAbbreviatedTermUsage

`public void **logAbbreviatedTermUsage**(org.apache.log4j.Level level)`

(IEC61850) For every abbreviated term, logs DOs using it; opposite to `logDONameDecomposition(Level)`.

org.tanjakostic.jcleancim.statistics
Class StatsPerOwner

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.statistics.StatsPerOwner
```

public class **StatsPerOwner**
extends java.lang.Object

Statistics per owner for model of any Nature.

Constructor Summary

public	StatsPerOwner(UmlModel model, OwningWg owner, Nature nature, Counter currentScopeCounter, Counter totalModelCounter)

Method Summary

int	getAssociationCount()
java.util.Map	getAssociations()
int	getAttributeCount()
java.util.Map	getAttributes()
int	getClassCount()
java.util.Map	getClasses()
java.util.Map	getDependencies()
int	getDependencyCount()
int	getDiagramCount()
java.util.Map	getDiagrams()
Nature	getNature()
int	getOperationCount()
java.util.Map	getOperations()
OwningWg	getOwner()
int	getPackageCount()

java.util.Map	getPackages()
int	getTagNameCount()
java.util.Map	getTags()
java.lang.String	toHtml()
java.lang.String	toString()

Methods inherited from class java.lang.Object
clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Constructors

StatsPerOwner

```
public StatsPerOwner(UmlModel model,
                     OwningWg owner,
                     Nature nature,
                     Counter currentScopeCounter,
                     Counter totalModelCounter)
```

Methods

getOwner

```
public final OwningWg getOwner()
```

getNature

```
public Nature getNature()
```

getPackages

```
public java.util.Map getPackages()
```

getClasses

```
public java.util.Map getClasses()
```

(continued from last page)

getAttributes

```
public java.util.Map getAttributes()
```

getAssociations

```
public java.util.Map getAssociations()
```

getOperations

```
public java.util.Map getOperations()
```

getDependencies

```
public java.util.Map getDependencies()
```

getDiagrams

```
public java.util.Map getDiagrams()
```

getTags

```
public java.util.Map getTags()
```

getPackageCount

```
public int getPackageCount()
```

getClassCount

```
public int getClassCount()
```

getAttributeCount

```
public int getAttributeCount()
```

getAssociationCount

```
public int getAssociationCount()
```

(continued from last page)

getOperationCount

```
public int getOperationCount()
```

getDependencyCount

```
public int getDependencyCount()
```

getDiagramCount

```
public int getDiagramCount()
```

getTagNameCount

```
public int getTagNameCount()
```

toHtml

```
public java.lang.String toHtml()
```

toString

```
public java.lang.String toString()
```

Package
org.tanjakostic.jcleancim.util

org.tanjakostic.jcleancim.util
Class ApplicationException

```
java.lang.Object
  +-java.lang.Throwable
      +-java.lang.Exception
          +-org.tanjakostic.jcleancim.util.ApplicationException
```

All Implemented Interfaces:
java.io.Serializable

Direct Known Subclasses:
UnsupportedInputFormatException, UnsupportedOutputFormatException

---

public class **ApplicationException**
extends java.lang.Exception

Application exception.

Implementation note: Wrap other exceptions into this one. If you need more specific exceptions, subclass this one.

Constructor Summary

public	ApplicationException()
public	ApplicationException(java.lang.String message, java.lang.Throwable cause)
public	ApplicationException(java.lang.String message)
public	ApplicationException(java.lang.Throwable cause)

Methods inherited from class `java.lang.Throwable`
addSuppressed, fillInStackTrace, getCause, getLocalizedMessage, getMessage, getStackTrace, getSuppressed, initCause, printStackTrace, printStackTrace, printStackTrace, setStackTrace, toString

Methods inherited from class `java.lang.Object`
clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

---

Constructors

ApplicationException

public **ApplicationException**()

ApplicationException

```
public ApplicationException(java.lang.String message,
                            java.lang.Throwable cause)
```

ApplicationException

```
public ApplicationException(java.lang.String message)
```

ApplicationException

```
public ApplicationException(java.lang.Throwable cause)
```

org.tanjakostic.jcleancim.util
Class BMPFile

```
java.lang.Object
  |-java.awt.Component
     |-org.tanjakostic.jcleancim.util.BMPFile
```

All Implemented Interfaces:
java.io.Serializable, java.awt.MenuContainer, java.awt.image.ImageObserver

---

public class **BMPFile**
extends java.awt.Component

Class copied (with pride :-) and adapted from: Java World.

Fields inherited from class `java.awt.Component`
`accessibleContext`, `BOTTOM_ALIGNMENT`, `CENTER_ALIGNMENT`, `LEFT_ALIGNMENT`, `RIGHT_ALIGNMENT`, `TOP_ALIGNMENT`

Fields inherited from interface `java.awt.image.ImageObserver`
`ABORT`, `ALLBITS`, `ERROR`, `FRAMEBITS`, `HEIGHT`, `PROPERTIES`, `SOMEBITS`, `WIDTH`

Constructor Summary

public	`BMPFile()`

Method Summary

void	`saveBitmap(java.lang.String filename, java.awt.Image image, int width, int height)`

Methods inherited from class `java.awt.Component`

action, add, addComponentListener, addFocusListener, addHierarchyBoundsListener, addHierarchyListener, addInputMethodListener, addKeyListener, addMouseListener, addMouseMotionListener, addMouseWheelListener, addNotify, addPropertyChangeListener, addPropertyChangeListener, applyComponentOrientation, areFocusTraversalKeysSet, bounds, checkImage, checkImage, coalesceEvents, contains, contains, createImage, createImage, createVolatileImage, createVolatileImage, deliverEvent, disable, disableEvents, dispatchEvent, doLayout, enable, enable, enableEvents, enableInputMethods, firePropertyChange, firePropertyChange, firePropertyChange, firePropertyChange, firePropertyChange, firePropertyChange, firePropertyChange, firePropertyChange, firePropertyChange, getAccessibleContext, getAlignmentX, getAlignmentY, getBackground, getBaseline, getBaselineResizeBehavior, getBounds, getBounds, getColorModel, getComponentAt, getComponentAt, getComponentListeners, getComponentOrientation, getCursor, getDropTarget, getFocusCycleRootAncestor, getFocusListeners, getFocusTraversalKeys, getFocusTraversalKeysEnabled, getFont, getFontMetrics, getForeground, getGraphics, getGraphicsConfiguration, getHeight, getHierarchyBoundsListeners, getHierarchyListeners, getIgnoreRepaint, getInputContext, getInputMethodListeners, getInputMethodRequests, getKeyListeners, getListeners, getLocale, getLocation, getLocation, getLocationOnScreen, getMaximumSize, getMinimumSize, getMouseListeners, getMouseMotionListeners, getMousePosition, getMouseWheelListeners, getName, getParent, getPeer, getPreferredSize, getPropertyChangeListeners, getPropertyChangeListeners, getSize, getSize, getToolkit, getTreeLock, getWidth, getX, getY, gotFocus, handleEvent, hasFocus, hide, imageUpdate, inside, invalidate, isBackgroundSet, isCursorSet, isDisplayable, isDoubleBuffered, isEnabled, isFocusable, isFocusCycleRoot, isFocusOwner, isFocusTraversable, isFontSet, isForegroundSet, isLightweight, isMaximumSizeSet, isMinimumSizeSet, isOpaque, isPreferredSizeSet, isShowing, isValid, isVisible, keyDown, keyUp, layout, list, list, list, list, list, locate, location, lostFocus, minimumSize, mouseDown, mouseDrag, mouseEnter, mouseExit, mouseMove, mouseUp, move, nextFocus, paint, paintAll, paramString, postEvent, preferredSize, prepareImage, prepareImage, print, printAll, processComponentEvent, processEvent, processFocusEvent, processHierarchyBoundsEvent, processHierarchyEvent, processInputMethodEvent, processKeyEvent, processMouseEvent, processMouseMotionEvent, processMouseWheelEvent, remove, removeComponentListener, removeFocusListener, removeHierarchyBoundsListener, removeHierarchyListener, removeInputMethodListener, removeKeyListener, removeMouseListener, removeMouseMotionListener, removeMouseWheelListener, removeNotify, removePropertyChangeListener, removePropertyChangeListener, repaint, repaint, repaint, repaint, requestFocus, requestFocus, requestFocusInWindow, requestFocusInWindow, reshape, resize, resize, revalidate, setBackground, setBounds, setBounds, setComponentOrientation, setCursor, setDropTarget, setEnabled, setFocusable, setFocusTraversalKeys, setFocusTraversalKeysEnabled, setFont, setForeground, setIgnoreRepaint, setLocale, setLocation, setLocation, setMaximumSize, setMinimumSize, setName, setPreferredSize, setSize, setSize, setVisible, show, show, size, toString, transferFocus, transferFocusBackward, transferFocusUpCycle, update, validate

Methods inherited from class `java.lang.Object`

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface `java.awt.image.ImageObserver`

imageUpdate

Methods inherited from interface `java.awt.MenuContainer`

getFont, postEvent, remove

Constructors

BMPFile

```
public BMPFile()
```

Methods

saveBitmap

```
public void saveBitmap(java.lang.String filename,
         java.awt.Image image,
         int width,
         int height)
  throws java.io.IOException
```

org.tanjakostic.jcleancim.util
Class HTMLUtil

```
java.lang.Object
   |
   +-org.tanjakostic.jcleancim.util.HTMLUtil
```

public class **HTMLUtil**
extends java.lang.Object

Field Summary

public static final	B_END Value: ****
public static final	B_START Value: ****
public static final	FONT_COLOUR_END Value: ****
public static final	FONT_COLOUR_START Value: **<font**
public static final	HTML_DOC_FMT Format string, to enclose the content of the body element (HTML snippet) into a valid HTML document. Value: **<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN"><html><head></head><body>%s</body></html>**
public static final	I_END Value: ****
public static final	I_START Value: ****
public static final	LI_END Value: ** **
public static final	LI_START Value: ** **
public static final	LIST_TAGS
public static final	NBSP Value: ** **

public static final	OL_END Value: ** **
public static final	OL_START Value: ** **
public static final	P_END Value: ** **
public static final	P_START Value: ** **
public static final	SUB_END Value: **</sub>**
public static final	SUB_START Value: **<sub>**
public static final	SUP_END Value: **</sup>**
public static final	SUP_START Value: **<sup>**
public static final	U_END Value: **</u>**
public static final	U_START Value: **<u>**
public static final	UL_END Value: ** **
public static final	UL_START Value: ** **

Methods inherited from class `java.lang.Object`
`clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait`

Fields

HTML_DOC_FMT

`public static final java.lang.String` **`HTML_DOC_FMT`**

Format string, to enclose the content of the body element (HTML snippet) into a valid HTML document.

(continued from last page)

Constant value: **<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN"><html><head></head><body>%s</body></html>**

UL_START

`public static final java.lang.String` **UL_START**

Constant value: **<ul>**

UL_END

`public static final java.lang.String` **UL_END**

Constant value: **
**

OL_START

`public static final java.lang.String` **OL_START**

Constant value: **<ol>**

OL_END

`public static final java.lang.String` **OL_END**

Constant value: **
**

LI_START

`public static final java.lang.String` **LI_START**

Constant value: **<li>**

LI_END

`public static final java.lang.String` **LI_END**

Constant value: **
**

LIST_TAGS

`public static final java.util.Set` **LIST_TAGS**

B_START

`public static final java.lang.String` **B_START**

Constant value: **<b>**

(continued from last page)

B_END

`public static final java.lang.String` **`B_END`**

Constant value: **`</b>`**

I_START

`public static final java.lang.String` **`I_START`**

Constant value: **`<i>`**

I_END

`public static final java.lang.String` **`I_END`**

Constant value: **`</i>`**

U_START

`public static final java.lang.String` **`U_START`**

Constant value: **`<u>`**

U_END

`public static final java.lang.String` **`U_END`**

Constant value: **`</u>`**

FONT_COLOUR_START

`public static final java.lang.String` **`FONT_COLOUR_START`**

Constant value: **`<font`**

FONT_COLOUR_END

`public static final java.lang.String` **`FONT_COLOUR_END`**

Constant value: **`</font>`**

SUP_START

`public static final java.lang.String` **`SUP_START`**

Constant value: **`<sup>`**

SUP_END

`public static final java.lang.String` **`SUP_END`**

(continued from last page)

Constant value: **</sup>**

SUB_START

`public static final java.lang.String` **SUB_START**

Constant value: **<sub>**

SUB_END

`public static final java.lang.String` **SUB_END**

Constant value: **</sub>**

P_START

`public static final java.lang.String` **P_START**

Constant value: ** **

P_END

`public static final java.lang.String` **P_END**

Constant value: ** **

NBSP

`public static final java.lang.String` **NBSP**

Constant value: ** **

org.tanjakostic.jcleancim.util
Class MapOfCollections

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.util.MapOfCollections
```

Direct Known Subclasses:
MapOfLists, MapOfSets

public abstract class **MapOfCollections**
extends java.lang.Object

Parameters:
`K` - type for key, `V` - type for collection elements

Constructor Summary

public	MapOfCollections()

Method Summary

void	addValue(java.lang.Object key, java.lang.Object value, java.lang.Object[] furtherValues)
int	calcValueSize() Returns cumulative size of all the values for all the keys in this map; note that this is caclulated at every invocation.
boolean	containsKey(java.lang.Object key)
abstract java.util.Collection	createCollection()
boolean	isEmpty()
java.util.Set	keys()
int	size() Returns size of this map (i.e., number of keys).
java.util.Collection	subCollection(java.lang.Object key) Returns sub-value for `key` if found, empty collection otherwise.
java.lang.String	toString()
java.util.List	toStringLines()
abstract java.lang.Object	value(java.lang.Object key, int idx) "Descends" `key` - `idx`; returns null if no such value.

Methods inherited from class `java.lang.Object`

```
clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait,
wait
```

Constructors

MapOfCollections

```
public MapOfCollections()
```

Methods

addValue

```
public final void addValue(java.lang.Object key,
         java.lang.Object value,
         java.lang.Object[] furtherValues)
```

subCollection

```
public java.util.Collection subCollection(java.lang.Object key)
```

Returns sub-value for `key` if found, empty collection otherwise.

value

```
public abstract java.lang.Object value(java.lang.Object key,
         int idx)
  throws java.lang.UnsupportedOperationException
```

"Descends" `key` - `idx`; returns null if no such value.

createCollection

```
protected abstract java.util.Collection createCollection()
```

size

```
public int size()
```

Returns size of this map (i.e., number of keys).

calcValueSize

```
public int calcValueSize()
```

Returns cumulative size of all the values for all the keys in this map; note that this is caclulated at every invocation.

(continued from last page)

keys

```
public java.util.Set keys()
```

isEmpty

```
public boolean isEmpty()
```

containsKey

```
public boolean containsKey(java.lang.Object key)
```

toStringLines

```
public java.util.List toStringLines()
```

toString

```
public java.lang.String toString()
```

org.tanjakostic.jcleancim.util
Class MapOfLists

```
java.lang.Object
  |-org.tanjakostic.jcleancim.util.MapOfCollections
    |-org.tanjakostic.jcleancim.util.MapOfLists
```

public class **MapOfLists**
extends MapOfCollections

Parameters:
K - type for key, V - type for list elements

Constructor Summary

public	MapOfLists()

Method Summary

java.util.List	createCollection()
java.lang.Object	value(java.lang.Object key, int idx)

Methods inherited from class org.tanjakostic.jcleancim.util.MapOfCollections

addValue, calcValueSize, containsKey, createCollection, isEmpty, keys, size, subCollection, toString, toStringLines, value

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Constructors

MapOfLists

public **MapOfLists**()

Methods

createCollection

protected java.util.List **createCollection**()

value

```
public java.lang.Object value(java.lang.Object key,
         int idx)
```

"Descends" `key` - `idx`; returns null if no such value.

org.tanjakostic.jcleancim.util
Class MapOfMaps

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.util.MapOfMaps
```

public class **MapOfMaps**
extends java.lang.Object

Data structure to hold two levels of keys.
Parameters:
`K` - type for key, `SK` - type for sub-key, `SV` - type for value elements

Constructor Summary

public	`MapOfMaps()`

Method Summary

int	`calcValueSize()` Returns cumulative size of all the values for all the keys in this map; note that this is caclulated at every invocation.
boolean	`containsKey(java.lang.Object key)`
java.util.Map	`createSubMap()`
boolean	`isEmpty()`
java.util.Set	`keys()`
void	`putValue(java.lang.Object key, java.lang.Object subkey, java.lang.Object value)`
int	`size()` Returns size of this map (i.e., number of keys).
java.util.Map	`subMap(java.lang.Object key)` Returns sub-values for `key` if found, empty map otherwise.
java.lang.String	`toString()`
java.util.List	`toStringLines()`
java.lang.Object	`value(java.lang.Object key, java.lang.Object subkey)` "Descends" `key` - `subkey`; returns null if no such value.

Methods inherited from class `java.lang.Object`

`clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait`

Constructors

MapOfMaps

```
public MapOfMaps()
```

Methods

putValue

```
public void putValue(java.lang.Object key,
         java.lang.Object subkey,
         java.lang.Object value)
```

subMap

```
public java.util.Map subMap(java.lang.Object key)
```

Returns sub-values for `key` if found, empty map otherwise.

value

```
public java.lang.Object value(java.lang.Object key,
         java.lang.Object subkey)
```

"Descends" `key` - `subkey`; returns null if no such value.

createSubMap

```
protected java.util.Map createSubMap()
```

size

```
public int size()
```

Returns size of this map (i.e., number of keys).

calcValueSize

```
public int calcValueSize()
```

Returns cumulative size of all the values for all the keys in this map; note that this is caclulated at every invocation.

keys

```
public java.util.Set keys()
```

(continued from last page)

isEmpty

```
public boolean isEmpty()
```

containsKey

```
public boolean containsKey(java.lang.Object key)
```

toStringLines

```
public java.util.List toStringLines()
```

toString

```
public java.lang.String toString()
```

org.tanjakostic.jcleancim.util
Class MapOfSets

```
java.lang.Object
  +-org.tanjakostic.jcleancim.util.MapOfCollections
    +-org.tanjakostic.jcleancim.util.MapOfSets
```

public class **MapOfSets**
extends MapOfCollections

Parameters:
K - type for key, V - type for set elements

Constructor Summary

public	MapOfSets()

Method Summary

java.util.Set	createCollection()
java.lang.Object	value(java.lang.Object key, int idx)

Methods inherited from class org.tanjakostic.jcleancim.util.MapOfCollections

addValue, calcValueSize, containsKey, createCollection, isEmpty, keys, size, subCollection, toString, toStringLines, value

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Constructors

MapOfSets

public **MapOfSets**()

Methods

createCollection

protected java.util.Set **createCollection**()

value

```
public java.lang.Object value(java.lang.Object key,
         int idx)
```

"Descends" `key` - `idx`; returns null if no such value.

org.tanjakostic.jcleancim.util
Class ProgrammerErrorException

```
java.lang.Object
  |-java.lang.Throwable
    |-java.lang.Exception
      |-java.lang.RuntimeException
        |-org.tanjakostic.jcleancim.util.ProgrammerErrorException
```

All Implemented Interfaces:
java.io.Serializable

---

public class **ProgrammerErrorException**
extends java.lang.RuntimeException

Simple wrapper for RT exception, used instead of asserts to indicate where the implementation does not respect some contract.

Constructor Summary

public	ProgrammerErrorException(java.lang.String message)
public	ProgrammerErrorException(java.lang.Throwable cause)
public	ProgrammerErrorException(java.lang.String message, java.lang.Throwable cause)

Methods inherited from class `java.lang.Throwable`
addSuppressed, fillInStackTrace, getCause, getLocalizedMessage, getMessage, getStackTrace, getSuppressed, initCause, printStackTrace, printStackTrace, printStackTrace, setStackTrace, toString

Methods inherited from class `java.lang.Object`
clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

---

Constructors

ProgrammerErrorException

public **ProgrammerErrorException**(java.lang.String message)

ProgrammerErrorException

public **ProgrammerErrorException**(java.lang.Throwable cause)

(continued from last page)

ProgrammerErrorException

```
public ProgrammerErrorException(java.lang.String message,
                                java.lang.Throwable cause)
```

org.tanjakostic.jcleancim.util
Class ProgressBar

```
java.lang.Object
   |
   +-org.tanjakostic.jcleancim.util.ProgressBar
```

public class **ProgressBar**
extends java.lang.Object

Class producing string for the console that prints always on the same line _progress of calculation done in a loop, which the user has to wait for to complete.

Adapted from c00kiemon5ter.

Constructor Summary

public	`ProgressBar(int total, int updatePercentStep)` Constructor.

Method Summary

void	`update(int doneInput)` Call this whenever progress bar needs to be updated.

Methods inherited from class `java.lang.Object`
`clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait`

Constructors

ProgressBar

```
public ProgressBar(int total,
                   int updatePercentStep)
```

Constructor.

Parameters:
`total` - the measure of the total work

Methods

update

```
public void update(int doneInput)
```

Call this whenever progress bar needs to be updated.

Parameters:
`doneInput` - the measure of the work done so far

org.tanjakostic.jcleancim.util
Class ResourceNotOnClasspathException

```
java.lang.Object
  +-java.lang.Throwable
      +-java.lang.Exception
          +-java.lang.RuntimeException
              +-org.tanjakostic.jcleancim.util.ResourceNotOnClasspathException
```

All Implemented Interfaces:
java.io.Serializable

---

public class **ResourceNotOnClasspathException**
extends java.lang.RuntimeException

Field Summary

public static final	CLASSPATH

Constructor Summary

public	ResourceNotOnClasspathException(java.lang.String filePath)
public	ResourceNotOnClasspathException(java.lang.String filePath, java.lang.Throwable cause)
public	ResourceNotOnClasspathException(java.lang.Throwable cause)

Methods inherited from class `java.lang.Throwable`

addSuppressed, fillInStackTrace, getCause, getLocalizedMessage, getMessage, getStackTrace, getSuppressed, initCause, printStackTrace, printStackTrace, printStackTrace, setStackTrace, toString

Methods inherited from class `java.lang.Object`

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

---

Fields

CLASSPATH

public static final java.lang.String **CLASSPATH**

(continued from last page)

Constructors

ResourceNotOnClasspathException

```
public ResourceNotOnClasspathException(java.lang.String filePath)
```

ResourceNotOnClasspathException

```
public ResourceNotOnClasspathException(java.lang.String filePath,
                                       java.lang.Throwable cause)
```

ResourceNotOnClasspathException

```
public ResourceNotOnClasspathException(java.lang.Throwable cause)
```

org.tanjakostic.jcleancim.util
Class Util

```
java.lang.Object
   |
   +-org.tanjakostic.jcleancim.util.Util
```

public class **Util**
extends java.lang.Object

Utility methods.

Nested Class Summary

class	Util.ImageFormat Util.ImageFormat

Field Summary

public static final	EN_DASH IEC editors were replacing the regular dash "-" in captions with EN DASH "–". Value: **8211**
public static final	FILE_SEP
public static final	INDENT_COUNT Value: **2**
public static final	NL
public static final	NON_BREAKING_WHITE_SPACE This is what in MS Word looks like degree celsius...: ' ' Value: **160**
public static final	PATH_SEP
public static final	TOKEN_DELIMITTER Character used to "enclose" a token that is to be concatenated with a separator. Value: **34**
public static final	TRUNCATE_GREATER_THAN Value: **30**
public static final	USER_DIR
public static final	USER_DIR_KEY Value: **user.dir**
public static final	ZERO

Method Summary

static java.lang.String	capitalise(java.lang.String input) Returns string starting with upper-case letter, all the other letters lower-case.
static void	clearClipboard() Clears system clipboard.
static java.lang.String	concatCharSeparatedTokens(java.lang.String separator, java.util.List tokens) Invokes concatStringSeparatedTokens(String, boolean, List) for the case you don't expect separator be embedded into tokens.
static java.lang.String	concatStringSeparatedTokens(java.lang.String separator, boolean delimitTokens, java.util.List tokens) Concatenates tokens with the separator string between consecutive ones, but *not* at the end, and returns the resulting string.
static void	copy(java.io.File src, java.io.File dst) Copies src file to dst file.
static void	copyHtmlToClipboard(java.lang.String htmlBody) Surrounds the non-empty, non-null htmlBody into doctype and html tags to produce a valid HTML document; no-op otherwise.
static void	copyImageToClipboard(java.io.File pic) Copies image in pic to clipboard.
static void	copyTextToClipboard(java.lang.String txt) Copies non-empty, non-null txt to clipboard, no-op otherwise.
static java.util.Map	createKeyValuePair(java.lang.Object key, java.lang.Object value) Retruns the map with a single key/value pair.
static java.io.File	createTempImageFile(java.lang.String dirAbsPath, java.lang.String fileName, Util.ImageFormat format, boolean deleteOnExit) Creates file in the given directory (or in default OS tmp directory) and returns the result.
static void	delete(java.io.File f) Wrapper for the File.delete() that accepts null argument and returns nothing.
static void	ensureContainsNoNull(java.util.Collection arg, java.lang.String name)
static void	ensureContainsNoNull(java.lang.Object[] arg, java.lang.String name)
static void	ensureNotEmpty(boolean[] arg, java.lang.String name)
static void	ensureNotEmpty(java.util.Collection arg, java.lang.String name)
static void	ensureNotEmpty(int[] arg, java.lang.String name)
static void	ensureNotEmpty(java.util.Map arg, java.lang.String name)
static void	ensureNotEmpty(java.lang.Object[] arg, java.lang.String name)
static void	ensureNotEmpty(java.lang.String arg, java.lang.String name)

static void	ensureNotNull(java.lang.Object arg, java.lang.String name)
static java.lang.String	fetchTextFromClipboard() Returns text contained in the clipboard (text could be plain or markup), null if clipboard is empty.
static java.lang.String	fillString(int count, char ch) Returns the string filled with number count of characters c.
static java.io.InputStream	findResourceOnClasspath(java.lang.String resourceName) Returns resource as input stream for its name, given that it is found on the classpath.
static java.lang.String	formatDuration(long millis)
static java.io.File	getDirectory(java.lang.String dirRelPath, boolean createIfMissing) Returns file representing directory dirName under #USER_DIR_KEY.
static java.lang.String	getFileExtension(java.lang.String filePath) Returns extension (after the last ".") if being part of filePath, null otherwise.
static java.lang.String	getFileExtensionWithDot(java.lang.String filePath) Returns extension with the "." if being part of filePath, null otherwise.
static java.lang.String	getIndentSpaces(int count) Returns string of spaces of the size equal to count * INDENT_COUNT.
static java.lang.Object	getKeyByValue(java.util.Map map, java.lang.Object value) FIXME: tests
static java.util.Set	getKeysByValue(java.util.Map map, java.lang.Object value) FIXME: tests
static java.lang.String	getNonBreakingSpaces(int count) Returns string of count non-breaking spaces.
static java.io.File	getOutputFileRenameIfExists(java.lang.String outDirName, java.lang.String outFileName) Returns file #USER_DIR_KEY/outDirName/outFileName .
static java.lang.String	getResourceAbsPath(java.lang.String resourceName, java.lang.String detail) Returns absolute path of the resource found on the classpath.
static boolean	hasContent(java.lang.String value) Returns true is value is not null, and the trimmed content is not empty.
static java.util.Properties	initPropsFromFile(java.lang.String propsFileName) Returns properties loaded from file propsFileName expected to be on the classpath, empty properties if the file has not been found.
static java.util.List	listFiles(java.io.File directory, java.io.FilenameFilter filter, boolean recurse) Returns potentially empty list of files under directory (and its sub-directories if recurse is true), filtered with filter.
static void	logCollection(org.apache.log4j.Level level, java.util.Collection objects, java.lang.String what) Logs each element in objects.

static void	logCompletion(org.apache.log4j.Level level, java.lang.String text, long startMillis, boolean skipTime) Logs text with level (and if skipTime=false, duration since startMillis).
static void	logMap(org.apache.log4j.Level level, java.util.Map objects, java.lang.String what) Logs each element in objects.
static void	logSubtitle(org.apache.log4j.Level level, java.lang.String subtitle) Logs subtitle with level (for sub-steps in the application).
static void	logTitle(org.apache.log4j.Level level, java.lang.String[] title) Logs title with level (for major steps in the application).
static boolean	looksLikePlural(java.lang.String token) Returns whether token looks like plural; returns false for null or empty arg.
static java.lang.String	null2empty(java.lang.String s) Returns empty string if s is null, s otherwise.
static java.lang.Integer	parseInt(java.lang.String intStr) Returns integer from intStr if it isn't null and isn't empty after trimming; otherwise returns null.
static java.lang.Integer	parseIntZero(java.lang.String intStr) Returns integer from intStr if it isn't null and isn't empty after trimming; otherwise returns integer with value 0.
static void	saveImageFromClipboard(java.io.File pic)
static java.io.File	saveToFile(java.lang.String filePath, java.lang.String content) Saves content to filePath and logs the confirmation with level and return the file.
static java.util.Map	sortByDecreasingLength(java.util.Map items) Returns copy of items sorted by decreasing length of keys (longest first).
static java.lang.String[]	sortByDecreasingLength(java.lang.String[] items) Returns copy of items sorted by decreasing length (longest first).
static java.util.List	splitCharSeparatedTokens(java.lang.String input, char c) Splits c-separated string into a list of non-empty tokens.
static java.util.List	splitCommaSeparatedTokens(java.lang.String input) Splits comma-separated string into a list of non-empty tokens.
static java.util.List	splitDirAndFileNames(java.lang.String basePath, java.lang.String relPath) Returns (potentially empty) list of split members of relPath, starting immediately after the basePath.
static java.util.List	splitLines(java.lang.String input, boolean compact) Uses buffered and string reader to identify lines in input and adds them to the result to return.
static java.util.List	splitStringSeparatedTokens(java.lang.String input, java.lang.String separator) Splits c-separated string into a list of non-empty tokens.

static java.lang.String	toCsvRecord(char separatorChar, boolean delimitTokens, java.util.List tokens) Invokes concatStringSeparatedTokens(String, boolean, List) and appends the separatorChar at the end of result, to return the string that is one record in a .csv format *without* new line character at the end.
static java.lang.String	truncateEnd(java.lang.String input) Identical to truncateEnd(String, int) with default value for charCount = #TRUNCATE_GREATER_THAN.
static java.lang.String	truncateEnd(java.lang.String input, int charCount) Truncates input to first charCount characters and appends "...".
static java.lang.String	truncateStart(java.lang.String input) Identical to truncateEnd(String, int) with default value for charCount = #TRUNCATE_GREATER_THAN.
static java.lang.String	truncateStart(java.lang.String input, int charCount) Truncates input to last charCount characters and prepends "...".

Methods inherited from class `java.lang.Object`
clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Fields

INDENT_COUNT

public static final int **INDENT_COUNT**

Constant value: **2**

NL

public static final java.lang.String **NL**

FILE_SEP

public static final java.lang.String **FILE_SEP**

PATH_SEP

public static final java.lang.String **PATH_SEP**

USER_DIR_KEY

public static final java.lang.String **USER_DIR_KEY**

(continued from last page)

Constant value: **`user.dir`**

USER_DIR

`public static final java.lang.String` **`USER_DIR`**

TRUNCATE_GREATER_THAN

`public static final int` **`TRUNCATE_GREATER_THAN`**

Constant value: **`30`**

TOKEN_DELIMITTER

`public static final char` **`TOKEN_DELIMITTER`**

Character used to "enclose" a token that is to be concatenated with a separator.
Constant value: **`34`**

EN_DASH

`public static final char` **`EN_DASH`**

IEC editors were replacing the regular dash "-" in captions with EN DASH "–".
Constant value: **`8211`**

NON_BREAKING_WHITE_SPACE

`public static final char` **`NON_BREAKING_WHITE_SPACE`**

This is what in MS Word looks like degree celsius...: ' '
Constant value: **`160`**

ZERO

`public static final java.lang.Integer` **`ZERO`**

Methods

splitCommaSeparatedTokens

```
public static java.util.List splitCommaSeparatedTokens(java.lang.String input)
```

Splits comma-separated string into a list of non-empty tokens. If `input` is null or empty, returns empty collection.

splitCharSeparatedTokens

```
public static java.util.List splitCharSeparatedTokens(java.lang.String input,
         char c)
```

Splits `c`-separated string into a list of non-empty tokens. If `input` is null or empty, returns empty collection.

If you work with XML text and want to split text content to lines, use `splitLines(String, boolean)` instead.

splitLines

```
public static java.util.List splitLines(java.lang.String input,
         boolean compact)
```

Uses buffered and string reader to identify lines in `input` and adds them to the result to return. If `compact` is true, every parsed line is trimmed and in case it is empty after trimming, that line is not added to the result.

If you work with XML text, use this method (rather than explicit `splitCharSeparatedTokens(String, char)` or `splitStringSeparatedTokens(String, String)`).

Parameters:

`input`

`compact` - whether to compact result

Returns:

`input` split to individual lines.

splitStringSeparatedTokens

```
public static java.util.List splitStringSeparatedTokens(java.lang.String input,
         java.lang.String separator)
```

Splits c-separated string into a list of non-empty tokens. If `input` is null or empty, returns empty collection.

If you work with XML text and want to split text content to lines, use `splitLines(String, boolean)` instead.

concatCharSeparatedTokens

```
public static java.lang.String concatCharSeparatedTokens(java.lang.String separator,
         java.util.List tokens)
```

Invokes `concatStringSeparatedTokens(String, boolean, List)` for the case you don't expect separator be embedded into tokens.

See Also:

`concatStringSeparatedTokens(String, boolean, List)`

toCsvRecord

```
public static java.lang.String toCsvRecord(char separatorChar,
         boolean delimitTokens,
         java.util.List tokens)
```

Invokes `concatStringSeparatedTokens(String, boolean, List)` and appends the `separatorChar` at the end of result, to return the string that is one record in a .csv format *without* new line character at the end.

See Also:

`concatStringSeparatedTokens(String, boolean, List)`

concatStringSeparatedTokens

```
public static java.lang.String concatStringSeparatedTokens(java.lang.String separator,
         boolean delimitTokens,
         java.util.List tokens)
```

(continued from last page)

Concatenates `tokens` with the `separator` string between consecutive ones, but *not* at the end, and returns the resulting string. If `tokens` is null or empty, returns empty string.

This method is useful to create a line for e.g. logging and debugging with any desired separation (e.g. " | ", " / ").

For comma-separated format, use `toCsvRecord(char, boolean, List)`, which appends the `separator` at the end of the result.

Parameters:

`separator` - separator string; if null, considered as empty string.
`delimitTokens` - whether to delimit tokens; set to true if any token may contain `separator` as substring, in which case each token will be enclosed by one character #TOKEN_DELIMITTER at its start, and one at its end.
`tokens` - tokens to concatenate.

fillString

```
public static java.lang.String fillString(int count,
          char ch)
```

Returns the string filled with number `count` of characters `c`.

Parameters:

`count` - number of characters.
`ch` - the character.

truncateEnd

```
public static java.lang.String truncateEnd(java.lang.String input)
```

Identical to `truncateEnd(String, int)` with default value for `charCount` = #TRUNCATE_GREATER_THAN.

truncateEnd

```
public static java.lang.String truncateEnd(java.lang.String input,
          int charCount)
```

Truncates `input` to first `charCount` characters and appends "...". If `input` is null or empty, returns empty string. If `charCount` is greater than the `input` length, returns `input` as is.

truncateStart

```
public static java.lang.String truncateStart(java.lang.String input)
```

Identical to `truncateEnd(String, int)` with default value for `charCount` = #TRUNCATE_GREATER_THAN.

truncateStart

```
public static java.lang.String truncateStart(java.lang.String input,
          int charCount)
```

Truncates `input` to last `charCount` characters and prepends "...". If `input` is null or empty, returns empty string. If `charCount` is greater than the `input` length, returns `input` as is.

getIndentSpaces

```
public static java.lang.String getIndentSpaces(int count)
```

Returns string of spaces of the size equal to count * `INDENT_COUNT`.

(continued from last page)

getNonBreakingSpaces

`public static java.lang.String` **`getNonBreakingSpaces`**`(int count)`

Returns string of `count` non-breaking spaces.

hasContent

`public static boolean` **`hasContent`**`(java.lang.String value)`

Returns true is `value` is not null, and the trimmed content is not empty.

capitalise

`public static java.lang.String` **`capitalise`**`(java.lang.String input)`

Returns string starting with upper-case letter, all the other letters lower-case. If `in` is null or empty string, returns empty string.

sortByDecreasingLength

`public static java.lang.String[]` **`sortByDecreasingLength`**`(java.lang.String[] items)`

Returns copy of `items` sorted by decreasing length (longest first).

sortByDecreasingLength

`public static java.util.Map` **`sortByDecreasingLength`**`(java.util.Map items)`

Returns copy of `items` sorted by decreasing length of keys (longest first).

looksLikePlural

`public static boolean` **`looksLikePlural`**`(java.lang.String token)`

Returns whether `token` looks like plural; returns false for null or empty arg.

parseInt

`public static java.lang.Integer` **`parseInt`**`(java.lang.String intStr)`

Returns integer from `intStr` if it isn't null and isn't empty after trimming; otherwise returns null.

parseIntZero

`public static java.lang.Integer` **`parseIntZero`**`(java.lang.String intStr)`

Returns integer from `intStr` if it isn't null and isn't empty after trimming; otherwise returns integer with value 0.

null2empty

`public static java.lang.String` **`null2empty`**`(java.lang.String s)`

Returns empty string if `s` is null, `s` otherwise.

(continued from last page)

createKeyValuePair

```
public static java.util.Map createKeyValuePair(java.lang.Object key,
         java.lang.Object value)
```

Retruns the map with a single key/value pair. Both areguments may be null.

getKeysByValue

```
public static java.util.Set getKeysByValue(java.util.Map map,
         java.lang.Object value)
```

FIXME: tests

getKeyByValue

```
public static java.lang.Object getKeyByValue(java.util.Map map,
         java.lang.Object value)
```

FIXME: tests

initPropsFromFile

```
public static java.util.Properties initPropsFromFile(java.lang.String propsFileName)
```

Returns properties loaded from file `propsFileName` expected to be on the classpath, empty properties if the file has not been found.

Parameters:

`propsFileName` - name of the properties file expected to be on the classpath.

getResourceAbsPath

```
public static java.lang.String getResourceAbsPath(java.lang.String resourceName,
         java.lang.String detail)
  throws ApplicationException
```

Returns absolute path of the resource found on the classpath.

Parameters:

`resourceName` - name of the resource.

`detail` - optional detail to display for logging.

Returns:

absolute path of the resource found on the classpath.

Throws:

`ApplicationException` - if resource with `resourceName` is not on the classpath.

findResourceOnClasspath

```
public static java.io.InputStream findResourceOnClasspath(java.lang.String
resourceName)
  throws ResourceNotOnClasspathException
```

Returns resource as input stream for its name, given that it is found on the classpath.

Note: In this project, we have set the following directories to be on the classpath: ./config, ./input, ./test/config and ./test/input.

FIXME: test

(continued from last page)

Throws:
ResourceNotOnClasspathException

listFiles

```
public static java.util.List listFiles(java.io.File directory,
         java.io.FilenameFilter filter,
         boolean recurse)
```

Returns potentially empty list of files under `directory` (and its sub-directories if `recurse` is `true`), filtered with `filter`.

Adapted from http://snippets.dzone.com/posts/show/1875.

Parameters:
`directory`
`filter`
`recurse`

splitDirAndFileNames

```
public static java.util.List splitDirAndFileNames(java.lang.String basePath,
         java.lang.String relPath)
```

Returns (potentially empty) list of split members of `relPath`, starting immediately after the `basePath`. If the last member in the path has an extension (.extension), the name of that file is returned without extension. This is useful for creating e.g. object structure from the structure in the file system.

Parameters:
`basePath` - `relPath` string is processed after this value; if null or empty, the whole `relPath` is processed. It does *not* contain file separator.
`relPath` - actual path that should be split; if null, or (trimmed) empty string, this method is no-op.

getOutputFileRenameIfExists

```
public static java.io.File getOutputFileRenameIfExists(java.lang.String outDirName,
         java.lang.String outFileName)
  throws ApplicationException
```

Returns file #USER_DIR_KEY/`outDirName`/`outFileName` . Creates #USER_DIR_KEY/`outDirName` if it does not already exist. If the file with `outFileName` already exists, renames it by appending the system nanotime to its name.

This method is useful when generating some output files, as it ensures that the path returned on success will be valid and a potentially existing file will have been backed up.

Parameters:
`outDirName` - subdirectory under #USER_DIR_KEY that will host `outFileName`
`outFileName` - new file name

Returns:
absolute path #USER_DIR_KEY/`outDirName`/`outFileName` .

Throws:
ApplicationException - if fails to create `outFileName`, if fails to rename existing file with name `outFileName`

getFileExtension

```
public static java.lang.String getFileExtension(java.lang.String filePath)
```

Returns extension (after the last ".") if being part of `filePath`, null otherwise. Implementation from StackOverflow

getFileExtensionWithDot

```
public static java.lang.String getFileExtensionWithDot(java.lang.String filePath)
```

Returns extension with the "." if being part of `filePath`, null otherwise.

getDirectory

```
public static java.io.File getDirectory(java.lang.String dirRelPath,
         boolean createIfMissing)
```

Returns file representing directory `dirName` under #USER_DIR_KEY.

Parameters:

`dirRelPath` - relative path of directory.

`createIfMissing` - whether to create `dirRelPath` if currently not existing under #USER_DIR_KEY

Returns:

file representing directory `dirRelPath` under #USER_DIR_KEY, or null if there was an OS-related problem that didn't allow for creation of directory.

copy

```
public static void copy(java.io.File src,
         java.io.File dst)
  throws java.io.IOException
```

Copies src file to dst file.

delete

```
public static void delete(java.io.File f)
```

Wrapper for the `File.delete()` that accepts null argument and returns nothing. In case delete failed, just logs the failure.

Parameters:

`f` - potentially null

saveToFile

```
public static java.io.File saveToFile(java.lang.String filePath,
         java.lang.String content)
  throws java.io.IOException
```

Saves `content` to `filePath` and logs the confirmation with `level` and return the file.

TODO: test

Throws:

`IOException`

createTempImageFile

```
public static java.io.File createTempImageFile(java.lang.String dirAbsPath,
         java.lang.String fileName,
         Util.ImageFormat format,
         boolean deleteOnExit)
  throws java.io.IOException
```

Creates file in the given directory (or in default OS tmp directory) and returns the result.

(continued from last page)

Parameters:

`dirAbsPath` - absolute path of the file; if null, temporary directory is used.
`fileName` - name of the file (witout path, without extension).
`format` - image format.
`deleteOnExit` - whether to delete the file on application exit.

Returns:

created temporary file.

Throws:

`IOException` - if a file could not be created.

clearClipboard

```
public static void clearClipboard()
  throws ApplicationException
```

Clears system clipboard.

Throws:

`ApplicationException`

copyTextToClipboard

```
public static void copyTextToClipboard(java.lang.String txt)
```

Copies non-empty, non-null `txt` to clipboard, no-op otherwise. Use this method if you have a raw text or a well-formed HTML document.

Parameters:

`txt` - text to put to the clipboard

See Also:

`copyHtmlToClipboard(String)`

copyHtmlToClipboard

```
public static void copyHtmlToClipboard(java.lang.String htmlBody)
```

Surrounds the non-empty, non-null `htmlBody` into doctype and html tags to produce a valid HTML document; no-op otherwise. Use this method if you have some markup snippet.

Parameters:

`htmlBody` - markup to put to the clipboard

See Also:

`copyTextToClipboard(String)`

fetchTextFromClipboard

```
public static java.lang.String fetchTextFromClipboard()
  throws ApplicationException
```

Returns text contained in the clipboard (text could be plain or markup), null if clipboard is empty.

Throws:

`ApplicationException` - if the data is no longer available in the clipboard in the requested flavor.

copyImageToClipboard

```
public static void copyImageToClipboard(java.io.File pic)
```

Copies image in `pic` to clipboard.

saveImageFromClipboard

```
public static void saveImageFromClipboard(java.io.File pic)
  throws java.io.IOException
```

Parameters:
`pic` - file where to store the image

Throws:
`IOException`

formatDuration

```
public static java.lang.String formatDuration(long millis)
```

ensureNotNull

```
public static void ensureNotNull(java.lang.Object arg,
         java.lang.String name)
```

ensureNotEmpty

```
public static void ensureNotEmpty(java.lang.String arg,
         java.lang.String name)
```

ensureNotEmpty

```
public static void ensureNotEmpty(java.util.Map arg,
         java.lang.String name)
```

ensureNotEmpty

```
public static void ensureNotEmpty(java.util.Collection arg,
         java.lang.String name)
```

ensureNotEmpty

```
public static void ensureNotEmpty(boolean[] arg,
         java.lang.String name)
```

(continued from last page)

ensureNotEmpty

```
public static void ensureNotEmpty(java.lang.Object[] arg,
         java.lang.String name)
```

ensureContainsNoNull

```
public static void ensureContainsNoNull(java.lang.Object[] arg,
         java.lang.String name)
```

ensureContainsNoNull

```
public static void ensureContainsNoNull(java.util.Collection arg,
         java.lang.String name)
```

ensureNotEmpty

```
public static void ensureNotEmpty(int[] arg,
         java.lang.String name)
```

logTitle

```
public static void logTitle(org.apache.log4j.Level level,
         java.lang.String[] title)
```

Logs `title` with `level` (for major steps in the application).

logSubtitle

```
public static void logSubtitle(org.apache.log4j.Level level,
         java.lang.String subtitle)
```

Logs `subtitle` with `level` (for sub-steps in the application).

logCompletion

```
public static void logCompletion(org.apache.log4j.Level level,
         java.lang.String text,
         long startMillis,
         boolean skipTime)
```

Logs `text` with `level` (and if `skipTime=false`, duration since `startMillis`).

logCollection

```
public static void logCollection(org.apache.log4j.Level level,
         java.util.Collection objects,
         java.lang.String what)
```

Logs each element in `objects`.

(continued from last page)

Parameters:

`level` - logging level.
`objects` - objects to log.
`what` - title to print when `objects` is not empty.

logMap

```
public static void logMap(org.apache.log4j.Level level,
          java.util.Map objects,
          java.lang.String what)
```

Logs each element in `objects`.

Parameters:

`level` - logging level.
`objects` - objects to log.
`what` - title to print when `objects` is not empty.

org.tanjakostic.jcleancim.util
Class Util.ImageFormat

```
java.lang.Object
  +-java.lang.Enum
    +-org.tanjakostic.jcleancim.util.Util.ImageFormat
```

All Implemented Interfaces:
java.io.Serializable, java.lang.Comparable

---

public static final class **Util.ImageFormat**
extends java.lang.Enum

Image formats supported for UML diagrams.

Field Summary

public static final	BMP
public static final	JPG
public static final	PNG

Method Summary

static Util.ImageFormat	getDefault() Default is PNG.
java.lang.String	getExtensionName()
java.lang.String	getExtensionWithDot()
static Util.ImageFormat	valueOf(java.lang.String name)
static Util.ImageFormat[]	values()

Methods inherited from class `java.lang.Enum`

clone, compareTo, equals, finalize, getDeclaringClass, hashCode, name, ordinal, toString, valueOf

Methods inherited from class `java.lang.Object`

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface `java.lang.Comparable`

compareTo

Fields

BMP

```
public static final org.tanjakostic.jcleancim.util.Util.ImageFormat BMP
```

JPG

```
public static final org.tanjakostic.jcleancim.util.Util.ImageFormat JPG
```

PNG

```
public static final org.tanjakostic.jcleancim.util.Util.ImageFormat PNG
```

Methods

values

```
public static Util.ImageFormat[] values()
```

valueOf

```
public static Util.ImageFormat valueOf(java.lang.String name)
```

getExtensionWithDot

```
public java.lang.String getExtensionWithDot()
```

getExtensionName

```
public java.lang.String getExtensionName()
```

getDefault

```
public static Util.ImageFormat getDefault()
```

Default is `PNG`.

Package
org.tanjakostic.jcleancim.validation

Classes responsible for validating the model and the rules to apply.

Main classes are:

- `ModelValidator` class - launches validation by delegating to other *Validator classes for the scope defined in the org.tanjakostic.jcleancim.common.Config#DEFAULT_PROPS_FILE_NAME file. These latter all inherit from `AbstractValidator` for the common implementation, which allows to have very thin concrete validators.
- `org.tanjakostic.jcleancim.validation.Rule` interface and the interfaces extending it (`SimpleRule`, `CrossRule`) - these allow for simplified processing implemented in `AbstractValidator`. Concrete rules inherit from `AbstractRule` and need to implement only the necessary minimum.

All concrete validators include mostly simple rules, and some include more complex (bulk and/or cross) rules.

To add a new rule, there are 2 things to do in the corresponding org.tanjakostic.jcleancim.validation.*Validator.java file:

- add a class for the new rule (similar to existing rules), make it inherit from `AbstractRule` and implement one of `SimpleRule` or `CrossRule` interfaces; and
- in the constructor of the corresponding validator, add that new rule to appropriate collection, following the same pattern as existing ones.

TODO:

Further validation rules to add:

- CMM doc, pg. 23 - Check against the naming rules and tag as warning everything that does not fit (according to CIM model management document)
- upperCase(CIM_WARN, "label should start with lower case"); to be applied to attributes and enum labels of classes, except for the following: UnitSymbol, UnitMultiplier, Currency, MonetaryAmountPerEnergyUnit, MonetaryAmountPerHeatUnit, MonetaryAmountRate; these must preserve the case
- missingSforMultipleSideRole(CIM_WARN, "label should end with an 's'"); applicable to association end names with multiplicity [0..n], [1..n]; one will have to add "exclusion filter"
- superfluousSforSingleSideRole(CIM_WARN, "label should not end with an 's'"): applicable to association end names with multiplicity [0..n], [1..n]; one will have to add "exclusion filter" (e.g., for address, status)

org.tanjakostic.jcleancim.validation
Class AbstractRule

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.validation.AbstractRule
```

All Implemented Interfaces:
Rule

Direct Known Subclasses:
AbstractRuleWithSubobjectsAndSkips, CimAssociationEndsNameShouldBeSingular, CimAssociationEndsNameShouldBePlural, CimAssociationEndsNameStartingWithLowerCase, Iec61850AssociationsWithDifferentEndVisibility, Iec61850AssociationsThatShouldBePrivate, AssociationsWithWrongSource, AssociationsWithNoMultiplicity, AssociationsMissingInformativeStereotype, AssociationsWithName, AssociationsWithSameDocOnBothEnds, AssociationsWithDoc, AssociationsWithRoleBadDirection, AssociationsWithExplicitDirection, Iec61850ConditionLiteralsNeverUsedAsConstraints, Iec61850DOAttributesWithSameNameDifferentType, Iec61850DOAbbreviationLiteralsNeverUsedInDOName, Iec61850DOAbbreviationLiteralsDuplicateDescription, Iec61850DOAbbreviationLiteralsDuplicateName, AttributesWithTypeFromUnallowedOwner, Iec61850DOAttributesNameStartingWithLowerCase, Iec61850AbbreviationLiteralsNameStartingWithLowerCase, CimAttributesNameShouldNotStartWithClassName, CimAttributesNameShouldBeSingular, CimAttributesNameStartingWithUpperCase, Iec61850DOAttributesWithNameMissingAbbreviation, AttributesWithInexistingEnumLiteralAsInitValue, Iec61850FCDAAttributesWithMissingConstraint, Iec61850DOAttributesWithTooLongName, CimAttributesWithFlagInName, Iec61850AttributesWithInexistingSibling, AttributesWhoseTypeIsInformative, CimAttributesThatShouldBeReplacedWithAssociation, AttributesThatAreEnumsInNonEnumeratedClass, AttributesThatAreConstNonStatic, CimAttributesThatAreNotStaticNonConstWithInitVal, AttributesThatAreStaticButNotConst, CimAttributesThatShouldBePublic, AttributesWithTypeIdMismatch, AttributesWithInvalidTypeString, AttributesWithInvalidTypeNull, CimAttributesThatShouldBeOptional, AttributesWithInvalidMultiplicity, EnumLiteralsWithoutEnumStereotype, EnumLiteralsWithSuperfluousType, CimClassesNeverUsedAsTypeForAttribute, ClassesWithSameName, EnumClassesWithDuplicateCodes, EnumClassesWithSomeCodesMissing, Iec61850LNClassesMalformedName, Iec61850LNClassesInWrongGroup, CimClassesNameShouldBeSingular, CimClassesNameStartingWithLowerCase, CimDatatypeClassesWithInvalidAttributes, Iec61850ClassesWithMissingCondIDTextInConstraints, Iec61850LNClassesWithSuperfluousConstraints, Iec61850ClassesWithInvalidConstraints, CimClassesNeverUsedInRelationships, Iec61850ClassesThatShouldHaveTaggedValuesForDocgen, Iec61850ClassesThatShouldHaveAliasAsTitle, ClassesThatShouldNotHaveNestingThroughAttribute, CimClassesThatShouldNotHaveExplicitDependencies, CimClassesThatShouldNotHaveOperations, CimClassesThatShouldNotBeAbstract, CimClassesUsedForAttributesButHaveSuperclasses, CimClassesUsedForAttributesButHaveSubclasses, CimClassesUsedForAttributesButHaveAssociations, CimClassesWithOldDatatypeStereotype, ClassesThatShouldNotBeAssociationClass, ClassesWithSuperclassesFromUnallowedOwner, ClassesWithMultipleSuperclasses, ClassesWithPersistentPropSet, ClassesWithRootPropSet, ClassesWithLeafPropSet, ClassesWithSelfDependency, ClassesWithDuplicateOwnOrInheritedAssociationEndNames, ClassesWithDuplicateInheritedAttributeNames, CimPrimitiveClassesWithIllegalOwner, CimPrimitiveClassesWithAttributes, ClassesWithSelfInheritance, EnumClassesWithBadName, ClassesWithQuestionableAttributeCount, ClassesWithUnexpectedConnectors, CimClassesWithUnexpectedElements, DependenciesWithUnallowedDirection, DiagramsWithBadOrientation, OperationsWithInvalidExcTypeNull, OperationsWithInvalidArgTypeNull, OperationsWithInvalidReturnTypeNull, OperationsWithUpperCaseName, PackagesWithSameName, Iec61850PackagesThatShouldHaveAliasAsTitle, PackagesTopLevelWithoutVersionClass, PackagesWithSelfDependency, PackageUnexpectedConnectors, PackageUnexpectedElements

---

public abstract class **AbstractRule**
extends java.lang.Object
implements Rule

Implements logging uniformly for all concrete implementations.

Nested Class Summary

class	AbstractRule.AbstractRuleWithSubobjectsAndSkips AbstractRule.AbstractRuleWithSubobjectsAndSkips
class	AbstractRule.UmlObjectsMissingDoc AbstractRule.UmlObjectsMissingDoc
class	AbstractRule.UmlObjectsWithBadCharacterInName AbstractRule.UmlObjectsWithBadCharacterInName
class	AbstractRule.UmlObjectsWithBadDocEnd AbstractRule.UmlObjectsWithBadDocEnd
class	AbstractRule.UmlObjectsWithBadDocStart AbstractRule.UmlObjectsWithBadDocStart
class	AbstractRule.UmlObjectsWithUnallowedStereotype AbstractRule.UmlObjectsWithUnallowedStereotype
class	AbstractRule.UmlObjectsWithUnallowedTagNames AbstractRule.UmlObjectsWithUnallowedTagNames

Constructor Summary

protected	AbstractRule(org.apache.log4j.Logger extLogger, java.lang.String hypothesis, java.lang.String howToFix) Constructor; default level is ERROR.
protected	AbstractRule(org.apache.log4j.Logger extLogger, org.apache.log4j.Level level, Rule.Severity severity, Rule.Category category, java.lang.String hypothesis, java.lang.String howToFix) Constructor.

Method Summary

ModelIssue	createIssue(UmlObject subject)
ModelIssue	createIssue(UmlObject subject, java.lang.String evidence)
ModelIssue	createIssue(UmlObject subject, java.lang.String evidence, java.lang.String subjectDescription, java.lang.String groupTag)
Rule.Category	getCategory()
java.lang.String	getHowToFix()
java.lang.String	getHypothesis()
org.apache.log4j.Level	getLogLevel()
Rule.Severity	getSeverity()
void	logDiagnosis(boolean verbose, ModelIssues issues)

Methods inherited from class `java.lang.Object`

`clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait`

Methods inherited from interface `org.tanjakostic.jcleancim.validation.Rule`
`getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis`

Constructors

AbstractRule

```
protected AbstractRule(org.apache.log4j.Logger extLogger,
                       java.lang.String hypothesis,
                       java.lang.String howToFix)
```

Constructor; default level is ERROR.

Parameters:

`extLogger` - logger to use; if null, abstract class logger is used.
`hypothesis` - non-null, non-empty string to use as a title of logging entries.
`howToFix` - non-null, non-empty string to use as a title of logging entries.

AbstractRule

```
protected AbstractRule(org.apache.log4j.Logger extLogger,
                       org.apache.log4j.Level level,
                       Rule.Severity severity,
                       Rule.Category category,
                       java.lang.String hypothesis,
                       java.lang.String howToFix)
```

Constructor.

Parameters:

`extLogger` - logger to use; if null, abstract class logger is used.
`level` - logging level; if null, default level is ERROR.
`severity` - severity; if null, default severity is high.
`category` - category; if null, default category is modellingRule.
`hypothesis` - non-null, non-empty string to use as a title of logging entries.
`howToFix` - non-null, non-empty string to use as a title of logging entries.

Methods

getCategory

```
public Rule.Category getCategory()
```

getSeverity

```
public Rule.Severity getSeverity()
```

getHypothesis

```
public java.lang.String getHypothesis()
```

getHowToFix

```
public java.lang.String getHowToFix()
```

logDiagnosis

```
public final void logDiagnosis(boolean verbose,
         ModelIssues issues)
```

getLogLevel

```
public final org.apache.log4j.Level getLogLevel()
```

createIssue

```
protected final ModelIssue createIssue(UmlObject subject)
```

createIssue

```
protected final ModelIssue createIssue(UmlObject subject,
         java.lang.String evidence)
```

createIssue

```
protected final ModelIssue createIssue(UmlObject subject,
         java.lang.String evidence,
         java.lang.String subjectDescription,
         java.lang.String groupTag)
```

org.tanjakostic.jcleancim.validation
Class AbstractRule.AbstractRuleWithSubobjectsAndSkips

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.validation.AbstractRule
      |
      +-
```
org.tanjakostic.jcleancim.validation.AbstractRule.AbstractRuleWithSubobjectsAndSkips

All Implemented Interfaces:
SimpleRule, Rule

Direct Known Subclasses:
UmlObjectsWithBadCharacterInName, UmlObjectsWithBadDocEnd, UmlObjectsWithBadDocStart, UmlObjectsMissingDoc, UmlObjectsWithUnallowedTagNames, UmlObjectsWithUnallowedStereotype

---

public static abstract class **AbstractRule.AbstractRuleWithSubobjectsAndSkips**
extends AbstractRule
implements Rule, SimpleRule

Common superclass where a rule applies to multiple UML object types (e.g., package and association) and their sub-objects (e.g., association ends) . Allows also to skip validation by letting subtypes override `skipValidation(T)` and/or `skipSubobjectValidation(T)`. This is necessary in particular for complex IEC61850 models, to avoid lots of noise where e.g. we don't care about a doc for something that is just a modelling artefact, but not really part of the official specification.
Parameters:
T

Constructor Summary

protected	`AbstractRuleWithSubobjectsAndSkips(org.apache.log4j.Logger extLogger, org.apache.log4j.Level level, Rule.Severity severity, Rule.Category category, java.lang.String hypothesis, java.lang.String howToFix, java.lang.String what)`

Method Summary

abstract void	`doValidate(UmlObject o, ModelIssues issues)`
java.util.List	`getSubObjects(UmlObject o)` Used for validation of "sub-objects" (such as association ends and operation parameters) that do not have their own validators, but get validated through their container.
boolean	`skipSubobjectValidation(UmlObject o)` This default implementation returns false (no skipping); override if sub-objects don't need validation.
boolean	`skipValidation(UmlObject o)` This default implementation returns false (no skipping); override if main object doesn't need validation.
void	`validate(UmlObject o, ModelIssues issues)`

Methods inherited from class `org.tanjakostic.jcleancim.validation.AbstractRule`

createIssue, createIssue, createIssue, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from class `java.lang.Object`
clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface `org.tanjakostic.jcleancim.validation.Rule`
getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from interface `org.tanjakostic.jcleancim.validation.SimpleRule`
validate

Methods inherited from interface `org.tanjakostic.jcleancim.validation.Rule`
getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Constructors

AbstractRuleWithSubobjectsAndSkips

```
protected AbstractRuleWithSubobjectsAndSkips(org.apache.log4j.Logger extLogger,
                                             org.apache.log4j.Level level,
                                             Rule.Severity severity,
                                             Rule.Category category,
                                             java.lang.String hypothesis,
                                             java.lang.String howToFix,
                                             java.lang.String what)
```

Methods

validate

```
public final void validate(UmlObject o,
         ModelIssues issues)
```

doValidate

```
protected abstract void doValidate(UmlObject o,
         ModelIssues issues)
```

getSubObjects

```
protected java.util.List getSubObjects(UmlObject o)
```

(continued from last page)

Used for validation of "sub-objects" (such as association ends and operation parameters) that do not have their own validators, but get validated through their container.

This default implementation returns empty list; subtypes having contained objects that need to be validated against this rule have to override this method.

skipSubobjectValidation

```
protected boolean skipSubobjectValidation(UmlObject o)
```

This default implementation returns false (no skipping); override if sub-objects don't need validation.

skipValidation

```
protected boolean skipValidation(UmlObject o)
```

This default implementation returns false (no skipping); override if main object doesn't need validation.

org.tanjakostic.jcleancim.validation
Class AbstractRule.UmlObjectsWithUnallowedStereotype

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.validation.AbstractRule
      |
      +-
org.tanjakostic.jcleancim.validation.AbstractRule.AbstractRuleWithSubobjectsAndSkips
          |
          +-
```
org.tanjakostic.jcleancim.validation.AbstractRule.UmlObjectsWithUnallowedStereotype

All Implemented Interfaces:
Rule, SimpleRule

Direct Known Subclasses:
AssociationEndsWithUnallowedStereotype, AssociationsWithUnallowedStereotype, AttributesWithUnallowedStereotype, ClassesWithUnallowedStereotype, DependenciesWithUnallowedStereotype, DiagramsWithUnallowedStereotype, OperationParametersWithUnallowedStereotype, OperationsWithUnallowedStereotype, PackagesWithUnallowedStereotype

---

public static abstract class **AbstractRule.UmlObjectsWithUnallowedStereotype**
extends AbstractRule.AbstractRuleWithSubobjectsAndSkips

Constructor Summary

protected	UmlObjectsWithUnallowedStereotype(org.apache.log4j.Logger logger, java.lang.String what, java.util.Map alloweds)

Method Summary

void	doValidate(UmlObject o, ModelIssues issues)

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule.AbstractRuleWithSubobjectsAndSkips

doValidate, getSubObjects, skipSubobjectValidation, skipValidation, validate

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule

createIssue, createIssue, createIssue, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from interface org.tanjakostic.jcleancim.validation.SimpleRule

validate

Methods inherited from interface `org.tanjakostic.jcleancim.validation.Rule`
`getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis`

Constructors

UmlObjectsWithUnallowedStereotype

```
protected UmlObjectsWithUnallowedStereotype(org.apache.log4j.Logger logger,
                                            java.lang.String what,
                                            java.util.Map alloweds)
```

Methods

doValidate

```
protected final void doValidate(UmlObject o,
         ModelIssues issues)
```

Matches any stereotype not in the set passed in at creation, as allowed stereotype for this o's concrete type, depending on its (model) nature.

org.tanjakostic.jcleancim.validation
Class AbstractRule.UmlObjectsWithUnallowedTagNames

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.validation.AbstractRule
      |
      +-
org.tanjakostic.jcleancim.validation.AbstractRule.AbstractRuleWithSubobjectsAndSkips
          |
          +-
org.tanjakostic.jcleancim.validation.AbstractRule.UmlObjectsWithUnallowedTagNames
```

All Implemented Interfaces:
Rule, SimpleRule

Direct Known Subclasses:
AssociationEndsWithUnallowedTagNames, AssociationsWithUnallowedTagNames, AttributesWithUnallowedTagNames, ClassesWithUnallowedTagNames, DependenciesWithUnallowedTagNames, OperationParametersWithUnallowedTagNames, OperationsWithUnallowedTagNames, PackagesWithUnallowedTagNames

---

public static abstract class **AbstractRule.UmlObjectsWithUnallowedTagNames**
extends AbstractRule.AbstractRuleWithSubobjectsAndSkips

Constructor Summary

public	UmlObjectsWithUnallowedTagNames(org.apache.log4j.Logger logger, java.lang.String what)

Method Summary

void	doValidate(UmlObject o, ModelIssues issues)

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule.AbstractRuleWithSubobjectsAndSkips

doValidate, getSubObjects, skipSubobjectValidation, skipValidation, validate

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule

createIssue, createIssue, createIssue, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from interface org.tanjakostic.jcleancim.validation.SimpleRule

validate

Methods inherited from interface `org.tanjakostic.jcleancim.validation.Rule`
`getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis`

Constructors

UmlObjectsWithUnallowedTagNames

```
public UmlObjectsWithUnallowedTagNames(org.apache.log4j.Logger logger,
                                       java.lang.String what)
```

Methods

doValidate

```
protected void doValidate(UmlObject o,
         ModelIssues issues)
```

org.tanjakostic.jcleancim.validation
Class AbstractRule.UmlObjectsMissingDoc

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.validation.AbstractRule
    |
    +-
org.tanjakostic.jcleancim.validation.AbstractRule.AbstractRuleWithSubobjectsAndSkips
      |
      +-org.tanjakostic.jcleancim.validation.AbstractRule.UmlObjectsMissingDoc
```

All Implemented Interfaces:
Rule, SimpleRule

Direct Known Subclasses:
AssociationEndsMissingDoc, AttributesMissingDoc, ClassesMissingDoc, DiagramsMissingDoc, OperationParametersMissingDoc, OperationsMissingDoc, PackagesMissingDoc

---

public static abstract class **AbstractRule.UmlObjectsMissingDoc**
extends AbstractRule.AbstractRuleWithSubobjectsAndSkips

Constructor Summary

protected	UmlObjectsMissingDoc(org.apache.log4j.Logger logger, java.lang.String what)

Method Summary

void	doValidate(UmlObject o, ModelIssues issues)

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule.AbstractRuleWithSubobjectsAndSkips

doValidate, getSubObjects, skipSubobjectValidation, skipValidation, validate

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule

createIssue, createIssue, createIssue, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from interface org.tanjakostic.jcleancim.validation.SimpleRule

validate

Methods inherited from interface `org.tanjakostic.jcleancim.validation.Rule`
`getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis`

Constructors

UmlObjectsMissingDoc

```
protected UmlObjectsMissingDoc(org.apache.log4j.Logger logger,
                               java.lang.String what)
```

Methods

doValidate

```
protected final void doValidate(UmlObject o,
         ModelIssues issues)
```

Matches normative items that miss description.

org.tanjakostic.jcleancim.validation
Class AbstractRule.UmlObjectsWithBadDocStart

```
java.lang.Object
  |-org.tanjakostic.jcleancim.validation.AbstractRule
    |-
org.tanjakostic.jcleancim.validation.AbstractRule.AbstractRuleWithSubobjectsAndSkips
      |-org.tanjakostic.jcleancim.validation.AbstractRule.UmlObjectsWithBadDocStart
```

All Implemented Interfaces:
Rule, SimpleRule

Direct Known Subclasses:
AssociationEndsWithBadDocStart, AttributesWithBadDocStart, ClassesWithBadDocStart, DiagramsWithBadDocStart, OperationParametersWithBadDocStart, OperationsWithBadDocStart, PackagesWithBadDocStart

---

public static abstract class **AbstractRule.UmlObjectsWithBadDocStart**
extends AbstractRule.AbstractRuleWithSubobjectsAndSkips

Constructor Summary

protected	UmlObjectsWithBadDocStart(org.apache.log4j.Logger logger, java.lang.String what)

Method Summary

void	doValidate(UmlObject o, ModelIssues issues)

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule.AbstractRuleWithSubobjectsAndSkips

doValidate, getSubObjects, skipSubobjectValidation, skipValidation, validate

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule

createIssue, createIssue, createIssue, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from interface org.tanjakostic.jcleancim.validation.SimpleRule

validate

Methods inherited from interface `org.tanjakostic.jcleancim.validation.Rule`
`getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis`

Constructors

UmlObjectsWithBadDocStart

```
protected UmlObjectsWithBadDocStart(org.apache.log4j.Logger logger,
                                    java.lang.String what)
```

Methods

doValidate

```
protected final void doValidate(UmlObject o,
         ModelIssues issues)
```

Matches objects with non-empty description starting with a non-upper case letter or another character not in the allowed list `DOCSTART_CHARS`.

org.tanjakostic.jcleancim.validation
Class AbstractRule.UmlObjectsWithBadDocEnd

```
java.lang.Object
  |-org.tanjakostic.jcleancim.validation.AbstractRule
    |-
org.tanjakostic.jcleancim.validation.AbstractRule.AbstractRuleWithSubobjectsAndSkips
      |-org.tanjakostic.jcleancim.validation.AbstractRule.UmlObjectsWithBadDocEnd
```

All Implemented Interfaces:
Rule, SimpleRule

Direct Known Subclasses:
AssociationEndsWithBadDocEnd, AttributesWithBadDocEnd, ClassesWithBadDocEnd, DiagramsWithBadDocEnd, OperationParametersWithBadDocEnd, OperationsWithBadDocEnd, PackagesWithBadDocEnd

---

public static abstract class **AbstractRule.UmlObjectsWithBadDocEnd**
extends AbstractRule.AbstractRuleWithSubobjectsAndSkips

Constructor Summary

protected	UmlObjectsWithBadDocEnd(org.apache.log4j.Logger logger, java.lang.String what)

Method Summary

void	doValidate(UmlObject o, ModelIssues issues)

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule.AbstractRuleWithSubobjectsAndSkips

doValidate, getSubObjects, skipSubobjectValidation, skipValidation, validate

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule

createIssue, createIssue, createIssue, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from interface org.tanjakostic.jcleancim.validation.SimpleRule

validate

Methods inherited from interface `org.tanjakostic.jcleancim.validation.Rule`
`getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis`

Constructors

UmlObjectsWithBadDocEnd

```
protected UmlObjectsWithBadDocEnd(org.apache.log4j.Logger logger,
                                  java.lang.String what)
```

Methods

doValidate

```
protected final void doValidate(UmlObject o,
         ModelIssues issues)
```

Matches objects with non-empty description ending with a character different than '.'.

org.tanjakostic.jcleancim.validation
Class AbstractRule.UmlObjectsWithBadCharacterInName

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.validation.AbstractRule
      |
      +-
org.tanjakostic.jcleancim.validation.AbstractRule.AbstractRuleWithSubobjectsAndSkips
          |
          +-
```
org.tanjakostic.jcleancim.validation.AbstractRule.UmlObjectsWithBadCharacterInName

All Implemented Interfaces:
Rule, SimpleRule

Direct Known Subclasses:
AssociationEndsWithBadCharacterInName, Iec61850AttributesWithBadCharacterInName, CimAttributesWithBadCharacterInName, ClassesWithBadCharacterInName, DiagramsWithBadCharacterInName, OperationParametersWithBadCharacterInName, OperationsWithBadCharacterInName, PackagesWithBadCharacterInName

---

public static abstract class **AbstractRule.UmlObjectsWithBadCharacterInName**
extends AbstractRule.AbstractRuleWithSubobjectsAndSkips

Constructor Summary

protected	UmlObjectsWithBadCharacterInName(org.apache.log4j.Logger logger, java.lang.String what)

Method Summary

void	doValidate(UmlObject o, ModelIssues issues)
abstract InvalidCharactersFinder	getInvalidCharacterFinder(UmlObject o)

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule.AbstractRuleWithSubobjectsAndSkips

doValidate, getSubObjects, skipSubobjectValidation, skipValidation, validate

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule

createIssue, createIssue, createIssue, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from interface org.tanjakostic.jcleancim.validation.SimpleRule
validate

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule
getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Constructors

UmlObjectsWithBadCharacterInName

```
protected UmlObjectsWithBadCharacterInName(org.apache.log4j.Logger logger,
                                           java.lang.String what)
```

Methods

doValidate

```
protected final void doValidate(UmlObject o,
         ModelIssues issues)
```

Matches non-empty name that has one or more characters as provided by getInvalidCharacterFinder(UmlObject).

getInvalidCharacterFinder

```
protected abstract InvalidCharactersFinder getInvalidCharacterFinder(UmlObject o)
```

org.tanjakostic.jcleancim.validation
Class AbstractValidator

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.validation.AbstractValidator
```

Direct Known Subclasses:
AssociationValidator, AttributeValidator, ClassValidator, DependencyValidator, DiagramValidator, OperationValidator, PackageValidator

---

public abstract class **AbstractValidator**
extends java.lang.Object

Common implementation for all UML object validators (package, class, etc.). An element validator instantiates concrete `org.tanjakostic.jcleancim.validation.Rule`-s.

This class controls the execution of validation with respect to:

- enabled/disabled validators - this status is specified with separate validation properties in the org.tanjakostic.jcleancim.common.Config#DEFAULT_PROPS_FILE_NAME file and available through the 7 configuration instance methods on configuration (e.g. `Config.isValidationAssociationsOn()`), one per validator type. Setting one of these to false at configuration disables validation for all the rules for that type of element (for this example, all rules validating associations).
- enabled/disabled status of individual rules - there is one configuration option whose value gets returned from configuration with `Config.getValidationRulesOff()`. It contains fine-grained filtering for individual rules: those specified in the configuration are skipped.

Violated rules on UML objects from the model produce issues, and they can be logged and reported.
Parameters:
`T`

Constructor Summary

protected	`AbstractValidator(Config cfg, int totalCount, java.lang.String which, ModelIssues issues)` Constructor.

Method Summary

boolean	`addCrossRule(CrossRule crossRule)`
boolean	`addSimpleRule(SimpleRule simpleRule)`
java.lang.String	`displayAllAvailableRuleNames()` Returns the list of strings, including heading, suitable for logging.
java.lang.String	`displayAvailableRuleNames(Nature nature)` Returns flattened list of strings, including heading, with new line character as separator; suitable for pasting into a document (e.g., CIM model management or 61850 UML model management document).
abstract boolean	`enabled()` Returns whether the validation for this validator has been enabled (by configuration).

java.util.List	getAllCrossRules() Returns all cross rules available.
java.util.List	getAllRules()
java.util.List	getAllSimpleRules() Returns all simple rules available.
Config	getCfg() Returns configuration.
java.util.List	getCheckedCrossRules() Returns only checked (non-disabled) cross rules.
java.util.List	getCheckedRules()
java.util.List	getCheckedSimpleRules() Returns only checked (non-disabled) simple rules.
ModelIssues	getCollectedIssues()
abstract java.util.List	getScopedUmlObjects() Returns the elements retained for validation, for the configured scope.
void	validate() If validation has been enabled in the configuration for the type T of element, performs validation according to (in the configuration) non-disabled individual rules, and logs diagnosis.

Methods inherited from class `java.lang.Object`
`clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait`

Constructors

AbstractValidator

```
protected AbstractValidator(Config cfg,
                            int totalCount,
                            java.lang.String which,
                            ModelIssues issues)
```

Constructor.

Parameters:

`cfg` - configuration
`totalCount` - total count of elements in the model
`which` - kind of element - used only for logging
`issues` - home for issues that get collected through validation

Methods

(continued from last page)

addSimpleRule

```
protected final boolean addSimpleRule(SimpleRule simpleRule)
```

addCrossRule

```
protected final boolean addCrossRule(CrossRule crossRule)
```

getCfg

```
public final Config getCfg()
```

Returns configuration.

getCollectedIssues

```
public ModelIssues getCollectedIssues()
```

validate

```
public final void validate()
```

If validation has been enabled in the configuration for the type T of element, performs validation according to (in the configuration) non-disabled individual rules, and logs diagnosis.

enabled

```
public abstract boolean enabled()
```

Returns whether the validation for this validator has been enabled (by configuration).

getScopedUmlObjects

```
public abstract java.util.List getScopedUmlObjects()
```

Returns the elements retained for validation, for the configured scope.

getCheckedRules

```
public final java.util.List getCheckedRules()
```

getAllRules

```
public final java.util.List getAllRules()
```

(continued from last page)

getCheckedSimpleRules

`public final java.util.List` **`getCheckedSimpleRules`**`()`

Returns only checked (non-disabled) simple rules.

getCheckedCrossRules

`public final java.util.List` **`getCheckedCrossRules`**`()`

Returns only checked (non-disabled) cross rules.

getAllSimpleRules

`public final java.util.List` **`getAllSimpleRules`**`()`

Returns all simple rules available.

getAllCrossRules

`public final java.util.List` **`getAllCrossRules`**`()`

Returns all cross rules available.

displayAllAvailableRuleNames

`public java.lang.String` **`displayAllAvailableRuleNames`**`()`

Returns the list of strings, including heading, suitable for logging.

displayAvailableRuleNames

`public java.lang.String` **`displayAvailableRuleNames`**`(Nature nature)`

Returns flattened list of strings, including heading, with new line character as separator; suitable for pasting into a document (e.g., CIM model management or 61850 UML model management document).

If `nature` is null, then vistis simply all the rules, without concern about their applicability per nature.

org.tanjakostic.jcleancim.validation
Class AssociationValidator

```
java.lang.Object
  |-org.tanjakostic.jcleancim.validation.AbstractValidator
    |-org.tanjakostic.jcleancim.validation.AssociationValidator
```

public class **AssociationValidator**
extends AbstractValidator

Validates associations.

Nested Class Summary

class	AssociationValidator.AssociationEndsMissingDoc AssociationValidator.AssociationEndsMissingDoc
class	AssociationValidator.AssociationEndsWithBadCharacterInName AssociationValidator.AssociationEndsWithBadCharacterInName
class	AssociationValidator.AssociationEndsWithBadDocEnd AssociationValidator.AssociationEndsWithBadDocEnd
class	AssociationValidator.AssociationEndsWithBadDocStart AssociationValidator.AssociationEndsWithBadDocStart
class	AssociationValidator.AssociationEndsWithUnallowedStereotype AssociationValidator.AssociationEndsWithUnallowedStereotype
class	AssociationValidator.AssociationEndsWithUnallowedTagNames AssociationValidator.AssociationEndsWithUnallowedTagNames
class	AssociationValidator.AssociationsMissingInformativeStereotype AssociationValidator.AssociationsMissingInformativeStereotype
class	AssociationValidator.AssociationsWithDoc AssociationValidator.AssociationsWithDoc
class	AssociationValidator.AssociationsWithExplicitDirection AssociationValidator.AssociationsWithExplicitDirection
class	AssociationValidator.AssociationsWithName AssociationValidator.AssociationsWithName
class	AssociationValidator.AssociationsWithNoMultiplicity AssociationValidator.AssociationsWithNoMultiplicity
class	AssociationValidator.AssociationsWithRoleBadDirection AssociationValidator.AssociationsWithRoleBadDirection
class	AssociationValidator.AssociationsWithSameDocOnBothEnds AssociationValidator.AssociationsWithSameDocOnBothEnds
class	AssociationValidator.AssociationsWithUnallowedStereotype AssociationValidator.AssociationsWithUnallowedStereotype

class	AssociationValidator.AssociationsWithUnallowedTagNames AssociationValidator.AssociationsWithUnallowedTagNames
class	AssociationValidator.AssociationsWithWrongSource AssociationValidator.AssociationsWithWrongSource
class	AssociationValidator.CimAssociationEndsNameShouldBePlural AssociationValidator.CimAssociationEndsNameShouldBePlural
class	AssociationValidator.CimAssociationEndsNameShouldBeSingular AssociationValidator.CimAssociationEndsNameShouldBeSingular
class	AssociationValidator.CimAssociationEndsNameStartingWithLowerCase AssociationValidator.CimAssociationEndsNameStartingWithLowerCase
class	AssociationValidator.Iec61850AssociationsThatShouldBePrivate AssociationValidator.Iec61850AssociationsThatShouldBePrivate
class	AssociationValidator.Iec61850AssociationsWithDifferentEndVisibility AssociationValidator.Iec61850AssociationsWithDifferentEndVisibility

Method Summary

boolean	enabled()
java.util.List	getScopedUmlObjects()

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractValidator

addCrossRule, addSimpleRule, displayAllAvailableRuleNames, displayAvailableRuleNames, enabled, getAllCrossRules, getAllRules, getAllSimpleRules, getCfg, getCheckedCrossRules, getCheckedRules, getCheckedSimpleRules, getCollectedIssues, getScopedUmlObjects, validate

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods

enabled

public boolean **enabled**()

Returns whether the validation for this validator has been enabled (by configuration).

getScopedUmlObjects

public java.util.List **getScopedUmlObjects**()

Returns the elements retained for validation, for the configured scope.

org.tanjakostic.jcleancim.validation
Class AssociationValidator.AssociationsWithExplicitDirection

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.validation.AbstractRule
      |
      +-
```
org.tanjakostic.jcleancim.validation.AssociationValidator.AssociationsWithExplicitDirection

All Implemented Interfaces:
SimpleRule, Rule

---

public static class **AssociationValidator.AssociationsWithExplicitDirection**
extends AbstractRule
implements Rule, SimpleRule

Constructor Summary

public	AssociationsWithExplicitDirection()

Method Summary

java.util.EnumSet	getApplicability()
void	validate(UmlAssociation o, ModelIssues issues)

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule

createIssue, createIssue, createIssue, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from interface org.tanjakostic.jcleancim.validation.SimpleRule

validate

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Constructors

AssociationsWithExplicitDirection

```
public AssociationsWithExplicitDirection()
```

Methods

getApplicability

```
public java.util.EnumSet getApplicability()
```

validate

```
public void validate(UmlAssociation o,
         ModelIssues issues)
```

org.tanjakostic.jcleancim.validation
Class AssociationValidator.AssociationsWithRoleBadDirection

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.validation.AbstractRule
      |
      +-
```
org.tanjakostic.jcleancim.validation.AssociationValidator.AssociationsWithRoleBadDirection

All Implemented Interfaces:
SimpleRule, Rule

---

public static class **AssociationValidator.AssociationsWithRoleBadDirection**
extends AbstractRule
implements Rule, SimpleRule

Constructor Summary

public	AssociationsWithRoleBadDirection()

Method Summary

java.util.EnumSet	getApplicability()
void	validate(UmlAssociation o, ModelIssues issues)

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule

createIssue, createIssue, createIssue, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from interface org.tanjakostic.jcleancim.validation.SimpleRule

validate

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Constructors

AssociationsWithRoleBadDirection

```
public AssociationsWithRoleBadDirection()
```

Methods

getApplicability

```
public java.util.EnumSet getApplicability()
```

validate

```
public void validate(UmlAssociation o,
         ModelIssues issues)
```

org.tanjakostic.jcleancim.validation
Class AssociationValidator.AssociationsWithDoc

```
java.lang.Object
  |-org.tanjakostic.jcleancim.validation.AbstractRule
    |-org.tanjakostic.jcleancim.validation.AssociationValidator.AssociationsWithDoc
```

All Implemented Interfaces:
SimpleRule, Rule

---

public static class **AssociationValidator.AssociationsWithDoc**
extends AbstractRule
implements Rule, SimpleRule

Constructor Summary

public	AssociationsWithDoc()

Method Summary

java.util.EnumSet	getApplicability()
void	validate(UmlAssociation o, ModelIssues issues)

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule

createIssue, createIssue, createIssue, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from interface org.tanjakostic.jcleancim.validation.SimpleRule

validate

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

---

(continued from last page)

Constructors

AssociationsWithDoc

```
public AssociationsWithDoc()
```

Methods

getApplicability

```
public java.util.EnumSet getApplicability()
```

---

validate

```
public void validate(UmlAssociation o,
         ModelIssues issues)
```

org.tanjakostic.jcleancim.validation
Class AssociationValidator.AssociationsWithSameDocOnBothEnds

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.validation.AbstractRule
      |
      +-
```
org.tanjakostic.jcleancim.validation.AssociationValidator.AssociationsWithSameDocOnBothEnds

All Implemented Interfaces:
SimpleRule, Rule

---

public static class **AssociationValidator.AssociationsWithSameDocOnBothEnds**
extends AbstractRule
implements Rule, SimpleRule

Constructor Summary

public	AssociationsWithSameDocOnBothEnds()

Method Summary

java.util.EnumSet	getApplicability()
void	validate(UmlAssociation o, ModelIssues issues)

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule

createIssue, createIssue, createIssue, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from interface org.tanjakostic.jcleancim.validation.SimpleRule

validate

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Constructors

AssociationsWithSameDocOnBothEnds

```
public AssociationsWithSameDocOnBothEnds()
```

Methods

getApplicability

```
public java.util.EnumSet getApplicability()
```

validate

```
public void validate(UmlAssociation o,
         ModelIssues issues)
```

org.tanjakostic.jcleancim.validation
Class AssociationValidator.AssociationsWithName

```
java.lang.Object
  |-org.tanjakostic.jcleancim.validation.AbstractRule
    |-org.tanjakostic.jcleancim.validation.AssociationValidator.AssociationsWithName
```

All Implemented Interfaces:
SimpleRule, Rule

public static class **AssociationValidator.AssociationsWithName**
extends AbstractRule
implements Rule, SimpleRule

Constructor Summary

public	AssociationsWithName()

Method Summary

java.util.EnumSet	getApplicability()
void	validate(UmlAssociation o, ModelIssues issues)

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule

createIssue, createIssue, createIssue, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from interface org.tanjakostic.jcleancim.validation.SimpleRule

validate

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

(continued from last page)

Constructors

AssociationsWithName

```
public AssociationsWithName()
```

Methods

getApplicability

```
public java.util.EnumSet getApplicability()
```

---

validate

```
public void validate(UmlAssociation o,
         ModelIssues issues)
```

org.tanjakostic.jcleancim.validation
Class AssociationValidator.AssociationsWithUnallowedStereotype

```
java.lang.Object
   |
   +-org.tanjakostic.jcleancim.validation.AbstractRule
      |
      +-
org.tanjakostic.jcleancim.validation.AbstractRule.AbstractRuleWithSubobjectsAndSkips
         |
         +-
org.tanjakostic.jcleancim.validation.AbstractRule.UmlObjectsWithUnallowedStereotype
            |
            +-
```
org.tanjakostic.jcleancim.validation.AssociationValidator.AssociationsWithUnallowedStereotype

All Implemented Interfaces:
Rule, SimpleRule

---

public static class **AssociationValidator.AssociationsWithUnallowedStereotype**
extends AbstractRule.UmlObjectsWithUnallowedStereotype

Constructor Summary

public	AssociationsWithUnallowedStereotype()

Method Summary

java.util.EnumSet	getApplicability()

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule.UmlObjectsWithUnallowedStereotype

doValidate

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule.AbstractRuleWithSubobjectsAndSkips

doValidate, getSubObjects, skipSubobjectValidation, skipValidation, validate

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule

createIssue, createIssue, createIssue, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from interface `org.tanjakostic.jcleancim.validation.SimpleRule`
`validate`

Methods inherited from interface `org.tanjakostic.jcleancim.validation.Rule`
`getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis`

Constructors

AssociationsWithUnallowedStereotype

```
public AssociationsWithUnallowedStereotype()
```

Methods

getApplicability

```
public java.util.EnumSet getApplicability()
```

org.tanjakostic.jcleancim.validation
Class AssociationValidator.AssociationEndsWithUnallowedStereotype

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.validation.AbstractRule
    |
    +-org.tanjakostic.jcleancim.validation.AbstractRule.AbstractRuleWithSubobjectsAndSkips
      |
      +-org.tanjakostic.jcleancim.validation.AbstractRule.UmlObjectsWithUnallowedStereotype
        |
        +-org.tanjakostic.jcleancim.validation.AssociationValidator.AssociationEndsWithUnallowedStereotype
```

All Implemented Interfaces:
Rule, SimpleRule

---

public static class **AssociationValidator.AssociationEndsWithUnallowedStereotype**
extends AbstractRule.UmlObjectsWithUnallowedStereotype

Constructor Summary

public	AssociationEndsWithUnallowedStereotype()

Method Summary

java.util.EnumSet	getApplicability()
java.util.List	getSubObjects(UmlAssociation o)

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule.UmlObjectsWithUnallowedStereotype

doValidate

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule.AbstractRuleWithSubobjectsAndSkips

doValidate, getSubObjects, skipSubobjectValidation, skipValidation, validate

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule

createIssue, createIssue, createIssue, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from interface org.tanjakostic.jcleancim.validation.SimpleRule
validate

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule
getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Constructors

AssociationEndsWithUnallowedStereotype

public **AssociationEndsWithUnallowedStereotype**()

Methods

getApplicability

public java.util.EnumSet **getApplicability**()

getSubObjects

protected java.util.List **getSubObjects**(UmlAssociation o)

Used for validation of "sub-objects" (such as association ends and operation parameters) that do not have their own validators, but get validated through their container.

This default implementation returns empty list; subtypes having contained objects that need to be validated against this rule have to override this method.

org.tanjakostic.jcleancim.validation
Class AssociationValidator.AssociationsMissingInformativeStereotype

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.validation.AbstractRule
      |
      +-
```
org.tanjakostic.jcleancim.validation.AssociationValidator.AssociationsMissingInformativeStereotype

All Implemented Interfaces:
SimpleRule, Rule

---

public static class **AssociationValidator.AssociationsMissingInformativeStereotype**
extends AbstractRule
implements Rule, SimpleRule

Constructor Summary

public	AssociationsMissingInformativeStereotype()

Method Summary

java.util.EnumSet	getApplicability()
void	validate(UmlAssociation o, ModelIssues issues)

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule

createIssue, createIssue, createIssue, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from interface org.tanjakostic.jcleancim.validation.SimpleRule

validate

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Constructors

AssociationsMissingInformativeStereotype

```
public AssociationsMissingInformativeStereotype()
```

Methods

getApplicability

```
public java.util.EnumSet getApplicability()
```

validate

```
public void validate(UmlAssociation o,
         ModelIssues issues)
```

org.tanjakostic.jcleancim.validation
Class AssociationValidator.AssociationsWithUnallowedTagNames

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.validation.AbstractRule
      |
      +-
org.tanjakostic.jcleancim.validation.AbstractRule.AbstractRuleWithSubobjectsAndSkips
          |
          +-
org.tanjakostic.jcleancim.validation.AbstractRule.UmlObjectsWithUnallowedTagNames
              |
              +-
```
org.tanjakostic.jcleancim.validation.AssociationValidator.AssociationsWithUnallowedTagNames

All Implemented Interfaces:
Rule, SimpleRule

---

public static class **AssociationValidator.AssociationsWithUnallowedTagNames**
extends AbstractRule.UmlObjectsWithUnallowedTagNames

Constructor Summary

public	AssociationsWithUnallowedTagNames()

Method Summary

java.util.EnumSet	getApplicability()

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule.UmlObjectsWithUnallowedTagNames

doValidate

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule.AbstractRuleWithSubobjectsAndSkips

doValidate, getSubObjects, skipSubobjectValidation, skipValidation, validate

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule

createIssue, createIssue, createIssue, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from interface `org.tanjakostic.jcleancim.validation.SimpleRule`
`validate`

Methods inherited from interface `org.tanjakostic.jcleancim.validation.Rule`
`getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis`

Constructors

AssociationsWithUnallowedTagNames

```
public AssociationsWithUnallowedTagNames()
```

Methods

getApplicability

```
public java.util.EnumSet getApplicability()
```

org.tanjakostic.jcleancim.validation
Class AssociationValidator.AssociationEndsWithUnallowedTagNames

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.validation.AbstractRule
    |
    +-
org.tanjakostic.jcleancim.validation.AbstractRule.AbstractRuleWithSubobjectsAndSkips
      |
      +-
org.tanjakostic.jcleancim.validation.AbstractRule.UmlObjectsWithUnallowedTagNames
        |
        +-
```
org.tanjakostic.jcleancim.validation.AssociationValidator.AssociationEndsWithUnallowedTagNames

All Implemented Interfaces:
Rule, SimpleRule

---

public static class **AssociationValidator.AssociationEndsWithUnallowedTagNames**
extends AbstractRule.UmlObjectsWithUnallowedTagNames

Constructor Summary

public	AssociationEndsWithUnallowedTagNames()

Method Summary

java.util.EnumSet	getApplicability()

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule.UmlObjectsWithUnallowedTagNames

doValidate

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule.AbstractRuleWithSubobjectsAndSkips

doValidate, getSubObjects, skipSubobjectValidation, skipValidation, validate

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule

createIssue, createIssue, createIssue, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from interface `org.tanjakostic.jcleancim.validation.SimpleRule`
`validate`

Methods inherited from interface `org.tanjakostic.jcleancim.validation.Rule`
`getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis`

Constructors

AssociationEndsWithUnallowedTagNames

`public` **`AssociationEndsWithUnallowedTagNames`**`()`

Methods

getApplicability

`public java.util.EnumSet` **`getApplicability`**`()`

org.tanjakostic.jcleancim.validation
Class AssociationValidator.AssociationsWithNoMultiplicity

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.validation.AbstractRule
      |
      +-
```
org.tanjakostic.jcleancim.validation.AssociationValidator.AssociationsWithNoMultiplicity

All Implemented Interfaces:
SimpleRule, Rule

---

public static class **AssociationValidator.AssociationsWithNoMultiplicity**
extends AbstractRule
implements Rule, SimpleRule

Constructor Summary

public	AssociationsWithNoMultiplicity()

Method Summary

java.util.EnumSet	getApplicability()
void	validate(UmlAssociation o, ModelIssues issues)

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule

createIssue, createIssue, createIssue, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from interface org.tanjakostic.jcleancim.validation.SimpleRule

validate

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Constructors

AssociationsWithNoMultiplicity

```
public AssociationsWithNoMultiplicity()
```

Methods

getApplicability

```
public java.util.EnumSet getApplicability()
```

validate

```
public void validate(UmlAssociation o,
         ModelIssues issues)
```

org.tanjakostic.jcleancim.validation
Class AssociationValidator.AssociationsWithWrongSource

```
java.lang.Object
  |-org.tanjakostic.jcleancim.validation.AbstractRule
    |-
```
org.tanjakostic.jcleancim.validation.AssociationValidator.AssociationsWithWrongSource

All Implemented Interfaces:
SimpleRule, Rule

---

public static class **AssociationValidator.AssociationsWithWrongSource**
extends AbstractRule
implements Rule, SimpleRule

Constructor Summary

public	AssociationsWithWrongSource()

Method Summary

java.util.EnumSet	getApplicability()
void	validate(UmlAssociation o, ModelIssues issues)

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule

createIssue, createIssue, createIssue, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from interface org.tanjakostic.jcleancim.validation.SimpleRule

validate

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

---

(continued from last page)

Constructors

AssociationsWithWrongSource

```
public AssociationsWithWrongSource()
```

Methods

getApplicability

```
public java.util.EnumSet getApplicability()
```

---

validate

```
public void validate(UmlAssociation o,
         ModelIssues issues)
```

org.tanjakostic.jcleancim.validation
Class AssociationValidator.Iec61850AssociationsThatShouldBePrivate

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.validation.AbstractRule
      |
      +-
```
org.tanjakostic.jcleancim.validation.AssociationValidator.Iec61850AssociationsThatShouldBePrivate

All Implemented Interfaces:
SimpleRule, Rule

---

public static class **AssociationValidator.Iec61850AssociationsThatShouldBePrivate**
extends AbstractRule
implements Rule, SimpleRule

Field Summary

public static final	VALID_PUBLIC_ASSOC Package SCL describes an XSD, where it is ok to have public associations, so we exclude it from that validation. Value: **SCL**

Constructor Summary

public	Iec61850AssociationsThatShouldBePrivate()

Method Summary

java.util.EnumSet	getApplicability()
void	validate(UmlAssociation o, ModelIssues issues)

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule

createIssue, createIssue, createIssue, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from interface org.tanjakostic.jcleancim.validation.SimpleRule

validate

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule
getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Fields

VALID_PUBLIC_ASSOC

```
public static final java.lang.String VALID_PUBLIC_ASSOC
```

Package SCL describes an XSD, where it is ok to have public associations, so we exclude it from that validation.
Constant value: **SCL**

Constructors

Iec61850AssociationsThatShouldBePrivate

```
public Iec61850AssociationsThatShouldBePrivate()
```

Methods

getApplicability

```
public java.util.EnumSet getApplicability()
```

validate

```
public void validate(UmlAssociation o,
         ModelIssues issues)
```

org.tanjakostic.jcleancim.validation
Class AssociationValidator.Iec61850AssociationsWithDifferentEndVisibility

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.validation.AbstractRule
      |
      +-
```
org.tanjakostic.jcleancim.validation.AssociationValidator.Iec61850AssociationsWithDifferentEndVisibility

All Implemented Interfaces:
SimpleRule, Rule

---

public static class **AssociationValidator.Iec61850AssociationsWithDifferentEndVisibility**
extends AbstractRule
implements Rule, SimpleRule

Constructor Summary

public	Iec61850AssociationsWithDifferentEndVisibility()

Method Summary

java.util.EnumSet	getApplicability()
void	validate(UmlAssociation o, ModelIssues issues)

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule

createIssue, createIssue, createIssue, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from interface org.tanjakostic.jcleancim.validation.SimpleRule

validate

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Constructors

Iec61850AssociationsWithDifferentEndVisibility

```
public Iec61850AssociationsWithDifferentEndVisibility()
```

Methods

getApplicability

```
public java.util.EnumSet getApplicability()
```

validate

```
public void validate(UmlAssociation o,
         ModelIssues issues)
```

org.tanjakostic.jcleancim.validation
Class AssociationValidator.AssociationEndsMissingDoc

```
java.lang.Object
  |-org.tanjakostic.jcleancim.validation.AbstractRule
    |-
org.tanjakostic.jcleancim.validation.AbstractRule.AbstractRuleWithSubobjectsAndSkips
      |-org.tanjakostic.jcleancim.validation.AbstractRule.UmlObjectsMissingDoc
        |-
org.tanjakostic.jcleancim.validation.AssociationValidator.AssociationEndsMissingDoc
```

All Implemented Interfaces:
Rule, SimpleRule

---

public static class **AssociationValidator.AssociationEndsMissingDoc**
extends AbstractRule.UmlObjectsMissingDoc

Constructor Summary

public	AssociationEndsMissingDoc()

Method Summary

java.util.EnumSet	getApplicability()
java.util.List	getSubObjects(UmlAssociation assoc)
boolean	skipValidation(UmlAssociation o)

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule.UmlObjectsMissingDoc

doValidate

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule.AbstractRuleWithSubobjectsAndSkips

doValidate, getSubObjects, skipSubobjectValidation, skipValidation, validate

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule

createIssue, createIssue, createIssue, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from interface org.tanjakostic.jcleancim.validation.SimpleRule
validate

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule
getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Constructors

AssociationEndsMissingDoc

public **AssociationEndsMissingDoc**()

Methods

getApplicability

public java.util.EnumSet **getApplicability**()

skipValidation

protected boolean **skipValidation**(UmlAssociation o)

This default implementation returns false (no skipping); override if main object doesn't need validation.

getSubObjects

protected final java.util.List **getSubObjects**(UmlAssociation assoc)

Used for validation of "sub-objects" (such as association ends and operation parameters) that do not have their own validators, but get validated through their container.

This default implementation returns empty list; subtypes having contained objects that need to be validated against this rule have to override this method.

org.tanjakostic.jcleancim.validation
Class AssociationValidator.AssociationEndsWithBadDocStart

```
java.lang.Object
  |-org.tanjakostic.jcleancim.validation.AbstractRule
    |-
org.tanjakostic.jcleancim.validation.AbstractRule.AbstractRuleWithSubobjectsAndSkips
      |-org.tanjakostic.jcleancim.validation.AbstractRule.UmlObjectsWithBadDocStart
        |-
org.tanjakostic.jcleancim.validation.AssociationValidator.AssociationEndsWithBadDocStart
```

All Implemented Interfaces:
Rule, SimpleRule

---

public static class **AssociationValidator.AssociationEndsWithBadDocStart**
extends AbstractRule.UmlObjectsWithBadDocStart

Constructor Summary

public	AssociationEndsWithBadDocStart()

Method Summary

java.util.EnumSet	getApplicability()
java.util.List	getSubObjects(UmlAssociation assoc)

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule.UmlObjectsWithBadDocStart

doValidate

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule.AbstractRuleWithSubobjectsAndSkips

doValidate, getSubObjects, skipSubobjectValidation, skipValidation, validate

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule

createIssue, createIssue, createIssue, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from interface org.tanjakostic.jcleancim.validation.SimpleRule
validate

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule
getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Constructors

AssociationEndsWithBadDocStart

public **AssociationEndsWithBadDocStart**()

Methods

getApplicability

public java.util.EnumSet **getApplicability**()

getSubObjects

protected final java.util.List **getSubObjects**(UmlAssociation assoc)

Used for validation of "sub-objects" (such as association ends and operation parameters) that do not have their own validators, but get validated through their container.

This default implementation returns empty list; subtypes having contained objects that need to be validated against this rule have to override this method.

org.tanjakostic.jcleancim.validation
Class AssociationValidator.AssociationEndsWithBadDocEnd

```
java.lang.Object
  |-org.tanjakostic.jcleancim.validation.AbstractRule
    |-
org.tanjakostic.jcleancim.validation.AbstractRule.AbstractRuleWithSubobjectsAndSkips
      |-org.tanjakostic.jcleancim.validation.AbstractRule.UmlObjectsWithBadDocEnd
        |-
org.tanjakostic.jcleancim.validation.AssociationValidator.AssociationEndsWithBadDocEnd
```

All Implemented Interfaces:
Rule, SimpleRule

---

public static class **AssociationValidator.AssociationEndsWithBadDocEnd**
extends AbstractRule.UmlObjectsWithBadDocEnd

Constructor Summary

public	AssociationEndsWithBadDocEnd()

Method Summary

java.util.EnumSet	getApplicability()
java.util.List	getSubObjects(UmlAssociation assoc)

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule.UmlObjectsWithBadDocEnd

doValidate

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule.AbstractRuleWithSubobjectsAndSkips

doValidate, getSubObjects, skipSubobjectValidation, skipValidation, validate

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule

createIssue, createIssue, createIssue, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from interface org.tanjakostic.jcleancim.validation.SimpleRule
validate

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule
getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Constructors

AssociationEndsWithBadDocEnd

public **AssociationEndsWithBadDocEnd**()

Methods

getApplicability

public java.util.EnumSet **getApplicability**()

getSubObjects

protected final java.util.List **getSubObjects**(UmlAssociation assoc)

Used for validation of "sub-objects" (such as association ends and operation parameters) that do not have their own validators, but get validated through their container.

This default implementation returns empty list; subtypes having contained objects that need to be validated against this rule have to override this method.

org.tanjakostic.jcleancim.validation
Class AssociationValidator.AssociationEndsWithBadCharacterInName

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.validation.AbstractRule
      |
      +-
org.tanjakostic.jcleancim.validation.AbstractRule.AbstractRuleWithSubobjectsAndSkips
          |
          +-
org.tanjakostic.jcleancim.validation.AbstractRule.UmlObjectsWithBadCharacterInName
              |
              +-
```
org.tanjakostic.jcleancim.validation.AssociationValidator.AssociationEndsWithBadCharacterInName

All Implemented Interfaces:
Rule, SimpleRule

---

public static class **AssociationValidator.AssociationEndsWithBadCharacterInName**
extends AbstractRule.UmlObjectsWithBadCharacterInName

Constructor Summary

public	AssociationEndsWithBadCharacterInName()

Method Summary

java.util.EnumSet	getApplicability()
InvalidCharactersFinder	getInvalidCharacterFinder(UmlObject o)
java.util.List	getSubObjects(UmlAssociation assoc)

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule.UmlObjectsWithBadCharacterInName

doValidate, getInvalidCharacterFinder

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule.AbstractRuleWithSubobjectsAndSkips

doValidate, getSubObjects, skipSubobjectValidation, skipValidation, validate

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule

createIssue, createIssue, createIssue, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface `org.tanjakostic.jcleancim.validation.Rule`
`getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis`

Methods inherited from interface `org.tanjakostic.jcleancim.validation.SimpleRule`
`validate`

Methods inherited from interface `org.tanjakostic.jcleancim.validation.Rule`
`getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis`

Constructors

AssociationEndsWithBadCharacterInName

`public` **`AssociationEndsWithBadCharacterInName`**`()`

Methods

getApplicability

`public java.util.EnumSet` **`getApplicability`**`()`

getSubObjects

`protected java.util.List` **`getSubObjects`**`(UmlAssociation assoc)`

Used for validation of "sub-objects" (such as association ends and operation parameters) that do not have their own validators, but get validated through their container.

This default implementation returns empty list; subtypes having contained objects that need to be validated against this rule have to override this method.

getInvalidCharacterFinder

`protected InvalidCharactersFinder` **`getInvalidCharacterFinder`**`(UmlObject o)`

org.tanjakostic.jcleancim.validation
Class
AssociationValidator.CimAssociationEndsNameStartingWithLowerCase

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.validation.AbstractRule
      |
      +-
```
org.tanjakostic.jcleancim.validation.AssociationValidator.CimAssociationEndsNameStartingWithLowerCase

All Implemented Interfaces:
SimpleRule, Rule

---

public static class **AssociationValidator.CimAssociationEndsNameStartingWithLowerCase**
extends AbstractRule
implements Rule, SimpleRule

Constructor Summary

public	CimAssociationEndsNameStartingWithLowerCase()

Method Summary

java.util.EnumSet	getApplicability()
void	validate(UmlAssociation ae, ModelIssues issues)

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule

createIssue, createIssue, createIssue, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from interface org.tanjakostic.jcleancim.validation.SimpleRule

validate

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Constructors

CimAssociationEndsNameStartingWithLowerCase

```
public CimAssociationEndsNameStartingWithLowerCase()
```

Methods

getApplicability

```
public java.util.EnumSet getApplicability()
```

validate

```
public void validate(UmlAssociation ae,
         ModelIssues issues)
```

org.tanjakostic.jcleancim.validation
Class AssociationValidator.CimAssociationEndsNameShouldBePlural

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.validation.AbstractRule
      |
      +-
```
org.tanjakostic.jcleancim.validation.AssociationValidator.CimAssociationEndsNameShouldBePlural

All Implemented Interfaces:
SimpleRule, Rule

---

public static class **AssociationValidator.CimAssociationEndsNameShouldBePlural**
extends AbstractRule
implements Rule, SimpleRule

Constructor Summary

public	CimAssociationEndsNameShouldBePlural()

Method Summary

java.util.EnumSet	getApplicability()
void	validate(UmlAssociation ae, ModelIssues issues)

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule

createIssue, createIssue, createIssue, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from interface org.tanjakostic.jcleancim.validation.SimpleRule

validate

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Constructors

CimAssociationEndsNameShouldBePlural

```
public CimAssociationEndsNameShouldBePlural()
```

Methods

getApplicability

```
public java.util.EnumSet getApplicability()
```

validate

```
public void validate(UmlAssociation ae,
         ModelIssues issues)
```

org.tanjakostic.jcleancim.validation
Class AssociationValidator.CimAssociationEndsNameShouldBeSingular

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.validation.AbstractRule
      |
      +-
```
org.tanjakostic.jcleancim.validation.AssociationValidator.CimAssociationEndsNameShouldBeSingular

All Implemented Interfaces:
SimpleRule, Rule

---

public static class **AssociationValidator.CimAssociationEndsNameShouldBeSingular**
extends AbstractRule
implements Rule, SimpleRule

Constructor Summary

public	CimAssociationEndsNameShouldBeSingular()

Method Summary

java.util.EnumSet	getApplicability()
void	validate(UmlAssociation ae, ModelIssues issues)

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule

createIssue, createIssue, createIssue, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from interface org.tanjakostic.jcleancim.validation.SimpleRule

validate

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Constructors

CimAssociationEndsNameShouldBeSingular

```
public CimAssociationEndsNameShouldBeSingular()
```

Methods

getApplicability

```
public java.util.EnumSet getApplicability()
```

validate

```
public void validate(UmlAssociation ae,
         ModelIssues issues)
```

org.tanjakostic.jcleancim.validation
Class AttributeValidator

```
java.lang.Object
  |-org.tanjakostic.jcleancim.validation.AbstractValidator
    |-org.tanjakostic.jcleancim.validation.AttributeValidator
```

public class **AttributeValidator**
extends AbstractValidator

Validates attributes.

Nested Class Summary

class	AttributeValidator.AttributesMissingDoc AttributeValidator.AttributesMissingDoc
class	AttributeValidator.AttributesThatAreConstNonStatic AttributeValidator.AttributesThatAreConstNonStatic
class	AttributeValidator.AttributesThatAreEnumsInNonEnumeratedClass AttributeValidator.AttributesThatAreEnumsInNonEnumeratedClass
class	AttributeValidator.AttributesThatAreStaticButNotConst AttributeValidator.AttributesThatAreStaticButNotConst
class	AttributeValidator.AttributesWhoseTypeIsInformative AttributeValidator.AttributesWhoseTypeIsInformative
class	AttributeValidator.AttributesWithBadDocEnd AttributeValidator.AttributesWithBadDocEnd
class	AttributeValidator.AttributesWithBadDocStart AttributeValidator.AttributesWithBadDocStart
class	AttributeValidator.AttributesWithInexistingEnumLiteralAsInitValue AttributeValidator.AttributesWithInexistingEnumLiteralAsInitValue
class	AttributeValidator.AttributesWithInvalidMultiplicity AttributeValidator.AttributesWithInvalidMultiplicity
class	AttributeValidator.AttributesWithInvalidTypeNull AttributeValidator.AttributesWithInvalidTypeNull
class	AttributeValidator.AttributesWithInvalidTypeString AttributeValidator.AttributesWithInvalidTypeString
class	AttributeValidator.AttributesWithTypeFromUnallowedOwner AttributeValidator.AttributesWithTypeFromUnallowedOwner
class	AttributeValidator.AttributesWithTypeIdMismatch AttributeValidator.AttributesWithTypeIdMismatch
class	AttributeValidator.AttributesWithUnallowedStereotype AttributeValidator.AttributesWithUnallowedStereotype

class	AttributeValidator.AttributesWithUnallowedTagNames AttributeValidator.AttributesWithUnallowedTagNames
class	AttributeValidator.CimAttributesNameShouldBeSingular AttributeValidator.CimAttributesNameShouldBeSingular
class	AttributeValidator.CimAttributesNameShouldNotStartWithClassName AttributeValidator.CimAttributesNameShouldNotStartWithClassName
class	AttributeValidator.CimAttributesNameStartingWithUpperCase AttributeValidator.CimAttributesNameStartingWithUpperCase
class	AttributeValidator.CimAttributesThatAreNotStaticNonConstWithInitVal AttributeValidator.CimAttributesThatAreNotStaticNonConstWithInitVal
class	AttributeValidator.CimAttributesThatShouldBeOptional AttributeValidator.CimAttributesThatShouldBeOptional
class	AttributeValidator.CimAttributesThatShouldBePublic AttributeValidator.CimAttributesThatShouldBePublic
class	AttributeValidator.CimAttributesThatShouldBeReplacedWithAssociation AttributeValidator.CimAttributesThatShouldBeReplacedWithAssociation
class	AttributeValidator.CimAttributesWithBadCharacterInName AttributeValidator.CimAttributesWithBadCharacterInName
class	AttributeValidator.CimAttributesWithFlagInName AttributeValidator.CimAttributesWithFlagInName
class	AttributeValidator.EnumLiteralsWithoutEnumStereotype AttributeValidator.EnumLiteralsWithoutEnumStereotype
class	AttributeValidator.EnumLiteralsWithSuperfluousType AttributeValidator.EnumLiteralsWithSuperfluousType
class	AttributeValidator.Iec61850AbbreviationLiteralsNameStartingWithLowerCase AttributeValidator.Iec61850AbbreviationLiteralsNameStartingWithLowerCase
class	AttributeValidator.Iec61850AttributesWithBadCharacterInName AttributeValidator.Iec61850AttributesWithBadCharacterInName
class	AttributeValidator.Iec61850AttributesWithInexistingSibling AttributeValidator.Iec61850AttributesWithInexistingSibling
class	AttributeValidator.Iec61850ConditionLiteralsNeverUsedAsConstraints AttributeValidator.Iec61850ConditionLiteralsNeverUsedAsConstraints
class	AttributeValidator.Iec61850DOAbbreviationLiteralsDuplicateDescription AttributeValidator.Iec61850DOAbbreviationLiteralsDuplicateDescription
class	AttributeValidator.Iec61850DOAbbreviationLiteralsDuplicateName AttributeValidator.Iec61850DOAbbreviationLiteralsDuplicateName
class	AttributeValidator.Iec61850DOAbbreviationLiteralsNeverUsedInDOName AttributeValidator.Iec61850DOAbbreviationLiteralsNeverUsedInDOName
class	AttributeValidator.Iec61850DOAttributesNameStartingWithLowerCase AttributeValidator.Iec61850DOAttributesNameStartingWithLowerCase

class	AttributeValidator.Iec61850DOAttributesWithNameMissingAbbreviation AttributeValidator.Iec61850DOAttributesWithNameMissingAbbreviation
class	AttributeValidator.Iec61850DOAttributesWithSameNameDifferentType AttributeValidator.Iec61850DOAttributesWithSameNameDifferentType
class	AttributeValidator.Iec61850DOAttributesWithTooLongName AttributeValidator.Iec61850DOAttributesWithTooLongName
class	AttributeValidator.Iec61850FCDAAttributesWithMissingConstraint AttributeValidator.Iec61850FCDAAttributesWithMissingConstraint

Method Summary

boolean	enabled()
java.util.List	getScopedUmlObjects()

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractValidator

addCrossRule, addSimpleRule, displayAllAvailableRuleNames, displayAvailableRuleNames, enabled, getAllCrossRules, getAllRules, getAllSimpleRules, getCfg, getCheckedCrossRules, getCheckedRules, getCheckedSimpleRules, getCollectedIssues, getScopedUmlObjects, validate

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods

enabled

public boolean **enabled**()

Returns whether the validation for this validator has been enabled (by configuration).

getScopedUmlObjects

public java.util.List **getScopedUmlObjects**()

Returns the elements retained for validation, for the configured scope.

org.tanjakostic.jcleancim.validation
Class AttributeValidator.EnumLiteralsWithSuperfluousType

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.validation.AbstractRule
      |
      +-
```
org.tanjakostic.jcleancim.validation.AttributeValidator.EnumLiteralsWithSuperfluousType

All Implemented Interfaces:
SimpleRule, Rule

---

public static class **AttributeValidator.EnumLiteralsWithSuperfluousType**
extends AbstractRule
implements Rule, SimpleRule

Constructor Summary

public	EnumLiteralsWithSuperfluousType()

Method Summary

java.util.EnumSet	getApplicability()
void	validate(UmlAttribute o, ModelIssues issues)

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule

createIssue, createIssue, createIssue, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from interface org.tanjakostic.jcleancim.validation.SimpleRule

validate

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

---

(continued from last page)

Constructors

EnumLiteralsWithSuperfluousType

```
public EnumLiteralsWithSuperfluousType()
```

Methods

getApplicability

```
public java.util.EnumSet getApplicability()
```

---

validate

```
public void validate(UmlAttribute o,
         ModelIssues issues)
```

org.tanjakostic.jcleancim.validation
Class AttributeValidator.EnumLiteralsWithoutEnumStereotype

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.validation.AbstractRule
      |
      +-
```
org.tanjakostic.jcleancim.validation.AttributeValidator.EnumLiteralsWithoutEnumStereotype

All Implemented Interfaces:
SimpleRule, Rule

---

public static class **AttributeValidator.EnumLiteralsWithoutEnumStereotype**
extends AbstractRule
implements Rule, SimpleRule

Constructor Summary

public	EnumLiteralsWithoutEnumStereotype()

Method Summary

java.util.EnumSet	getApplicability()
void	validate(UmlAttribute o, ModelIssues issues)

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule

createIssue, createIssue, createIssue, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from interface org.tanjakostic.jcleancim.validation.SimpleRule

validate

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Constructors

EnumLiteralsWithoutEnumStereotype

```
public EnumLiteralsWithoutEnumStereotype()
```

Methods

getApplicability

```
public java.util.EnumSet getApplicability()
```

validate

```
public void validate(UmlAttribute o,
         ModelIssues issues)
```

org.tanjakostic.jcleancim.validation
Class AttributeValidator.AttributesWithInvalidMultiplicity

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.validation.AbstractRule
      |
      +-
```
org.tanjakostic.jcleancim.validation.AttributeValidator.AttributesWithInvalidMultiplicity

All Implemented Interfaces:
SimpleRule, Rule

---

public static class **AttributeValidator.AttributesWithInvalidMultiplicity**
extends AbstractRule
implements Rule, SimpleRule

Constructor Summary

public	AttributesWithInvalidMultiplicity()

Method Summary

java.util.EnumSet	getApplicability()
void	validate(UmlAttribute o, ModelIssues issues)

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule

createIssue, createIssue, createIssue, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from interface org.tanjakostic.jcleancim.validation.SimpleRule

validate

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Constructors

AttributesWithInvalidMultiplicity

```
public AttributesWithInvalidMultiplicity()
```

Methods

getApplicability

```
public java.util.EnumSet getApplicability()
```

validate

```
public void validate(UmlAttribute o,
         ModelIssues issues)
```

org.tanjakostic.jcleancim.validation
Class AttributeValidator.CimAttributesThatShouldBeOptional

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.validation.AbstractRule
      |
      +-
```
org.tanjakostic.jcleancim.validation.AttributeValidator.CimAttributesThatShouldBeOptional

All Implemented Interfaces:
SimpleRule, Rule

---

public static class **AttributeValidator.CimAttributesThatShouldBeOptional**
extends AbstractRule
implements Rule, SimpleRule

Constructor Summary

public	CimAttributesThatShouldBeOptional()

Method Summary

java.util.EnumSet	getApplicability()
void	validate(UmlAttribute o, ModelIssues issues)

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule

createIssue, createIssue, createIssue, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from interface org.tanjakostic.jcleancim.validation.SimpleRule

validate

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Constructors

CimAttributesThatShouldBeOptional

```
public CimAttributesThatShouldBeOptional()
```

Methods

getApplicability

```
public java.util.EnumSet getApplicability()
```

validate

```
public void validate(UmlAttribute o,
         ModelIssues issues)
```

org.tanjakostic.jcleancim.validation
Class AttributeValidator.AttributesWithInvalidTypeNull

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.validation.AbstractRule
      |
      +-
```
org.tanjakostic.jcleancim.validation.AttributeValidator.AttributesWithInvalidTypeNull

All Implemented Interfaces:
SimpleRule, Rule

---

public static class **AttributeValidator.AttributesWithInvalidTypeNull**
extends AbstractRule
implements Rule, SimpleRule

Constructor Summary

public	AttributesWithInvalidTypeNull()

Method Summary

java.util.EnumSet	getApplicability()
void	validate(UmlAttribute o, ModelIssues issues)

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule

createIssue, createIssue, createIssue, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from interface org.tanjakostic.jcleancim.validation.SimpleRule

validate

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

---

(continued from last page)

Constructors

AttributesWithInvalidTypeNull

```
public AttributesWithInvalidTypeNull()
```

Methods

getApplicability

```
public java.util.EnumSet getApplicability()
```

---

validate

```
public void validate(UmlAttribute o,
         ModelIssues issues)
```

org.tanjakostic.jcleancim.validation
Class AttributeValidator.AttributesWithInvalidTypeString

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.validation.AbstractRule
      |
      +-
```
org.tanjakostic.jcleancim.validation.AttributeValidator.AttributesWithInvalidTypeString

All Implemented Interfaces:
SimpleRule, Rule

---

public static class **AttributeValidator.AttributesWithInvalidTypeString**
extends AbstractRule
implements Rule, SimpleRule

Constructor Summary

public	AttributesWithInvalidTypeString()

Method Summary

java.util.EnumSet	getApplicability()
void	validate(UmlAttribute o, ModelIssues issues)

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule

createIssue, createIssue, createIssue, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from interface org.tanjakostic.jcleancim.validation.SimpleRule

validate

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

---

(continued from last page)

Constructors

AttributesWithInvalidTypeString

```
public AttributesWithInvalidTypeString()
```

Methods

getApplicability

```
public java.util.EnumSet getApplicability()
```

---

validate

```
public void validate(UmlAttribute o,
         ModelIssues issues)
```

org.tanjakostic.jcleancim.validation
Class AttributeValidator.AttributesWithTypeIdMismatch

```
java.lang.Object
  |-org.tanjakostic.jcleancim.validation.AbstractRule
    |-
```
org.tanjakostic.jcleancim.validation.AttributeValidator.AttributesWithTypeIdMismatch

All Implemented Interfaces:
SimpleRule, Rule

---

public static class **AttributeValidator.AttributesWithTypeIdMismatch**
extends AbstractRule
implements Rule, SimpleRule

Constructor Summary

public	AttributesWithTypeIdMismatch()

Method Summary

java.util.EnumSet	getApplicability()
void	validate(UmlAttribute o, ModelIssues issues)

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule

createIssue, createIssue, createIssue, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from interface org.tanjakostic.jcleancim.validation.SimpleRule

validate

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

---

(continued from last page)

Constructors

AttributesWithTypeIdMismatch

```
public AttributesWithTypeIdMismatch()
```

Methods

getApplicability

```
public java.util.EnumSet getApplicability()
```

---

validate

```
public void validate(UmlAttribute o,
         ModelIssues issues)
```

org.tanjakostic.jcleancim.validation
Class AttributeValidator.CimAttributesThatShouldBePublic

```
java.lang.Object
  |-org.tanjakostic.jcleancim.validation.AbstractRule
    |-
```
org.tanjakostic.jcleancim.validation.AttributeValidator.CimAttributesThatShouldBePublic

All Implemented Interfaces:
SimpleRule, Rule

---

public static class **AttributeValidator.CimAttributesThatShouldBePublic**
extends AbstractRule
implements Rule, SimpleRule

Constructor Summary

public	CimAttributesThatShouldBePublic()

Method Summary

java.util.EnumSet	getApplicability()
void	validate(UmlAttribute o, ModelIssues issues)

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule

createIssue, createIssue, createIssue, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from interface org.tanjakostic.jcleancim.validation.SimpleRule

validate

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

---

(continued from last page)

Constructors

CimAttributesThatShouldBePublic

```
public CimAttributesThatShouldBePublic()
```

Methods

getApplicability

```
public java.util.EnumSet getApplicability()
```

---

validate

```
public void validate(UmlAttribute o,
         ModelIssues issues)
```

org.tanjakostic.jcleancim.validation
Class AttributeValidator.AttributesThatAreStaticButNotConst

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.validation.AbstractRule
      |
      +-
```
org.tanjakostic.jcleancim.validation.AttributeValidator.AttributesThatAreStaticButNotConst

All Implemented Interfaces:
SimpleRule, Rule

---

public static class **AttributeValidator.AttributesThatAreStaticButNotConst**
extends AbstractRule
implements Rule, SimpleRule

Field Summary

public static final	EXCLUDE_VALID_INITVAL_PCKG Package SCL describes an XSD, where it is ok to have default initial value, so we exclude it from that validation. Value: **SCL**

Constructor Summary

public	AttributesThatAreStaticButNotConst()

Method Summary

java.util.EnumSet	getApplicability()
void	validate(UmlAttribute o, ModelIssues issues)

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule

createIssue, createIssue, createIssue, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from interface org.tanjakostic.jcleancim.validation.SimpleRule

validate

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule
getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Fields

EXCLUDE_VALID_INITVAL_PCKG

```
public static final java.lang.String EXCLUDE_VALID_INITVAL_PCKG
```

Package SCL describes an XSD, where it is ok to have default initial value, so we exclude it from that validation.
Constant value: **SCL**

Constructors

AttributesThatAreStaticButNotConst

```
public AttributesThatAreStaticButNotConst()
```

Methods

getApplicability

```
public java.util.EnumSet getApplicability()
```

validate

```
public void validate(UmlAttribute o,
         ModelIssues issues)
```

org.tanjakostic.jcleancim.validation
Class
AttributeValidator.CimAttributesThatAreNotStaticNonConstWithInitVal

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.validation.AbstractRule
      |
      +-
```
org.tanjakostic.jcleancim.validation.AttributeValidator.CimAttributesThatAreNotStaticNonConstWithInitVal

All Implemented Interfaces:
SimpleRule, Rule

---

public static class **AttributeValidator.CimAttributesThatAreNotStaticNonConstWithInitVal**
extends AbstractRule
implements Rule, SimpleRule

Constructor Summary

public	CimAttributesThatAreNotStaticNonConstWithInitVal()

Method Summary

java.util.EnumSet	getApplicability()
void	validate(UmlAttribute o, ModelIssues issues)

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule

createIssue, createIssue, createIssue, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from interface org.tanjakostic.jcleancim.validation.SimpleRule

validate

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Constructors

CimAttributesThatAreNotStaticNonConstWithInitVal

```
public CimAttributesThatAreNotStaticNonConstWithInitVal()
```

Methods

getApplicability

```
public java.util.EnumSet getApplicability()
```

validate

```
public void validate(UmlAttribute o,
         ModelIssues issues)
```

org.tanjakostic.jcleancim.validation
Class AttributeValidator.AttributesThatAreConstNonStatic

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.validation.AbstractRule
      |
      +-
```
org.tanjakostic.jcleancim.validation.AttributeValidator.AttributesThatAreConstNonStatic

All Implemented Interfaces:
SimpleRule, Rule

---

public static class **AttributeValidator.AttributesThatAreConstNonStatic**
extends AbstractRule
implements Rule, SimpleRule

Constructor Summary

public	AttributesThatAreConstNonStatic()

Method Summary

java.util.EnumSet	getApplicability()
void	validate(UmlAttribute o, ModelIssues issues)

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule

createIssue, createIssue, createIssue, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from interface org.tanjakostic.jcleancim.validation.SimpleRule

validate

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

---

(continued from last page)

Constructors

AttributesThatAreConstNonStatic

```
public AttributesThatAreConstNonStatic()
```

Methods

getApplicability

```
public java.util.EnumSet getApplicability()
```

---

validate

```
public void validate(UmlAttribute o,
         ModelIssues issues)
```

org.tanjakostic.jcleancim.validation
Class AttributeValidator.AttributesWithUnallowedStereotype

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.validation.AbstractRule
      |
      +-
org.tanjakostic.jcleancim.validation.AbstractRule.AbstractRuleWithSubobjectsAndSkips
          |
          +-
org.tanjakostic.jcleancim.validation.AbstractRule.UmlObjectsWithUnallowedStereotype
              |
              +-
```
org.tanjakostic.jcleancim.validation.AttributeValidator.AttributesWithUnallowedStereotype

All Implemented Interfaces:
Rule, SimpleRule

---

public static class **AttributeValidator.AttributesWithUnallowedStereotype**
extends AbstractRule.UmlObjectsWithUnallowedStereotype

Constructor Summary

public	AttributesWithUnallowedStereotype()

Method Summary

java.util.EnumSet	getApplicability()

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule.UmlObjectsWithUnallowedStereotype

doValidate

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule.AbstractRuleWithSubobjectsAndSkips

doValidate, getSubObjects, skipSubobjectValidation, skipValidation, validate

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule

createIssue, createIssue, createIssue, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from interface `org.tanjakostic.jcleancim.validation.SimpleRule`
`validate`

Methods inherited from interface `org.tanjakostic.jcleancim.validation.Rule`
`getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis`

Constructors

AttributesWithUnallowedStereotype

`public` **`AttributesWithUnallowedStereotype`**`()`

Methods

getApplicability

`public java.util.EnumSet` **`getApplicability`**`()`

org.tanjakostic.jcleancim.validation
Class AttributeValidator.AttributesThatAreEnumsInNonEnumeratedClass

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.validation.AbstractRule
      |
      +-
```
org.tanjakostic.jcleancim.validation.AttributeValidator.AttributesThatAreEnumsInNonEnumeratedClass

All Implemented Interfaces:
SimpleRule, Rule

---

public static class **AttributeValidator.AttributesThatAreEnumsInNonEnumeratedClass**
extends AbstractRule
implements Rule, SimpleRule

Constructor Summary

public	AttributesThatAreEnumsInNonEnumeratedClass()

Method Summary

java.util.EnumSet	getApplicability()
void	validate(UmlAttribute o, ModelIssues issues)

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule

createIssue, createIssue, createIssue, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from interface org.tanjakostic.jcleancim.validation.SimpleRule

validate

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Constructors

AttributesThatAreEnumsInNonEnumeratedClass

```
public AttributesThatAreEnumsInNonEnumeratedClass()
```

Methods

getApplicability

```
public java.util.EnumSet getApplicability()
```

validate

```
public void validate(UmlAttribute o,
         ModelIssues issues)
```

org.tanjakostic.jcleancim.validation
Class
AttributeValidator.CimAttributesThatShouldBeReplacedWithAssociation

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.validation.AbstractRule
      |
      +-
```
org.tanjakostic.jcleancim.validation.AttributeValidator.CimAttributesThatShouldBeReplacedWithAssociation

All Implemented Interfaces:
SimpleRule, Rule

---

public static class **AttributeValidator.CimAttributesThatShouldBeReplacedWithAssociation**
extends AbstractRule
implements Rule, SimpleRule

Constructor Summary

public	CimAttributesThatShouldBeReplacedWithAssociation()

Method Summary

java.util.EnumSet	getApplicability()
void	validate(UmlAttribute o, ModelIssues issues)

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule

createIssue, createIssue, createIssue, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from interface org.tanjakostic.jcleancim.validation.SimpleRule

validate

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Constructors

CimAttributesThatShouldBeReplacedWithAssociation

```
public CimAttributesThatShouldBeReplacedWithAssociation()
```

Methods

getApplicability

```
public java.util.EnumSet getApplicability()
```

validate

```
public void validate(UmlAttribute o,
         ModelIssues issues)
```

org.tanjakostic.jcleancim.validation
Class AttributeValidator.AttributesWhoseTypeIsInformative

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.validation.AbstractRule
      |
      +-
```
org.tanjakostic.jcleancim.validation.AttributeValidator.AttributesWhoseTypeIsInformative

All Implemented Interfaces:
SimpleRule, Rule

---

public static class **AttributeValidator.AttributesWhoseTypeIsInformative**
extends AbstractRule
implements Rule, SimpleRule

Constructor Summary

public	AttributesWhoseTypeIsInformative()

Method Summary

java.util.EnumSet	getApplicability()
void	validate(UmlAttribute o, ModelIssues issues)

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule

createIssue, createIssue, createIssue, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from interface org.tanjakostic.jcleancim.validation.SimpleRule

validate

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Constructors

AttributesWhoseTypeIsInformative

```
public AttributesWhoseTypeIsInformative()
```

Methods

getApplicability

```
public java.util.EnumSet getApplicability()
```

validate

```
public void validate(UmlAttribute o,
         ModelIssues issues)
```

org.tanjakostic.jcleancim.validation
Class AttributeValidator.AttributesWithUnallowedTagNames

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.validation.AbstractRule
      |
      +-
org.tanjakostic.jcleancim.validation.AbstractRule.AbstractRuleWithSubobjectsAndSkips
          |
          +-
org.tanjakostic.jcleancim.validation.AbstractRule.UmlObjectsWithUnallowedTagNames
              |
              +-
```

org.tanjakostic.jcleancim.validation.AttributeValidator.AttributesWithUnallowedTagNames

All Implemented Interfaces:
Rule, SimpleRule

---

public static class **AttributeValidator.AttributesWithUnallowedTagNames**
extends AbstractRule.UmlObjectsWithUnallowedTagNames

Constructor Summary

public	AttributesWithUnallowedTagNames()

Method Summary

java.util.EnumSet	getApplicability()

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule.UmlObjectsWithUnallowedTagNames

doValidate

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule.AbstractRuleWithSubobjectsAndSkips

doValidate, getSubObjects, skipSubobjectValidation, skipValidation, validate

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule

createIssue, createIssue, createIssue, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from interface `org.tanjakostic.jcleancim.validation.SimpleRule`
`validate`

Methods inherited from interface `org.tanjakostic.jcleancim.validation.Rule`
`getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis`

Constructors

AttributesWithUnallowedTagNames

`public` **`AttributesWithUnallowedTagNames`**`()`

Methods

getApplicability

`public java.util.EnumSet` **`getApplicability`**`()`

org.tanjakostic.jcleancim.validation
Class AttributeValidator.Iec61850AttributesWithInexistingSibling

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.validation.AbstractRule
      |
      +-
```
org.tanjakostic.jcleancim.validation.AttributeValidator.Iec61850AttributesWithInexistingSibling

All Implemented Interfaces:
SimpleRule, Rule

---

public static class **AttributeValidator.Iec61850AttributesWithInexistingSibling**
extends AbstractRule
implements Rule, SimpleRule

Constructor Summary

public	Iec61850AttributesWithInexistingSibling()

Method Summary

java.util.EnumSet	getApplicability()
void	validate(UmlAttribute o, ModelIssues issues)

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule

createIssue, createIssue, createIssue, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from interface org.tanjakostic.jcleancim.validation.SimpleRule

validate

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Constructors

Iec61850AttributesWithInexistingSibling

```
public Iec61850AttributesWithInexistingSibling()
```

Methods

getApplicability

```
public java.util.EnumSet getApplicability()
```

validate

```
public void validate(UmlAttribute o,
         ModelIssues issues)
```

org.tanjakostic.jcleancim.validation
Class AttributeValidator.CimAttributesWithFlagInName

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.validation.AbstractRule
      |
      +-
```
org.tanjakostic.jcleancim.validation.AttributeValidator.CimAttributesWithFlagInName

All Implemented Interfaces:
SimpleRule, Rule

---

public static class **AttributeValidator.CimAttributesWithFlagInName**
extends AbstractRule
implements Rule, SimpleRule

Field Summary

public static final	FLAG Value: **flag**

Constructor Summary

public	CimAttributesWithFlagInName()

Method Summary

java.util.EnumSet	getApplicability()
void	validate(UmlAttribute o, ModelIssues issues)

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule

createIssue, createIssue, createIssue, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from interface org.tanjakostic.jcleancim.validation.SimpleRule

validate

Methods inherited from interface `org.tanjakostic.jcleancim.validation.Rule`
`getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis`

Fields

FLAG

```
public static final java.lang.String FLAG
```

Constant value: **`flag`**

Constructors

CimAttributesWithFlagInName

```
public CimAttributesWithFlagInName()
```

Methods

getApplicability

```
public java.util.EnumSet getApplicability()
```

validate

```
public void validate(UmlAttribute o,
         ModelIssues issues)
```

org.tanjakostic.jcleancim.validation
Class AttributeValidator.AttributesMissingDoc

```
java.lang.Object
  |-org.tanjakostic.jcleancim.validation.AbstractRule
    |-
org.tanjakostic.jcleancim.validation.AbstractRule.AbstractRuleWithSubobjectsAndSkips
      |-org.tanjakostic.jcleancim.validation.AbstractRule.UmlObjectsMissingDoc
        |-
org.tanjakostic.jcleancim.validation.AttributeValidator.AttributesMissingDoc
```

All Implemented Interfaces:
Rule, SimpleRule

---

public static class **AttributeValidator.AttributesMissingDoc**
extends AbstractRule.UmlObjectsMissingDoc

Constructor Summary

public	AttributesMissingDoc()

Method Summary

java.util.EnumSet	getApplicability()
boolean	skipValidation(UmlAttribute o)

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule.UmlObjectsMissingDoc

doValidate

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule.AbstractRuleWithSubobjectsAndSkips

doValidate, getSubObjects, skipSubobjectValidation, skipValidation, validate

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule

createIssue, createIssue, createIssue, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from interface org.tanjakostic.jcleancim.validation.SimpleRule
validate

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule
getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Constructors

AttributesMissingDoc

public **AttributesMissingDoc**()

Methods

getApplicability

public java.util.EnumSet **getApplicability**()

skipValidation

protected boolean **skipValidation**(UmlAttribute o)

This default implementation returns false (no skipping); override if main object doesn't need validation.

org.tanjakostic.jcleancim.validation
Class AttributeValidator.AttributesWithBadDocStart

```
java.lang.Object
  |-org.tanjakostic.jcleancim.validation.AbstractRule
    |-
org.tanjakostic.jcleancim.validation.AbstractRule.AbstractRuleWithSubobjectsAndSkips
      |-org.tanjakostic.jcleancim.validation.AbstractRule.UmlObjectsWithBadDocStart
        |-
org.tanjakostic.jcleancim.validation.AttributeValidator.AttributesWithBadDocStart
```

All Implemented Interfaces:
Rule, SimpleRule

---

public static class **AttributeValidator.AttributesWithBadDocStart**
extends AbstractRule.UmlObjectsWithBadDocStart

Constructor Summary

public	AttributesWithBadDocStart()

Method Summary

java.util.EnumSet	getApplicability()
boolean	skipValidation(UmlAttribute o)

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule.UmlObjectsWithBadDocStart

doValidate

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule.AbstractRuleWithSubobjectsAndSkips

doValidate, getSubObjects, skipSubobjectValidation, skipValidation, validate

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule

createIssue, createIssue, createIssue, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from interface org.tanjakostic.jcleancim.validation.SimpleRule
validate

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule
getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Constructors

AttributesWithBadDocStart

public **AttributesWithBadDocStart**()

Methods

getApplicability

public java.util.EnumSet **getApplicability**()

skipValidation

protected boolean **skipValidation**(UmlAttribute o)

This default implementation returns false (no skipping); override if main object doesn't need validation.

org.tanjakostic.jcleancim.validation
Class AttributeValidator.AttributesWithBadDocEnd

```
java.lang.Object
  |-org.tanjakostic.jcleancim.validation.AbstractRule
    |-
org.tanjakostic.jcleancim.validation.AbstractRule.AbstractRuleWithSubobjectsAndSkips
      |-org.tanjakostic.jcleancim.validation.AbstractRule.UmlObjectsWithBadDocEnd
        |-
org.tanjakostic.jcleancim.validation.AttributeValidator.AttributesWithBadDocEnd
```

All Implemented Interfaces:
Rule, SimpleRule

---

public static class **AttributeValidator.AttributesWithBadDocEnd**
extends AbstractRule.UmlObjectsWithBadDocEnd

Constructor Summary

public	AttributesWithBadDocEnd()

Method Summary

java.util.EnumSet	getApplicability()
boolean	skipValidation(UmlAttribute o)

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule.UmlObjectsWithBadDocEnd

doValidate

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule.AbstractRuleWithSubobjectsAndSkips

doValidate, getSubObjects, skipSubobjectValidation, skipValidation, validate

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule

createIssue, createIssue, createIssue, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from interface org.tanjakostic.jcleancim.validation.SimpleRule
validate

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule
getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Constructors

AttributesWithBadDocEnd

public **AttributesWithBadDocEnd**()

Methods

getApplicability

public java.util.EnumSet **getApplicability**()

skipValidation

protected boolean **skipValidation**(UmlAttribute o)

This default implementation returns false (no skipping); override if main object doesn't need validation.

org.tanjakostic.jcleancim.validation
Class AttributeValidator.CimAttributesWithBadCharacterInName

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.validation.AbstractRule
      |
      +-
org.tanjakostic.jcleancim.validation.AbstractRule.AbstractRuleWithSubobjectsAndSkips
          |
          +-
org.tanjakostic.jcleancim.validation.AbstractRule.UmlObjectsWithBadCharacterInName
              |
              +-
```
org.tanjakostic.jcleancim.validation.AttributeValidator.CimAttributesWithBadCharacterInName

All Implemented Interfaces:
Rule, SimpleRule

---

public static class **AttributeValidator.CimAttributesWithBadCharacterInName**
extends AbstractRule.UmlObjectsWithBadCharacterInName

Constructor Summary

public	CimAttributesWithBadCharacterInName()

Method Summary

java.util.EnumSet	getApplicability()
InvalidCharactersFinder	getInvalidCharacterFinder(UmlObject o)
boolean	skipValidation(UmlAttribute o)

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule.UmlObjectsWithBadCharacterInName

doValidate, getInvalidCharacterFinder

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule.AbstractRuleWithSubobjectsAndSkips

doValidate, getSubObjects, skipSubobjectValidation, skipValidation, validate

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule

createIssue, createIssue, createIssue, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface `org.tanjakostic.jcleancim.validation.Rule`
`getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis`

Methods inherited from interface `org.tanjakostic.jcleancim.validation.SimpleRule`
`validate`

Methods inherited from interface `org.tanjakostic.jcleancim.validation.Rule`
`getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis`

Constructors

CimAttributesWithBadCharacterInName

`public` **`CimAttributesWithBadCharacterInName`**`()`

Methods

getApplicability

`public java.util.EnumSet` **`getApplicability`**`()`

skipValidation

`protected boolean` **`skipValidation`**`(UmlAttribute o)`

This default implementation returns false (no skipping); override if main object doesn't need validation.

getInvalidCharacterFinder

`protected InvalidCharactersFinder` **`getInvalidCharacterFinder`**`(UmlObject o)`

org.tanjakostic.jcleancim.validation
Class AttributeValidator.Iec61850AttributesWithBadCharacterInName

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.validation.AbstractRule
      |
      +-
org.tanjakostic.jcleancim.validation.AbstractRule.AbstractRuleWithSubobjectsAndSkips
          |
          +-
org.tanjakostic.jcleancim.validation.AbstractRule.UmlObjectsWithBadCharacterInName
              |
              +-
```
org.tanjakostic.jcleancim.validation.AttributeValidator.Iec61850AttributesWithBadCharacterInName

All Implemented Interfaces:
Rule, SimpleRule

---

public static class **AttributeValidator.Iec61850AttributesWithBadCharacterInName**
extends AbstractRule.UmlObjectsWithBadCharacterInName

Constructor Summary

public	Iec61850AttributesWithBadCharacterInName()

Method Summary

java.util.EnumSet	getApplicability()
InvalidCharactersFinder	getInvalidCharacterFinder(UmlObject o)
boolean	skipValidation(UmlAttribute o)

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule.UmlObjectsWithBadCharacterInName

doValidate, getInvalidCharacterFinder

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule.AbstractRuleWithSubobjectsAndSkips

doValidate, getSubObjects, skipSubobjectValidation, skipValidation, validate

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule

createIssue, createIssue, createIssue, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface `org.tanjakostic.jcleancim.validation.Rule`
`getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis`

Methods inherited from interface `org.tanjakostic.jcleancim.validation.SimpleRule`
`validate`

Methods inherited from interface `org.tanjakostic.jcleancim.validation.Rule`
`getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis`

Constructors

Iec61850AttributesWithBadCharacterInName

```
public Iec61850AttributesWithBadCharacterInName()
```

Methods

getApplicability

```
public java.util.EnumSet getApplicability()
```

skipValidation

```
protected boolean skipValidation(UmlAttribute o)
```

This default implementation returns false (no skipping); override if main object doesn't need validation.

getInvalidCharacterFinder

```
protected InvalidCharactersFinder getInvalidCharacterFinder(UmlObject o)
```

org.tanjakostic.jcleancim.validation
Class AttributeValidator.Iec61850DOAttributesWithTooLongName

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.validation.AbstractRule
      |
      +-
```
org.tanjakostic.jcleancim.validation.AttributeValidator.Iec61850DOAttributesWithTooLongName

All Implemented Interfaces:
SimpleRule, Rule

---

public static class **AttributeValidator.Iec61850DOAttributesWithTooLongName**
extends AbstractRule
implements Rule, SimpleRule

Constructor Summary

public	Iec61850DOAttributesWithTooLongName()

Method Summary

java.util.EnumSet	getApplicability()
void	validate(UmlAttribute o, ModelIssues issues)

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule

createIssue, createIssue, createIssue, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from interface org.tanjakostic.jcleancim.validation.SimpleRule

validate

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Constructors

Iec61850DOAttributesWithTooLongName

```
public Iec61850DOAttributesWithTooLongName()
```

Methods

getApplicability

```
public java.util.EnumSet getApplicability()
```

validate

```
public void validate(UmlAttribute o,
         ModelIssues issues)
```

org.tanjakostic.jcleancim.validation
Class AttributeValidator.Iec61850FCDAAttributesWithMissingConstraint

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.validation.AbstractRule
      |
      +-
```
org.tanjakostic.jcleancim.validation.AttributeValidator.Iec61850FCDAAttributesWithMissingConstraint

All Implemented Interfaces:
SimpleRule, Rule

---

public static class **AttributeValidator.Iec61850FCDAAttributesWithMissingConstraint**
extends AbstractRule
implements Rule, SimpleRule

Constructor Summary

public	Iec61850FCDAAttributesWithMissingConstraint()

Method Summary

java.util.EnumSet	getApplicability()
void	validate(UmlAttribute o, ModelIssues issues)

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule

createIssue, createIssue, createIssue, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from interface org.tanjakostic.jcleancim.validation.SimpleRule

validate

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Constructors

Iec61850FCDAAttributesWithMissingConstraint

```
public Iec61850FCDAAttributesWithMissingConstraint()
```

Methods

getApplicability

```
public java.util.EnumSet getApplicability()
```

validate

```
public void validate(UmlAttribute o,
         ModelIssues issues)
```

org.tanjakostic.jcleancim.validation
Class AttributeValidator.AttributesWithInexistingEnumLiteralAsInitValue

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.validation.AbstractRule
      |
      +-
```
org.tanjakostic.jcleancim.validation.AttributeValidator.AttributesWithInexistingEnumLiteralAsInitValue

All Implemented Interfaces:
SimpleRule, Rule

---

public static class **AttributeValidator.AttributesWithInexistingEnumLiteralAsInitValue**
extends AbstractRule
implements Rule, SimpleRule

Constructor Summary

public	AttributesWithInexistingEnumLiteralAsInitValue()

Method Summary

java.util.EnumSet	getApplicability()
void	validate(UmlAttribute o, ModelIssues issues)

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule

createIssue, createIssue, createIssue, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from interface org.tanjakostic.jcleancim.validation.SimpleRule

validate

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Constructors

AttributesWithInexistingEnumLiteralAsInitValue

```
public AttributesWithInexistingEnumLiteralAsInitValue()
```

Methods

getApplicability

```
public java.util.EnumSet getApplicability()
```

validate

```
public void validate(UmlAttribute o,
         ModelIssues issues)
```

org.tanjakostic.jcleancim.validation
Class
AttributeValidator.Iec61850DOAttributesWithNameMissingAbbreviation

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.validation.AbstractRule
      |
      +-
```
org.tanjakostic.jcleancim.validation.AttributeValidator.Iec61850DOAttributesWithNameMissingAbbreviation

All Implemented Interfaces:
SimpleRule, Rule

---

public static class **AttributeValidator.Iec61850DOAttributesWithNameMissingAbbreviation**
extends AbstractRule
implements Rule, SimpleRule

Constructor Summary

public	Iec61850DOAttributesWithNameMissingAbbreviation()

Method Summary

java.util.EnumSet	getApplicability()
void	validate(UmlAttribute o, ModelIssues issues)

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule

createIssue, createIssue, createIssue, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from interface org.tanjakostic.jcleancim.validation.SimpleRule

validate

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Constructors

Iec61850DOAttributesWithNameMissingAbbreviation

```
public Iec61850DOAttributesWithNameMissingAbbreviation()
```

Methods

getApplicability

```
public java.util.EnumSet getApplicability()
```

validate

```
public void validate(UmlAttribute o,
         ModelIssues issues)
```

org.tanjakostic.jcleancim.validation
Class AttributeValidator.CimAttributesNameStartingWithUpperCase

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.validation.AbstractRule
      |
      +-
```
org.tanjakostic.jcleancim.validation.AttributeValidator.CimAttributesNameStartingWithUpperCase

All Implemented Interfaces:
SimpleRule, Rule

---

public static class **AttributeValidator.CimAttributesNameStartingWithUpperCase**
extends AbstractRule
implements Rule, SimpleRule

Constructor Summary

public	CimAttributesNameStartingWithUpperCase()

Method Summary

java.util.EnumSet	getApplicability()
void	validate(UmlAttribute o, ModelIssues issues)

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule

createIssue, createIssue, createIssue, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from interface org.tanjakostic.jcleancim.validation.SimpleRule

validate

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Constructors

CimAttributesNameStartingWithUpperCase

```
public CimAttributesNameStartingWithUpperCase()
```

Methods

getApplicability

```
public java.util.EnumSet getApplicability()
```

validate

```
public void validate(UmlAttribute o,
         ModelIssues issues)
```

org.tanjakostic.jcleancim.validation
Class AttributeValidator.CimAttributesNameShouldBeSingular

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.validation.AbstractRule
      |
      +-
```
org.tanjakostic.jcleancim.validation.AttributeValidator.CimAttributesNameShouldBeSingular

All Implemented Interfaces:
SimpleRule, Rule

---

public static class **AttributeValidator.CimAttributesNameShouldBeSingular**
extends AbstractRule
implements Rule, SimpleRule

Constructor Summary

public	CimAttributesNameShouldBeSingular()

Method Summary

java.util.EnumSet	getApplicability()
void	validate(UmlAttribute o, ModelIssues issues)

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule

createIssue, createIssue, createIssue, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from interface org.tanjakostic.jcleancim.validation.SimpleRule

validate

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Constructors

CimAttributesNameShouldBeSingular

```
public CimAttributesNameShouldBeSingular()
```

Methods

getApplicability

```
public java.util.EnumSet getApplicability()
```

validate

```
public void validate(UmlAttribute o,
         ModelIssues issues)
```

org.tanjakostic.jcleancim.validation
Class
AttributeValidator.CimAttributesNameShouldNotStartWithClassName

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.validation.AbstractRule
      |
      +-
```
org.tanjakostic.jcleancim.validation.AttributeValidator.CimAttributesNameShouldNotStartWithClassName

All Implemented Interfaces:
SimpleRule, Rule

---

public static class **AttributeValidator.CimAttributesNameShouldNotStartWithClassName**
extends AbstractRule
implements Rule, SimpleRule

Constructor Summary

public	CimAttributesNameShouldNotStartWithClassName()

Method Summary

java.util.EnumSet	getApplicability()
void	validate(UmlAttribute o, ModelIssues issues)

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule

createIssue, createIssue, createIssue, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from interface org.tanjakostic.jcleancim.validation.SimpleRule

validate

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Constructors

CimAttributesNameShouldNotStartWithClassName

```
public CimAttributesNameShouldNotStartWithClassName()
```

Methods

getApplicability

```
public java.util.EnumSet getApplicability()
```

validate

```
public void validate(UmlAttribute o,
         ModelIssues issues)
```

org.tanjakostic.jcleancim.validation
Class AttributeValidator.Iec61850AbbreviationLiteralsNameStartingWithLowerCase

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.validation.AbstractRule
      |
      +-
```
org.tanjakostic.jcleancim.validation.AttributeValidator.Iec61850AbbreviationLiteralsNameStartingWithLowerCase

All Implemented Interfaces:
SimpleRule, Rule

---

public static class **AttributeValidator.Iec61850AbbreviationLiteralsNameStartingWithLowerCase**
extends AbstractRule
implements Rule, SimpleRule

Constructor Summary

public	Iec61850AbbreviationLiteralsNameStartingWithLowerCase()

Method Summary

java.util.EnumSet	getApplicability()
void	validate(UmlAttribute o, ModelIssues issues)

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule

createIssue, createIssue, createIssue, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from interface org.tanjakostic.jcleancim.validation.SimpleRule

validate

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

```
getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity,
logDiagnosis
```

Constructors

Iec61850AbbreviationLiteralsNameStartingWithLowerCase

```
public Iec61850AbbreviationLiteralsNameStartingWithLowerCase()
```

Methods

getApplicability

```
public java.util.EnumSet getApplicability()
```

validate

```
public void validate(UmlAttribute o,
         ModelIssues issues)
```

org.tanjakostic.jcleancim.validation
Class
AttributeValidator.Iec61850DOAttributesNameStartingWithLowerCase

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.validation.AbstractRule
      |
      +-
```
org.tanjakostic.jcleancim.validation.AttributeValidator.Iec61850DOAttributesNameStartingWithLowerCase

All Implemented Interfaces:
SimpleRule, Rule

---

public static class **AttributeValidator.Iec61850DOAttributesNameStartingWithLowerCase**
extends AbstractRule
implements Rule, SimpleRule

Constructor Summary

public	Iec61850DOAttributesNameStartingWithLowerCase()

Method Summary

java.util.EnumSet	getApplicability()
void	validate(UmlAttribute o, ModelIssues issues)

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule

createIssue, createIssue, createIssue, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from interface org.tanjakostic.jcleancim.validation.SimpleRule

validate

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Constructors

Iec61850DOAttributesNameStartingWithLowerCase

```
public Iec61850DOAttributesNameStartingWithLowerCase()
```

Methods

getApplicability

```
public java.util.EnumSet getApplicability()
```

validate

```
public void validate(UmlAttribute o,
         ModelIssues issues)
```

org.tanjakostic.jcleancim.validation
Class AttributeValidator.AttributesWithTypeFromUnallowedOwner

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.validation.AbstractRule
      |
      +-
```
org.tanjakostic.jcleancim.validation.AttributeValidator.AttributesWithTypeFromUnallowedOwner

All Implemented Interfaces:
SimpleRule, Rule

---

public static class **AttributeValidator.AttributesWithTypeFromUnallowedOwner**
extends AbstractRule
implements Rule, SimpleRule

Constructor Summary

public	AttributesWithTypeFromUnallowedOwner()

Method Summary

java.util.EnumSet	getApplicability()
void	validate(UmlAttribute o, ModelIssues issues)

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule

createIssue, createIssue, createIssue, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from interface org.tanjakostic.jcleancim.validation.SimpleRule

validate

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Constructors

AttributesWithTypeFromUnallowedOwner

```
public AttributesWithTypeFromUnallowedOwner()
```

Methods

getApplicability

```
public java.util.EnumSet getApplicability()
```

validate

```
public void validate(UmlAttribute o,
         ModelIssues issues)
```

org.tanjakostic.jcleancim.validation
Class AttributeValidator.Iec61850DOAbbreviationLiteralsDuplicateName

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.validation.AbstractRule
      |
      +-
```
org.tanjakostic.jcleancim.validation.AttributeValidator.Iec61850DOAbbreviationLiteralsDuplicateName

All Implemented Interfaces:
CrossRule, Rule

---

public static class **AttributeValidator.Iec61850DOAbbreviationLiteralsDuplicateName**
extends AbstractRule
implements Rule, CrossRule

Constructor Summary

public	Iec61850DOAbbreviationLiteralsDuplicateName(java.util.Collection allAbbrLiterals)

Method Summary

java.util.EnumSet	getApplicability()
java.util.Collection	getObjsToTestAgainst()
void	validate(java.util.List attributes, ModelIssues issues)

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule

createIssue, createIssue, createIssue, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from interface org.tanjakostic.jcleancim.validation.CrossRule

getObjsToTestAgainst, validate

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

```
getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity,
logDiagnosis
```

Constructors

Iec61850DOAbbreviationLiteralsDuplicateName

```
public Iec61850DOAbbreviationLiteralsDuplicateName(java.util.Collection
allAbbrLiterals)
```

Methods

getApplicability

```
public java.util.EnumSet getApplicability()
```

getObjsToTestAgainst

```
public java.util.Collection getObjsToTestAgainst()
```

validate

```
public void validate(java.util.List attributes,
         ModelIssues issues)
```

org.tanjakostic.jcleancim.validation
Class
AttributeValidator.Iec61850DOAbbreviationLiteralsDuplicateDescription

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.validation.AbstractRule
      |
      +-
```
org.tanjakostic.jcleancim.validation.AttributeValidator.Iec61850DOAbbreviationLiteralsDuplicateDescription

All Implemented Interfaces:
CrossRule, Rule

---

public static class **AttributeValidator.Iec61850DOAbbreviationLiteralsDuplicateDescription**
extends AbstractRule
implements Rule, CrossRule

Constructor Summary

public	Iec61850DOAbbreviationLiteralsDuplicateDescription(java.util.Collection allAbbrLiterals)

Method Summary

java.util.EnumSet	getApplicability()
java.util.Collection	getObjsToTestAgainst()
void	validate(java.util.List attributes, ModelIssues issues)

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule

createIssue, createIssue, createIssue, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from interface org.tanjakostic.jcleancim.validation.CrossRule

getObjsToTestAgainst, validate

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

```
getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity,
logDiagnosis
```

Constructors

Iec61850DOAbbreviationLiteralsDuplicateDescription

```
public Iec61850DOAbbreviationLiteralsDuplicateDescription(java.util.Collection
allAbbrLiterals)
```

Methods

getApplicability

```
public java.util.EnumSet getApplicability()
```

getObjsToTestAgainst

```
public java.util.Collection getObjsToTestAgainst()
```

validate

```
public void validate(java.util.List attributes,
         ModelIssues issues)
```

org.tanjakostic.jcleancim.validation
Class
AttributeValidator.Iec61850DOAbbreviationLiteralsNeverUsedInDOName

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.validation.AbstractRule
      |
      +-
```
org.tanjakostic.jcleancim.validation.AttributeValidator.Iec61850DOAbbreviationLiteralsNeverUsedInDOName

All Implemented Interfaces:
CrossRule, Rule

---

public static class **AttributeValidator.Iec61850DOAbbreviationLiteralsNeverUsedInDOName**
extends AbstractRule
implements Rule, CrossRule

Constructor Summary

public	Iec61850DOAbbreviationLiteralsNeverUsedInDOName(java.util.Collection allAbbrLiterals)

Method Summary

java.util.EnumSet	getApplicability()
java.util.Collection	getObjsToTestAgainst()
void	validate(java.util.List attributes, ModelIssues issues)

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule

createIssue, createIssue, createIssue, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from interface org.tanjakostic.jcleancim.validation.CrossRule

getObjsToTestAgainst, validate

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

```
getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity,
logDiagnosis
```

Constructors

Iec61850DOAbbreviationLiteralsNeverUsedInDOName

```
public Iec61850DOAbbreviationLiteralsNeverUsedInDOName(java.util.Collection
allAbbrLiterals)
```

Methods

getApplicability

```
public java.util.EnumSet getApplicability()
```

getObjsToTestAgainst

```
public java.util.Collection getObjsToTestAgainst()
```

validate

```
public void validate(java.util.List attributes,
         ModelIssues issues)
```

org.tanjakostic.jcleancim.validation
Class
AttributeValidator.Iec61850DOAttributesWithSameNameDifferentType

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.validation.AbstractRule
      |
      +-
```
org.tanjakostic.jcleancim.validation.AttributeValidator.Iec61850DOAttributesWithSameNameDifferentType

All Implemented Interfaces:
CrossRule, Rule

---

public static class **AttributeValidator.Iec61850DOAttributesWithSameNameDifferentType**
extends AbstractRule
implements Rule, CrossRule

Constructor Summary

public	Iec61850DOAttributesWithSameNameDifferentType(java.util.Collection allAttributes)

Method Summary

java.util.EnumSet	getApplicability()
java.util.Collection	getObjsToTestAgainst()
void	validate(java.util.List attributes, ModelIssues issues)

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule

createIssue, createIssue, createIssue, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from interface org.tanjakostic.jcleancim.validation.CrossRule

getObjsToTestAgainst, validate

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

```
getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity,
logDiagnosis
```

Constructors

Iec61850DOAttributesWithSameNameDifferentType

```
public Iec61850DOAttributesWithSameNameDifferentType(java.util.Collection
allAttributes)
```

Methods

getApplicability

```
public java.util.EnumSet getApplicability()
```

getObjsToTestAgainst

```
public java.util.Collection getObjsToTestAgainst()
```

validate

```
public void validate(java.util.List attributes,
         ModelIssues issues)
```

org.tanjakostic.jcleancim.validation
Class
AttributeValidator.Iec61850ConditionLiteralsNeverUsedAsConstraints

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.validation.AbstractRule
      |
      +-
```
org.tanjakostic.jcleancim.validation.AttributeValidator.Iec61850ConditionLiteralsNeverUsedAsConstraints

All Implemented Interfaces:
CrossRule, Rule

---

public static class **AttributeValidator.Iec61850ConditionLiteralsNeverUsedAsConstraints**
extends AbstractRule
implements Rule, CrossRule

Constructor Summary

public	Iec61850ConditionLiteralsNeverUsedAsConstraints(java.util.Collection allPresCondLiterals)

Method Summary

java.util.EnumSet	getApplicability()
java.util.Collection	getObjsToTestAgainst()
void	validate(java.util.List attributes, ModelIssues issues)

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule

createIssue, createIssue, createIssue, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from interface org.tanjakostic.jcleancim.validation.CrossRule

getObjsToTestAgainst, validate

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

```
getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity,
logDiagnosis
```

Constructors

Iec61850ConditionLiteralsNeverUsedAsConstraints

```
public Iec61850ConditionLiteralsNeverUsedAsConstraints(java.util.Collection
allPresCondLiterals)
```

Methods

getApplicability

```
public java.util.EnumSet getApplicability()
```

getObjsToTestAgainst

```
public java.util.Collection getObjsToTestAgainst()
```

validate

```
public void validate(java.util.List attributes,
         ModelIssues issues)
```

org.tanjakostic.jcleancim.validation
Class ClassValidator

```
java.lang.Object
  |-org.tanjakostic.jcleancim.validation.AbstractValidator
    |-org.tanjakostic.jcleancim.validation.ClassValidator
```

public class **ClassValidator**
extends AbstractValidator

Validates classes.

Nested Class Summary

class	ClassValidator.CimClassesNameShouldBeSingular ClassValidator.CimClassesNameShouldBeSingular
class	ClassValidator.CimClassesNameStartingWithLowerCase ClassValidator.CimClassesNameStartingWithLowerCase
class	ClassValidator.CimClassesNeverUsedAsTypeForAttribute ClassValidator.CimClassesNeverUsedAsTypeForAttribute
class	ClassValidator.CimClassesNeverUsedInRelationships ClassValidator.CimClassesNeverUsedInRelationships
class	ClassValidator.CimClassesThatShouldNotBeAbstract ClassValidator.CimClassesThatShouldNotBeAbstract
class	ClassValidator.CimClassesThatShouldNotHaveExplicitDependencies ClassValidator.CimClassesThatShouldNotHaveExplicitDependencies
class	ClassValidator.CimClassesThatShouldNotHaveOperations ClassValidator.CimClassesThatShouldNotHaveOperations
class	ClassValidator.CimClassesUsedForAttributesButHaveAssociations ClassValidator.CimClassesUsedForAttributesButHaveAssociations
class	ClassValidator.CimClassesUsedForAttributesButHaveSubclasses ClassValidator.CimClassesUsedForAttributesButHaveSubclasses
class	ClassValidator.CimClassesUsedForAttributesButHaveSuperclasses ClassValidator.CimClassesUsedForAttributesButHaveSuperclasses
class	ClassValidator.CimClassesWithOldDatatypeStereotype ClassValidator.CimClassesWithOldDatatypeStereotype
class	ClassValidator.CimClassesWithUnexpectedElements ClassValidator.CimClassesWithUnexpectedElements
class	ClassValidator.CimCompoundClassesWithNoAttributes ClassValidator.CimCompoundClassesWithNoAttributes
class	ClassValidator.CimDatatypeClassesWithInvalidAttributes ClassValidator.CimDatatypeClassesWithInvalidAttributes

class	ClassValidator.CimPrimitiveClassesWithAttributes ClassValidator.CimPrimitiveClassesWithAttributes
class	ClassValidator.CimPrimitiveClassesWithIllegalOwner ClassValidator.CimPrimitiveClassesWithIllegalOwner
class	ClassValidator.ClassesMissingDoc ClassValidator.ClassesMissingDoc
class	ClassValidator.ClassesThatShouldNotBeAssociationClass ClassValidator.ClassesThatShouldNotBeAssociationClass
class	ClassValidator.ClassesThatShouldNotHaveNestingThroughAttribute ClassValidator.ClassesThatShouldNotHaveNestingThroughAttribute
class	ClassValidator.ClassesWithBadCharacterInName ClassValidator.ClassesWithBadCharacterInName
class	ClassValidator.ClassesWithBadDocEnd ClassValidator.ClassesWithBadDocEnd
class	ClassValidator.ClassesWithBadDocStart ClassValidator.ClassesWithBadDocStart
class	ClassValidator.ClassesWithDuplicateInheritedAttributeNames ClassValidator.ClassesWithDuplicateInheritedAttributeNames
class	ClassValidator.ClassesWithDuplicateOwnOrInheritedAssociationEndNames ClassValidator.ClassesWithDuplicateOwnOrInheritedAssociationEndNames
class	ClassValidator.ClassesWithLeafPropSet ClassValidator.ClassesWithLeafPropSet
class	ClassValidator.ClassesWithMultipleSuperclasses ClassValidator.ClassesWithMultipleSuperclasses
class	ClassValidator.ClassesWithPersistentPropSet ClassValidator.ClassesWithPersistentPropSet
class	ClassValidator.ClassesWithQuestionableAttributeCount ClassValidator.ClassesWithQuestionableAttributeCount
class	ClassValidator.ClassesWithRootPropSet ClassValidator.ClassesWithRootPropSet
class	ClassValidator.ClassesWithSameName ClassValidator.ClassesWithSameName
class	ClassValidator.ClassesWithSelfDependency ClassValidator.ClassesWithSelfDependency
class	ClassValidator.ClassesWithSelfInheritance ClassValidator.ClassesWithSelfInheritance
class	ClassValidator.ClassesWithSuperclassesFromUnallowedOwner ClassValidator.ClassesWithSuperclassesFromUnallowedOwner
class	ClassValidator.ClassesWithUnallowedStereotype ClassValidator.ClassesWithUnallowedStereotype

class	ClassValidator.ClassesWithUnallowedTagNames ClassValidator.ClassesWithUnallowedTagNames
class	ClassValidator.ClassesWithUnexpectedConnectors ClassValidator.ClassesWithUnexpectedConnectors
class	ClassValidator.EnumClassesWithBadName ClassValidator.EnumClassesWithBadName
class	ClassValidator.EnumClassesWithDuplicateCodes ClassValidator.EnumClassesWithDuplicateCodes
class	ClassValidator.EnumClassesWithNoLiterals ClassValidator.EnumClassesWithNoLiterals
class	ClassValidator.EnumClassesWithSingleLiteral ClassValidator.EnumClassesWithSingleLiteral
class	ClassValidator.EnumClassesWithSomeCodesMissing ClassValidator.EnumClassesWithSomeCodesMissing
class	ClassValidator.EnumClassesWithTwoLiterals ClassValidator.EnumClassesWithTwoLiterals
class	ClassValidator.Iec61850ClassesThatShouldHaveAliasAsTitle ClassValidator.Iec61850ClassesThatShouldHaveAliasAsTitle
class	ClassValidator.Iec61850ClassesThatShouldHaveTaggedValuesForDocgen ClassValidator.Iec61850ClassesThatShouldHaveTaggedValuesForDocgen
class	ClassValidator.Iec61850ClassesWithInvalidConstraints ClassValidator.Iec61850ClassesWithInvalidConstraints
class	ClassValidator.Iec61850ClassesWithMissingCondIDTextInConstraints ClassValidator.Iec61850ClassesWithMissingCondIDTextInConstraints
class	ClassValidator.Iec61850LNClassesInWrongGroup ClassValidator.Iec61850LNClassesInWrongGroup
class	ClassValidator.Iec61850LNClassesMalformedName ClassValidator.Iec61850LNClassesMalformedName
class	ClassValidator.Iec61850LNClassesWithSuperfluousConstraints ClassValidator.Iec61850LNClassesWithSuperfluousConstraints

Method Summary

boolean	enabled()
java.util.List	getScopedUmlObjects()

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractValidator

addCrossRule, addSimpleRule, displayAllAvailableRuleNames, displayAvailableRuleNames, enabled, getAllCrossRules, getAllRules, getAllSimpleRules, getCfg, getCheckedCrossRules, getCheckedRules, getCheckedSimpleRules, getCollectedIssues, getScopedUmlObjects, validate

Methods inherited from class `java.lang.Object`
`clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait`

Methods

enabled

`public boolean` **`enabled`**`()`

Returns whether the validation for this validator has been enabled (by configuration).

getScopedUmlObjects

`public java.util.List` **`getScopedUmlObjects`**`()`

Returns the elements retained for validation, for the configured scope.

org.tanjakostic.jcleancim.validation
Class ClassValidator.CimClassesWithUnexpectedElements

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.validation.AbstractRule
      |
      +-
```
org.tanjakostic.jcleancim.validation.ClassValidator.CimClassesWithUnexpectedElements

All Implemented Interfaces:
SimpleRule, Rule

---

public static class **ClassValidator.CimClassesWithUnexpectedElements**
extends AbstractRule
implements Rule, SimpleRule

Constructor Summary

public	CimClassesWithUnexpectedElements()

Method Summary

java.util.EnumSet	getApplicability()
void	validate(UmlClass o, ModelIssues issues)

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule

createIssue, createIssue, createIssue, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from interface org.tanjakostic.jcleancim.validation.SimpleRule

validate

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

---

(continued from last page)

Constructors

CimClassesWithUnexpectedElements

```
public CimClassesWithUnexpectedElements()
```

Methods

getApplicability

```
public java.util.EnumSet getApplicability()
```

---

validate

```
public void validate(UmlClass o,
         ModelIssues issues)
```

org.tanjakostic.jcleancim.validation
Class ClassValidator.ClassesWithUnexpectedConnectors

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.validation.AbstractRule
      |
      +-
```
org.tanjakostic.jcleancim.validation.ClassValidator.ClassesWithUnexpectedConnectors

All Implemented Interfaces:
SimpleRule, Rule

---

public static class **ClassValidator.ClassesWithUnexpectedConnectors**
extends AbstractRule
implements Rule, SimpleRule

Constructor Summary

public	ClassesWithUnexpectedConnectors()

Method Summary

java.util.EnumSet	getApplicability()
void	validate(UmlClass o, ModelIssues issues)

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule

createIssue, createIssue, createIssue, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from interface org.tanjakostic.jcleancim.validation.SimpleRule

validate

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

---

(continued from last page)

Constructors

ClassesWithUnexpectedConnectors

```
public ClassesWithUnexpectedConnectors()
```

Methods

getApplicability

```
public java.util.EnumSet getApplicability()
```

---

validate

```
public void validate(UmlClass o,
         ModelIssues issues)
```

org.tanjakostic.jcleancim.validation
Class ClassValidator.ClassesWithQuestionableAttributeCount

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.validation.AbstractRule
      |
      +-
```
org.tanjakostic.jcleancim.validation.ClassValidator.ClassesWithQuestionableAttributeCount

All Implemented Interfaces:
SimpleRule, Rule

Direct Known Subclasses:
EnumClassesWithTwoLiterals, EnumClassesWithSingleLiteral, CimCompoundClassesWithNoAttributes, EnumClassesWithNoLiterals

---

public static abstract class **ClassValidator.ClassesWithQuestionableAttributeCount**
extends AbstractRule
implements Rule, SimpleRule

Constructor Summary

public	ClassesWithQuestionableAttributeCount(org.apache.log4j.Level level, Rule.Severity severity, java.lang.String hypo, java.lang.String howToFix)

Method Summary

abstract boolean	satisfiesCondition(UmlClass clazz)
void	validate(UmlClass o, ModelIssues issues)

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule

createIssue, createIssue, createIssue, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from interface org.tanjakostic.jcleancim.validation.SimpleRule

validate

Methods inherited from interface `org.tanjakostic.jcleancim.validation.Rule`
`getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis`

Constructors

ClassesWithQuestionableAttributeCount

```
public ClassesWithQuestionableAttributeCount(org.apache.log4j.Level level,
                                             Rule.Severity severity,
                                             java.lang.String hypo,
                                             java.lang.String howToFix)
```

Methods

validate

```
public final void validate(UmlClass o,
         ModelIssues issues)
```

satisfiesCondition

```
protected abstract boolean satisfiesCondition(UmlClass clazz)
```

org.tanjakostic.jcleancim.validation
Class ClassValidator.EnumClassesWithNoLiterals

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.validation.AbstractRule
      |
      +-
org.tanjakostic.jcleancim.validation.ClassValidator.ClassesWithQuestionableAttributeCou
nt
          |
          +-
```
org.tanjakostic.jcleancim.validation.ClassValidator.EnumClassesWithNoLiterals

All Implemented Interfaces:
Rule, SimpleRule

---

public static class **ClassValidator.EnumClassesWithNoLiterals**
extends ClassValidator.ClassesWithQuestionableAttributeCount

Constructor Summary

public	EnumClassesWithNoLiterals()

Method Summary

java.util.EnumSet	getApplicability()
boolean	satisfiesCondition(UmlClass o)

Methods inherited from class org.tanjakostic.jcleancim.validation.ClassValidator.ClassesWithQuestionableAttributeCount

satisfiesCondition, validate

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule

createIssue, createIssue, createIssue, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from interface org.tanjakostic.jcleancim.validation.SimpleRule

validate

Methods inherited from interface `org.tanjakostic.jcleancim.validation.Rule`
`getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis`

Constructors

EnumClassesWithNoLiterals

```
public EnumClassesWithNoLiterals()
```

Methods

getApplicability

```
public java.util.EnumSet getApplicability()
```

satisfiesCondition

```
protected boolean satisfiesCondition(UmlClass o)
```

org.tanjakostic.jcleancim.validation
Class ClassValidator.CimCompoundClassesWithNoAttributes

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.validation.AbstractRule
      |
      +-
org.tanjakostic.jcleancim.validation.ClassValidator.ClassesWithQuestionableAttributeCou
nt
          |
          +-
```
org.tanjakostic.jcleancim.validation.ClassValidator.CimCompoundClassesWithNoAttributes

All Implemented Interfaces:
Rule, SimpleRule

---

public static class **ClassValidator.CimCompoundClassesWithNoAttributes**
extends ClassValidator.ClassesWithQuestionableAttributeCount

Constructor Summary

public	CimCompoundClassesWithNoAttributes()

Method Summary

java.util.EnumSet	getApplicability()
boolean	satisfiesCondition(UmlClass o)

Methods inherited from class org.tanjakostic.jcleancim.validation.ClassValidator.ClassesWithQuestionableAttributeCount

satisfiesCondition, validate

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule

createIssue, createIssue, createIssue, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from interface org.tanjakostic.jcleancim.validation.SimpleRule

validate

Methods inherited from interface `org.tanjakostic.jcleancim.validation.Rule`
`getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis`

Constructors

CimCompoundClassesWithNoAttributes

```
public CimCompoundClassesWithNoAttributes()
```

Methods

getApplicability

```
public java.util.EnumSet getApplicability()
```

satisfiesCondition

```
protected boolean satisfiesCondition(UmlClass o)
```

org.tanjakostic.jcleancim.validation
Class ClassValidator.EnumClassesWithSingleLiteral

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.validation.AbstractRule
      |
      +-
org.tanjakostic.jcleancim.validation.ClassValidator.ClassesWithQuestionableAttributeCou
nt
          |
          +-
org.tanjakostic.jcleancim.validation.ClassValidator.EnumClassesWithSingleLiteral
```

All Implemented Interfaces:
Rule, SimpleRule

---

public static class **ClassValidator.EnumClassesWithSingleLiteral**
extends ClassValidator.ClassesWithQuestionableAttributeCount

Constructor Summary

public	EnumClassesWithSingleLiteral()

Method Summary

java.util.EnumSet	getApplicability()
boolean	satisfiesCondition(UmlClass o)

Methods inherited from class org.tanjakostic.jcleancim.validation.ClassValidator.ClassesWithQuestionableAttributeCount

satisfiesCondition, validate

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule

createIssue, createIssue, createIssue, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from interface org.tanjakostic.jcleancim.validation.SimpleRule

validate

Methods inherited from interface `org.tanjakostic.jcleancim.validation.Rule`
`getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis`

Constructors

EnumClassesWithSingleLiteral

```
public EnumClassesWithSingleLiteral()
```

Methods

getApplicability

```
public java.util.EnumSet getApplicability()
```

satisfiesCondition

```
protected boolean satisfiesCondition(UmlClass o)
```

org.tanjakostic.jcleancim.validation
Class ClassValidator.EnumClassesWithTwoLiterals

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.validation.AbstractRule
      |
      +-
org.tanjakostic.jcleancim.validation.ClassValidator.ClassesWithQuestionableAttributeCou
nt
          |
          +-
org.tanjakostic.jcleancim.validation.ClassValidator.EnumClassesWithTwoLiterals
```

All Implemented Interfaces:
Rule, SimpleRule

---

public static class **ClassValidator.EnumClassesWithTwoLiterals**
extends ClassValidator.ClassesWithQuestionableAttributeCount

Constructor Summary

public	EnumClassesWithTwoLiterals()

Method Summary

java.util.EnumSet	getApplicability()
boolean	satisfiesCondition(UmlClass o)

Methods inherited from class org.tanjakostic.jcleancim.validation.ClassValidator.ClassesWithQuestionableAttributeCount

satisfiesCondition, validate

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule

createIssue, createIssue, createIssue, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from interface org.tanjakostic.jcleancim.validation.SimpleRule

validate

Methods inherited from interface `org.tanjakostic.jcleancim.validation.Rule`
`getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis`

Constructors

EnumClassesWithTwoLiterals

```
public EnumClassesWithTwoLiterals()
```

Methods

getApplicability

```
public java.util.EnumSet getApplicability()
```

satisfiesCondition

```
protected boolean satisfiesCondition(UmlClass o)
```

org.tanjakostic.jcleancim.validation
Class ClassValidator.EnumClassesWithBadName

```
java.lang.Object
  |-org.tanjakostic.jcleancim.validation.AbstractRule
    |-org.tanjakostic.jcleancim.validation.ClassValidator.EnumClassesWithBadName
```

All Implemented Interfaces:
SimpleRule, Rule

---

public static class **ClassValidator.EnumClassesWithBadName**
extends AbstractRule
implements Rule, SimpleRule

Field Summary

public static final	ENUM Value: **enum**
public static final	KIND Value: **Kind**
public static final	TYPE Value: **type**

Constructor Summary

public	EnumClassesWithBadName()

Method Summary

java.util.EnumSet	getApplicability()
void	validate(UmlClass o, ModelIssues issues)

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule

createIssue, createIssue, createIssue, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from interface org.tanjakostic.jcleancim.validation.SimpleRule
validate

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule
getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Fields

TYPE

public static final java.lang.String **TYPE**

Constant value: **type**

ENUM

public static final java.lang.String **ENUM**

Constant value: **enum**

KIND

public static final java.lang.String **KIND**

Constant value: **Kind**

Constructors

EnumClassesWithBadName

public **EnumClassesWithBadName**()

Methods

getApplicability

public java.util.EnumSet **getApplicability**()

validate

public void **validate**(UmlClass o,
ModelIssues issues)

(continued from last page)

org.tanjakostic.jcleancim.validation
Class ClassValidator.ClassesWithSelfInheritance

```
java.lang.Object
  |-org.tanjakostic.jcleancim.validation.AbstractRule
    |-org.tanjakostic.jcleancim.validation.ClassValidator.ClassesWithSelfInheritance
```

All Implemented Interfaces:
SimpleRule, Rule

public static class **ClassValidator.ClassesWithSelfInheritance**
extends AbstractRule
implements Rule, SimpleRule

Constructor Summary

public	ClassesWithSelfInheritance()

Method Summary

java.util.EnumSet	getApplicability()
void	validate(UmlClass o, ModelIssues issues)

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule

createIssue, createIssue, createIssue, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from interface org.tanjakostic.jcleancim.validation.SimpleRule

validate

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

(continued from last page)

Constructors

ClassesWithSelfInheritance

```
public ClassesWithSelfInheritance()
```

Methods

getApplicability

```
public java.util.EnumSet getApplicability()
```

---

validate

```
public void validate(UmlClass o,
         ModelIssues issues)
```

org.tanjakostic.jcleancim.validation
Class ClassValidator.CimPrimitiveClassesWithAttributes

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.validation.AbstractRule
      |
      +-
```
org.tanjakostic.jcleancim.validation.ClassValidator.CimPrimitiveClassesWithAttributes

All Implemented Interfaces:
SimpleRule, Rule

---

public static class **ClassValidator.CimPrimitiveClassesWithAttributes**
extends AbstractRule
implements Rule, SimpleRule

Constructor Summary

public	CimPrimitiveClassesWithAttributes()

Method Summary

java.util.EnumSet	getApplicability()
void	validate(UmlClass o, ModelIssues issues)

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule

createIssue, createIssue, createIssue, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from interface org.tanjakostic.jcleancim.validation.SimpleRule

validate

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

---

(continued from last page)

Constructors

CimPrimitiveClassesWithAttributes

```
public CimPrimitiveClassesWithAttributes()
```

Methods

getApplicability

```
public java.util.EnumSet getApplicability()
```

---

validate

```
public void validate(UmlClass o,
         ModelIssues issues)
```

org.tanjakostic.jcleancim.validation
Class ClassValidator.CimPrimitiveClassesWithIllegalOwner

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.validation.AbstractRule
      |
      +-
```
org.tanjakostic.jcleancim.validation.ClassValidator.CimPrimitiveClassesWithIllegalOwner

All Implemented Interfaces:
SimpleRule, Rule

---

public static class **ClassValidator.CimPrimitiveClassesWithIllegalOwner**
extends AbstractRule
implements Rule, SimpleRule

Constructor Summary

public	CimPrimitiveClassesWithIllegalOwner()

Method Summary

java.util.EnumSet	getApplicability()
void	validate(UmlClass o, ModelIssues issues)

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule

createIssue, createIssue, createIssue, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from interface org.tanjakostic.jcleancim.validation.SimpleRule

validate

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

---

(continued from last page)

Constructors

CimPrimitiveClassesWithIllegalOwner

```
public CimPrimitiveClassesWithIllegalOwner()
```

Methods

getApplicability

```
public java.util.EnumSet getApplicability()
```

---

validate

```
public void validate(UmlClass o,
         ModelIssues issues)
```

org.tanjakostic.jcleancim.validation
Class ClassValidator.ClassesWithDuplicateInheritedAttributeNames

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.validation.AbstractRule
      |
      +-
```
org.tanjakostic.jcleancim.validation.ClassValidator.ClassesWithDuplicateInheritedAttributeNames

All Implemented Interfaces:
SimpleRule, Rule

---

public static class **ClassValidator.ClassesWithDuplicateInheritedAttributeNames**
extends AbstractRule
implements Rule, SimpleRule

Constructor Summary

public	ClassesWithDuplicateInheritedAttributeNames()

Method Summary

java.util.EnumSet	getApplicability()
void	validate(UmlClass o, ModelIssues issues)

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule

createIssue, createIssue, createIssue, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from interface org.tanjakostic.jcleancim.validation.SimpleRule

validate

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Constructors

ClassesWithDuplicateInheritedAttributeNames

```
public ClassesWithDuplicateInheritedAttributeNames()
```

Methods

getApplicability

```
public java.util.EnumSet getApplicability()
```

validate

```
public void validate(UmlClass o,
         ModelIssues issues)
```

org.tanjakostic.jcleancim.validation
Class
ClassValidator.ClassesWithDuplicateOwnOrInheritedAssociationEndNames

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.validation.AbstractRule
      |
      +-
```
org.tanjakostic.jcleancim.validation.ClassValidator.ClassesWithDuplicateOwnOrInheritedAssociationEndNames

All Implemented Interfaces:
SimpleRule, Rule

---

public static class **ClassValidator.ClassesWithDuplicateOwnOrInheritedAssociationEndNames**
extends AbstractRule
implements Rule, SimpleRule

Constructor Summary

public	ClassesWithDuplicateOwnOrInheritedAssociationEndNames()

Method Summary

java.util.EnumSet	getApplicability()
void	validate(UmlClass o, ModelIssues issues)

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule

createIssue, createIssue, createIssue, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from interface org.tanjakostic.jcleancim.validation.SimpleRule

validate

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Constructors

ClassesWithDuplicateOwnOrInheritedAssociationEndNames

```
public ClassesWithDuplicateOwnOrInheritedAssociationEndNames()
```

Methods

getApplicability

```
public java.util.EnumSet getApplicability()
```

validate

```
public void validate(UmlClass o,
         ModelIssues issues)
```

org.tanjakostic.jcleancim.validation
Class ClassValidator.ClassesWithSelfDependency

```
java.lang.Object
  |-org.tanjakostic.jcleancim.validation.AbstractRule
    |-org.tanjakostic.jcleancim.validation.ClassValidator.ClassesWithSelfDependency
```

All Implemented Interfaces:
SimpleRule, Rule

public static class **ClassValidator.ClassesWithSelfDependency**
extends AbstractRule
implements Rule, SimpleRule

Constructor Summary

public	ClassesWithSelfDependency()

Method Summary

java.util.EnumSet	getApplicability()
void	validate(UmlClass o, ModelIssues issues)

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule

createIssue, createIssue, createIssue, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from interface org.tanjakostic.jcleancim.validation.SimpleRule

validate

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

(continued from last page)

Constructors

ClassesWithSelfDependency

```
public ClassesWithSelfDependency()
```

Methods

getApplicability

```
public java.util.EnumSet getApplicability()
```

---

validate

```
public void validate(UmlClass o,
         ModelIssues issues)
```

org.tanjakostic.jcleancim.validation
Class ClassValidator.ClassesWithLeafPropSet

```
java.lang.Object
  |-org.tanjakostic.jcleancim.validation.AbstractRule
    |-org.tanjakostic.jcleancim.validation.ClassValidator.ClassesWithLeafPropSet
```

All Implemented Interfaces:
SimpleRule, Rule

---

public static class **ClassValidator.ClassesWithLeafPropSet**
extends AbstractRule
implements Rule, SimpleRule

Constructor Summary

public	ClassesWithLeafPropSet()

Method Summary

java.util.EnumSet	getApplicability()
void	validate(UmlClass o, ModelIssues issues)

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule

createIssue, createIssue, createIssue, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from interface org.tanjakostic.jcleancim.validation.SimpleRule

validate

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

---

(continued from last page)

Constructors

ClassesWithLeafPropSet

```
public ClassesWithLeafPropSet()
```

Methods

getApplicability

```
public java.util.EnumSet getApplicability()
```

---

validate

```
public void validate(UmlClass o,
         ModelIssues issues)
```

org.tanjakostic.jcleancim.validation
Class ClassValidator.ClassesWithRootPropSet

```
java.lang.Object
  |-org.tanjakostic.jcleancim.validation.AbstractRule
    |-org.tanjakostic.jcleancim.validation.ClassValidator.ClassesWithRootPropSet
```

All Implemented Interfaces:
SimpleRule, Rule

---

public static class **ClassValidator.ClassesWithRootPropSet**
extends AbstractRule
implements Rule, SimpleRule

Constructor Summary

public	ClassesWithRootPropSet()

Method Summary

java.util.EnumSet	getApplicability()
void	validate(UmlClass o, ModelIssues issues)

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule

createIssue, createIssue, createIssue, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from interface org.tanjakostic.jcleancim.validation.SimpleRule

validate

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

---

(continued from last page)

Constructors

ClassesWithRootPropSet

```
public ClassesWithRootPropSet()
```

Methods

getApplicability

```
public java.util.EnumSet getApplicability()
```

---

validate

```
public void validate(UmlClass o,
         ModelIssues issues)
```

org.tanjakostic.jcleancim.validation
Class ClassValidator.ClassesWithPersistentPropSet

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.validation.AbstractRule
      |
      +-
```
org.tanjakostic.jcleancim.validation.ClassValidator.ClassesWithPersistentPropSet

All Implemented Interfaces:
SimpleRule, Rule

---

public static class **ClassValidator.ClassesWithPersistentPropSet**
extends AbstractRule
implements Rule, SimpleRule

Constructor Summary

public	ClassesWithPersistentPropSet()

Method Summary

java.util.EnumSet	getApplicability()
void	validate(UmlClass o, ModelIssues issues)

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule

createIssue, createIssue, createIssue, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from interface org.tanjakostic.jcleancim.validation.SimpleRule

validate

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

---

(continued from last page)

Constructors

ClassesWithPersistentPropSet

```
public ClassesWithPersistentPropSet()
```

Methods

getApplicability

```
public java.util.EnumSet getApplicability()
```

---

validate

```
public void validate(UmlClass o,
         ModelIssues issues)
```

org.tanjakostic.jcleancim.validation
Class ClassValidator.ClassesWithMultipleSuperclasses

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.validation.AbstractRule
      |
      +-
```
org.tanjakostic.jcleancim.validation.ClassValidator.ClassesWithMultipleSuperclasses

All Implemented Interfaces:
SimpleRule, Rule

---

public static class **ClassValidator.ClassesWithMultipleSuperclasses**
extends AbstractRule
implements Rule, SimpleRule

Constructor Summary

public	ClassesWithMultipleSuperclasses()

Method Summary

java.util.EnumSet	getApplicability()
void	validate(UmlClass o, ModelIssues issues)

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule

createIssue, createIssue, createIssue, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from interface org.tanjakostic.jcleancim.validation.SimpleRule

validate

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

---

(continued from last page)

Constructors

ClassesWithMultipleSuperclasses

```
public ClassesWithMultipleSuperclasses()
```

Methods

getApplicability

```
public java.util.EnumSet getApplicability()
```

---

validate

```
public void validate(UmlClass o,
         ModelIssues issues)
```

org.tanjakostic.jcleancim.validation
Class ClassValidator.ClassesWithSuperclassesFromUnallowedOwner

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.validation.AbstractRule
      |
      +-
```
org.tanjakostic.jcleancim.validation.ClassValidator.ClassesWithSuperclassesFromUnallowedOwner

All Implemented Interfaces:
SimpleRule, Rule

---

public static class **ClassValidator.ClassesWithSuperclassesFromUnallowedOwner**
extends AbstractRule
implements Rule, SimpleRule

Constructor Summary

public	ClassesWithSuperclassesFromUnallowedOwner()

Method Summary

java.util.EnumSet	getApplicability()
void	validate(UmlClass o, ModelIssues issues)

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule

createIssue, createIssue, createIssue, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from interface org.tanjakostic.jcleancim.validation.SimpleRule

validate

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Constructors

ClassesWithSuperclassesFromUnallowedOwner

```
public ClassesWithSuperclassesFromUnallowedOwner()
```

Methods

getApplicability

```
public java.util.EnumSet getApplicability()
```

validate

```
public void validate(UmlClass o,
         ModelIssues issues)
```

org.tanjakostic.jcleancim.validation
Class ClassValidator.ClassesThatShouldNotBeAssociationClass

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.validation.AbstractRule
      |
      +-
```
org.tanjakostic.jcleancim.validation.ClassValidator.ClassesThatShouldNotBeAssociationClass

All Implemented Interfaces:
SimpleRule, Rule

---

public static class **ClassValidator.ClassesThatShouldNotBeAssociationClass**
extends AbstractRule
implements Rule, SimpleRule

Constructor Summary

public	ClassesThatShouldNotBeAssociationClass()

Method Summary

java.util.EnumSet	getApplicability()
void	validate(UmlClass o, ModelIssues issues)

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule

createIssue, createIssue, createIssue, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from interface org.tanjakostic.jcleancim.validation.SimpleRule

validate

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Constructors

ClassesThatShouldNotBeAssociationClass

```
public ClassesThatShouldNotBeAssociationClass()
```

Methods

getApplicability

```
public java.util.EnumSet getApplicability()
```

validate

```
public void validate(UmlClass o,
         ModelIssues issues)
```

org.tanjakostic.jcleancim.validation
Class ClassValidator.ClassesWithUnallowedStereotype

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.validation.AbstractRule
      |
      +-
org.tanjakostic.jcleancim.validation.AbstractRule.AbstractRuleWithSubobjectsAndSkips
          |
          +-
org.tanjakostic.jcleancim.validation.AbstractRule.UmlObjectsWithUnallowedStereotype
              |
              +-
```

org.tanjakostic.jcleancim.validation.ClassValidator.ClassesWithUnallowedStereotype

All Implemented Interfaces:
Rule, SimpleRule

---

public static class **ClassValidator.ClassesWithUnallowedStereotype**
extends AbstractRule.UmlObjectsWithUnallowedStereotype

Constructor Summary

public	ClassesWithUnallowedStereotype()

Method Summary

java.util.EnumSet	getApplicability()

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule.UmlObjectsWithUnallowedStereotype

doValidate

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule.AbstractRuleWithSubobjectsAndSkips

doValidate, getSubObjects, skipSubobjectValidation, skipValidation, validate

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule

createIssue, createIssue, createIssue, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from interface `org.tanjakostic.jcleancim.validation.SimpleRule`
`validate`

Methods inherited from interface `org.tanjakostic.jcleancim.validation.Rule`
`getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis`

Constructors

ClassesWithUnallowedStereotype

`public` **`ClassesWithUnallowedStereotype`**`()`

Methods

getApplicability

`public java.util.EnumSet` **`getApplicability`**`()`

org.tanjakostic.jcleancim.validation
Class ClassValidator.CimClassesWithOldDatatypeStereotype

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.validation.AbstractRule
      |
      +-
```
org.tanjakostic.jcleancim.validation.ClassValidator.CimClassesWithOldDatatypeStereotype

All Implemented Interfaces:
SimpleRule, Rule

---

public static class **ClassValidator.CimClassesWithOldDatatypeStereotype**
extends AbstractRule
implements Rule, SimpleRule

Constructor Summary

public	CimClassesWithOldDatatypeStereotype()

Method Summary

java.util.EnumSet	getApplicability()
void	validate(UmlClass o, ModelIssues issues)

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule

createIssue, createIssue, createIssue, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from interface org.tanjakostic.jcleancim.validation.SimpleRule

validate

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

---

(continued from last page)

Constructors

CimClassesWithOldDatatypeStereotype

```
public CimClassesWithOldDatatypeStereotype()
```

Methods

getApplicability

```
public java.util.EnumSet getApplicability()
```

---

validate

```
public void validate(UmlClass o,
         ModelIssues issues)
```

org.tanjakostic.jcleancim.validation
Class ClassValidator.CimClassesUsedForAttributesButHaveAssociations

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.validation.AbstractRule
      |
      +-
```
org.tanjakostic.jcleancim.validation.ClassValidator.CimClassesUsedForAttributesButHaveAssociations

All Implemented Interfaces:
SimpleRule, Rule

---

public static class **ClassValidator.CimClassesUsedForAttributesButHaveAssociations**
extends AbstractRule
implements Rule, SimpleRule

Constructor Summary

public	CimClassesUsedForAttributesButHaveAssociations()

Method Summary

java.util.EnumSet	getApplicability()
void	validate(UmlClass o, ModelIssues issues)

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule

createIssue, createIssue, createIssue, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from interface org.tanjakostic.jcleancim.validation.SimpleRule

validate

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Constructors

CimClassesUsedForAttributesButHaveAssociations

```
public CimClassesUsedForAttributesButHaveAssociations()
```

Methods

getApplicability

```
public java.util.EnumSet getApplicability()
```

validate

```
public void validate(UmlClass o,
         ModelIssues issues)
```

org.tanjakostic.jcleancim.validation
Class ClassValidator.CimClassesUsedForAttributesButHaveSubclasses

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.validation.AbstractRule
      |
      +-
```
org.tanjakostic.jcleancim.validation.ClassValidator.CimClassesUsedForAttributesButHaveSubclasses

All Implemented Interfaces:
SimpleRule, Rule

---

public static class **ClassValidator.CimClassesUsedForAttributesButHaveSubclasses**
extends AbstractRule
implements Rule, SimpleRule

Constructor Summary

public	CimClassesUsedForAttributesButHaveSubclasses()

Method Summary

java.util.EnumSet	getApplicability()
void	validate(UmlClass o, ModelIssues issues)

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule

createIssue, createIssue, createIssue, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from interface org.tanjakostic.jcleancim.validation.SimpleRule

validate

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Constructors

CimClassesUsedForAttributesButHaveSubclasses

```
public CimClassesUsedForAttributesButHaveSubclasses()
```

Methods

getApplicability

```
public java.util.EnumSet getApplicability()
```

validate

```
public void validate(UmlClass o,
         ModelIssues issues)
```

org.tanjakostic.jcleancim.validation
Class ClassValidator.CimClassesUsedForAttributesButHaveSuperclasses

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.validation.AbstractRule
      |
      +-
```
org.tanjakostic.jcleancim.validation.ClassValidator.CimClassesUsedForAttributesButHaveSuperclasses

All Implemented Interfaces:
SimpleRule, Rule

---

public static class **ClassValidator.CimClassesUsedForAttributesButHaveSuperclasses**
extends AbstractRule
implements Rule, SimpleRule

Constructor Summary

public	CimClassesUsedForAttributesButHaveSuperclasses()

Method Summary

java.util.EnumSet	getApplicability()
void	validate(UmlClass o, ModelIssues issues)

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule

createIssue, createIssue, createIssue, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from interface org.tanjakostic.jcleancim.validation.SimpleRule

validate

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Constructors

CimClassesUsedForAttributesButHaveSuperclasses

```
public CimClassesUsedForAttributesButHaveSuperclasses()
```

Methods

getApplicability

```
public java.util.EnumSet getApplicability()
```

validate

```
public void validate(UmlClass o,
         ModelIssues issues)
```

org.tanjakostic.jcleancim.validation
Class ClassValidator.CimClassesThatShouldNotBeAbstract

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.validation.AbstractRule
      |
      +-
```
org.tanjakostic.jcleancim.validation.ClassValidator.CimClassesThatShouldNotBeAbstract

All Implemented Interfaces:
SimpleRule, Rule

---

public static class **ClassValidator.CimClassesThatShouldNotBeAbstract**
extends AbstractRule
implements Rule, SimpleRule

Constructor Summary

public	CimClassesThatShouldNotBeAbstract()

Method Summary

java.util.EnumSet	getApplicability()
void	validate(UmlClass o, ModelIssues issues)

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule

createIssue, createIssue, createIssue, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from interface org.tanjakostic.jcleancim.validation.SimpleRule

validate

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

---

(continued from last page)

Constructors

CimClassesThatShouldNotBeAbstract

```
public CimClassesThatShouldNotBeAbstract()
```

Methods

getApplicability

```
public java.util.EnumSet getApplicability()
```

---

validate

```
public void validate(UmlClass o,
         ModelIssues issues)
```

org.tanjakostic.jcleancim.validation
Class ClassValidator.CimClassesThatShouldNotHaveOperations

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.validation.AbstractRule
      |
      +-
```
org.tanjakostic.jcleancim.validation.ClassValidator.CimClassesThatShouldNotHaveOperations

All Implemented Interfaces:
SimpleRule, Rule

---

public static class **ClassValidator.CimClassesThatShouldNotHaveOperations**
extends AbstractRule
implements Rule, SimpleRule

Constructor Summary

public	CimClassesThatShouldNotHaveOperations()

Method Summary

java.util.EnumSet	getApplicability()
void	validate(UmlClass o, ModelIssues issues)

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule

createIssue, createIssue, createIssue, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from interface org.tanjakostic.jcleancim.validation.SimpleRule

validate

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Constructors

CimClassesThatShouldNotHaveOperations

```
public CimClassesThatShouldNotHaveOperations()
```

Methods

getApplicability

```
public java.util.EnumSet getApplicability()
```

validate

```
public void validate(UmlClass o,
         ModelIssues issues)
```

org.tanjakostic.jcleancim.validation
Class ClassValidator.CimClassesThatShouldNotHaveExplicitDependencies

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.validation.AbstractRule
      |
      +-
```
org.tanjakostic.jcleancim.validation.ClassValidator.CimClassesThatShouldNotHaveExplicitDependencies

All Implemented Interfaces:
SimpleRule, Rule

---

public static class **ClassValidator.CimClassesThatShouldNotHaveExplicitDependencies**
extends AbstractRule
implements Rule, SimpleRule

Constructor Summary

public	CimClassesThatShouldNotHaveExplicitDependencies()

Method Summary

java.util.EnumSet	getApplicability()
void	validate(UmlClass o, ModelIssues issues)

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule

createIssue, createIssue, createIssue, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from interface org.tanjakostic.jcleancim.validation.SimpleRule

validate

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Constructors

CimClassesThatShouldNotHaveExplicitDependencies

```
public CimClassesThatShouldNotHaveExplicitDependencies()
```

Methods

getApplicability

```
public java.util.EnumSet getApplicability()
```

validate

```
public void validate(UmlClass o,
         ModelIssues issues)
```

org.tanjakostic.jcleancim.validation
Class ClassValidator.ClassesThatShouldNotHaveNestingThroughAttribute

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.validation.AbstractRule
      |
      +-
```
org.tanjakostic.jcleancim.validation.ClassValidator.ClassesThatShouldNotHaveNestingThroughAttribute

All Implemented Interfaces:
SimpleRule, Rule

---

public static class **ClassValidator.ClassesThatShouldNotHaveNestingThroughAttribute**
extends AbstractRule
implements Rule, SimpleRule

Constructor Summary

public	ClassesThatShouldNotHaveNestingThroughAttribute()

Method Summary

java.util.EnumSet	getApplicability()
void	validate(UmlClass o, ModelIssues issues)

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule

createIssue, createIssue, createIssue, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from interface org.tanjakostic.jcleancim.validation.SimpleRule

validate

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Constructors

ClassesThatShouldNotHaveNestingThroughAttribute

```
public ClassesThatShouldNotHaveNestingThroughAttribute()
```

Methods

getApplicability

```
public java.util.EnumSet getApplicability()
```

validate

```
public void validate(UmlClass o,
         ModelIssues issues)
```

org.tanjakostic.jcleancim.validation
Class ClassValidator.Iec61850ClassesThatShouldHaveAliasAsTitle

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.validation.AbstractRule
      |
      +-
```
org.tanjakostic.jcleancim.validation.ClassValidator.Iec61850ClassesThatShouldHaveAliasAsTitle

All Implemented Interfaces:
SimpleRule, Rule

---

public static class **ClassValidator.Iec61850ClassesThatShouldHaveAliasAsTitle**
extends AbstractRule
implements Rule, SimpleRule

Constructor Summary

public	Iec61850ClassesThatShouldHaveAliasAsTitle()

Method Summary

java.util.EnumSet	getApplicability()
void	validate(UmlClass o, ModelIssues issues)

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule

createIssue, createIssue, createIssue, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from interface org.tanjakostic.jcleancim.validation.SimpleRule

validate

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Constructors

Iec61850ClassesThatShouldHaveAliasAsTitle

```
public Iec61850ClassesThatShouldHaveAliasAsTitle()
```

Methods

getApplicability

```
public java.util.EnumSet getApplicability()
```

validate

```
public void validate(UmlClass o,
         ModelIssues issues)
```

org.tanjakostic.jcleancim.validation
Class
ClassValidator.Iec61850ClassesThatShouldHaveTaggedValuesForDocgen

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.validation.AbstractRule
      |
      +-
```
org.tanjakostic.jcleancim.validation.ClassValidator.Iec61850ClassesThatShouldHaveTaggedValuesForDocgen

All Implemented Interfaces:
SimpleRule, Rule

---

public static class **ClassValidator.Iec61850ClassesThatShouldHaveTaggedValuesForDocgen**
extends AbstractRule
implements Rule, SimpleRule

Constructor Summary

public	Iec61850ClassesThatShouldHaveTaggedValuesForDocgen()

Method Summary

java.util.EnumSet	getApplicability()
void	validate(UmlClass o, ModelIssues issues)

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule

createIssue, createIssue, createIssue, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from interface org.tanjakostic.jcleancim.validation.SimpleRule

validate

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Constructors

Iec61850ClassesThatShouldHaveTaggedValuesForDocgen

```
public Iec61850ClassesThatShouldHaveTaggedValuesForDocgen()
```

Methods

getApplicability

```
public java.util.EnumSet getApplicability()
```

validate

```
public void validate(UmlClass o,
         ModelIssues issues)
```

org.tanjakostic.jcleancim.validation
Class ClassValidator.CimClassesNeverUsedInRelationships

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.validation.AbstractRule
      |
      +-
```
org.tanjakostic.jcleancim.validation.ClassValidator.CimClassesNeverUsedInRelationships

All Implemented Interfaces:
SimpleRule, Rule

---

public static class **ClassValidator.CimClassesNeverUsedInRelationships**
extends AbstractRule
implements Rule, SimpleRule

Constructor Summary

public	CimClassesNeverUsedInRelationships()

Method Summary

java.util.EnumSet	getApplicability()
void	validate(UmlClass o, ModelIssues issues)

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule

createIssue, createIssue, createIssue, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from interface org.tanjakostic.jcleancim.validation.SimpleRule

validate

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

---

(continued from last page)

Constructors

CimClassesNeverUsedInRelationships

```
public CimClassesNeverUsedInRelationships()
```

Methods

getApplicability

```
public java.util.EnumSet getApplicability()
```

---

validate

```
public void validate(UmlClass o,
         ModelIssues issues)
```

org.tanjakostic.jcleancim.validation
Class ClassValidator.ClassesWithUnallowedTagNames

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.validation.AbstractRule
      |
      +-
org.tanjakostic.jcleancim.validation.AbstractRule.AbstractRuleWithSubobjectsAndSkips
          |
          +-
org.tanjakostic.jcleancim.validation.AbstractRule.UmlObjectsWithUnallowedTagNames
              |
              +-
```

org.tanjakostic.jcleancim.validation.ClassValidator.ClassesWithUnallowedTagNames

All Implemented Interfaces:
Rule, SimpleRule

---

public static class **ClassValidator.ClassesWithUnallowedTagNames**
extends AbstractRule.UmlObjectsWithUnallowedTagNames

Constructor Summary

public	ClassesWithUnallowedTagNames()

Method Summary

java.util.EnumSet	getApplicability()

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule.UmlObjectsWithUnallowedTagNames

doValidate

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule.AbstractRuleWithSubobjectsAndSkips

doValidate, getSubObjects, skipSubobjectValidation, skipValidation, validate

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule

createIssue, createIssue, createIssue, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from interface `org.tanjakostic.jcleancim.validation.SimpleRule`
`validate`

Methods inherited from interface `org.tanjakostic.jcleancim.validation.Rule`
`getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis`

Constructors

ClassesWithUnallowedTagNames

```
public ClassesWithUnallowedTagNames()
```

Methods

getApplicability

```
public java.util.EnumSet getApplicability()
```

org.tanjakostic.jcleancim.validation
Class ClassValidator.Iec61850ClassesWithInvalidConstraints

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.validation.AbstractRule
      |
      +-
```
org.tanjakostic.jcleancim.validation.ClassValidator.Iec61850ClassesWithInvalidConstraints

All Implemented Interfaces:
SimpleRule, Rule

---

public static class **ClassValidator.Iec61850ClassesWithInvalidConstraints**
extends AbstractRule
implements Rule, SimpleRule

Constructor Summary

public	Iec61850ClassesWithInvalidConstraints()

Method Summary

java.util.EnumSet	getApplicability()
void	validate(UmlClass o, ModelIssues issues)

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule

createIssue, createIssue, createIssue, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from interface org.tanjakostic.jcleancim.validation.SimpleRule

validate

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Constructors

Iec61850ClassesWithInvalidConstraints

```
public Iec61850ClassesWithInvalidConstraints()
```

Methods

getApplicability

```
public java.util.EnumSet getApplicability()
```

validate

```
public void validate(UmlClass o,
         ModelIssues issues)
```

org.tanjakostic.jcleancim.validation
Class ClassValidator.Iec61850LNClassesWithSuperfluousConstraints

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.validation.AbstractRule
      |
      +-
```
org.tanjakostic.jcleancim.validation.ClassValidator.Iec61850LNClassesWithSuperfluousConstraints

All Implemented Interfaces:
SimpleRule, Rule

---

public static class **ClassValidator.Iec61850LNClassesWithSuperfluousConstraints**
extends AbstractRule
implements Rule, SimpleRule

Constructor Summary

public	Iec61850LNClassesWithSuperfluousConstraints()

Method Summary

java.util.EnumSet	getApplicability()
void	validate(UmlClass o, ModelIssues issues)

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule

createIssue, createIssue, createIssue, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from interface org.tanjakostic.jcleancim.validation.SimpleRule

validate

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Constructors

Iec61850LNClassesWithSuperfluousConstraints

```
public Iec61850LNClassesWithSuperfluousConstraints()
```

Methods

getApplicability

```
public java.util.EnumSet getApplicability()
```

validate

```
public void validate(UmlClass o,
         ModelIssues issues)
```

org.tanjakostic.jcleancim.validation
Class ClassValidator.Iec61850ClassesWithMissingCondIDTextInConstraints

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.validation.AbstractRule
      |
      +-
```
org.tanjakostic.jcleancim.validation.ClassValidator.Iec61850ClassesWithMissingCondIDTextInConstraints

All Implemented Interfaces:
SimpleRule, Rule

---

public static class **ClassValidator.Iec61850ClassesWithMissingCondIDTextInConstraints**
extends AbstractRule
implements Rule, SimpleRule

Constructor Summary

public	Iec61850ClassesWithMissingCondIDTextInConstraints()

Method Summary

java.util.EnumSet	getApplicability()
void	validate(UmlClass o, ModelIssues issues)

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule

createIssue, createIssue, createIssue, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from interface org.tanjakostic.jcleancim.validation.SimpleRule

validate

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Constructors

Iec61850ClassesWithMissingCondIDTextInConstraints

```
public Iec61850ClassesWithMissingCondIDTextInConstraints()
```

Methods

getApplicability

```
public java.util.EnumSet getApplicability()
```

validate

```
public void validate(UmlClass o,
         ModelIssues issues)
```

org.tanjakostic.jcleancim.validation
Class ClassValidator.CimDatatypeClassesWithInvalidAttributes

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.validation.AbstractRule
      |
      +-
```
org.tanjakostic.jcleancim.validation.ClassValidator.CimDatatypeClassesWithInvalidAttributes

All Implemented Interfaces:
SimpleRule, Rule

---

public static class **ClassValidator.CimDatatypeClassesWithInvalidAttributes**
extends AbstractRule
implements Rule, SimpleRule

Constructor Summary

public	CimDatatypeClassesWithInvalidAttributes()

Method Summary

java.util.EnumSet	getApplicability()
void	validate(UmlClass o, ModelIssues issues)

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule

createIssue, createIssue, createIssue, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from interface org.tanjakostic.jcleancim.validation.SimpleRule

validate

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Constructors

CimDatatypeClassesWithInvalidAttributes

```
public CimDatatypeClassesWithInvalidAttributes()
```

Methods

getApplicability

```
public java.util.EnumSet getApplicability()
```

validate

```
public void validate(UmlClass o,
         ModelIssues issues)
```

org.tanjakostic.jcleancim.validation
Class ClassValidator.ClassesMissingDoc

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.validation.AbstractRule
    |
    +-
org.tanjakostic.jcleancim.validation.AbstractRule.AbstractRuleWithSubobjectsAndSkips
      |
      +-org.tanjakostic.jcleancim.validation.AbstractRule.UmlObjectsMissingDoc
        |
        +-org.tanjakostic.jcleancim.validation.ClassValidator.ClassesMissingDoc
```

All Implemented Interfaces:
Rule, SimpleRule

---

public static class **ClassValidator.ClassesMissingDoc**
extends AbstractRule.UmlObjectsMissingDoc

Constructor Summary

public	ClassesMissingDoc()

Method Summary

java.util.EnumSet	getApplicability()
boolean	skipValidation(UmlClass o)

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule.UmlObjectsMissingDoc

doValidate

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule.AbstractRuleWithSubobjectsAndSkips

doValidate, getSubObjects, skipSubobjectValidation, skipValidation, validate

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule

createIssue, createIssue, createIssue, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from interface `org.tanjakostic.jcleancim.validation.SimpleRule`
`validate`

Methods inherited from interface `org.tanjakostic.jcleancim.validation.Rule`
`getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis`

Constructors

ClassesMissingDoc

```
public ClassesMissingDoc()
```

Methods

getApplicability

```
public java.util.EnumSet getApplicability()
```

skipValidation

```
protected boolean skipValidation(UmlClass o)
```

This default implementation returns false (no skipping); override if main object doesn't need validation.

org.tanjakostic.jcleancim.validation
Class ClassValidator.ClassesWithBadDocStart

```
java.lang.Object
  |-org.tanjakostic.jcleancim.validation.AbstractRule
    |-
org.tanjakostic.jcleancim.validation.AbstractRule.AbstractRuleWithSubobjectsAndSkips
      |-org.tanjakostic.jcleancim.validation.AbstractRule.UmlObjectsWithBadDocStart
        |-
org.tanjakostic.jcleancim.validation.ClassValidator.ClassesWithBadDocStart
```

All Implemented Interfaces:
Rule, SimpleRule

---

public static class **ClassValidator.ClassesWithBadDocStart**
extends AbstractRule.UmlObjectsWithBadDocStart

Constructor Summary

public	ClassesWithBadDocStart()

Method Summary

java.util.EnumSet	getApplicability()

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule.UmlObjectsWithBadDocStart

doValidate

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule.AbstractRuleWithSubobjectsAndSkips

doValidate, getSubObjects, skipSubobjectValidation, skipValidation, validate

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule

createIssue, createIssue, createIssue, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from interface org.tanjakostic.jcleancim.validation.SimpleRule

validate

Methods inherited from interface `org.tanjakostic.jcleancim.validation.Rule`
`getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis`

Constructors

ClassesWithBadDocStart

```
public ClassesWithBadDocStart()
```

Methods

getApplicability

```
public java.util.EnumSet getApplicability()
```

org.tanjakostic.jcleancim.validation
Class ClassValidator.ClassesWithBadDocEnd

```
java.lang.Object
  |-org.tanjakostic.jcleancim.validation.AbstractRule
    |-
org.tanjakostic.jcleancim.validation.AbstractRule.AbstractRuleWithSubobjectsAndSkips
      |-org.tanjakostic.jcleancim.validation.AbstractRule.UmlObjectsWithBadDocEnd
        |-org.tanjakostic.jcleancim.validation.ClassValidator.ClassesWithBadDocEnd
```

All Implemented Interfaces:
Rule, SimpleRule

---

public static class **ClassValidator.ClassesWithBadDocEnd**
extends AbstractRule.UmlObjectsWithBadDocEnd

Constructor Summary

public	ClassesWithBadDocEnd()

Method Summary

java.util.EnumSet	getApplicability()

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule.UmlObjectsWithBadDocEnd

doValidate

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule.AbstractRuleWithSubobjectsAndSkips

doValidate, getSubObjects, skipSubobjectValidation, skipValidation, validate

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule

createIssue, createIssue, createIssue, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from interface org.tanjakostic.jcleancim.validation.SimpleRule

validate

Methods inherited from interface `org.tanjakostic.jcleancim.validation.Rule`
`getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis`

Constructors

ClassesWithBadDocEnd

```
public ClassesWithBadDocEnd()
```

Methods

getApplicability

```
public java.util.EnumSet getApplicability()
```

org.tanjakostic.jcleancim.validation
Class ClassValidator.ClassesWithBadCharacterInName

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.validation.AbstractRule
      |
      +-
org.tanjakostic.jcleancim.validation.AbstractRule.AbstractRuleWithSubobjectsAndSkips
          |
          +-
org.tanjakostic.jcleancim.validation.AbstractRule.UmlObjectsWithBadCharacterInName
              |
              +-
```

org.tanjakostic.jcleancim.validation.ClassValidator.ClassesWithBadCharacterInName

All Implemented Interfaces:
Rule, SimpleRule

---

public static class **ClassValidator.ClassesWithBadCharacterInName**
extends AbstractRule.UmlObjectsWithBadCharacterInName

Constructor Summary

public	ClassesWithBadCharacterInName()

Method Summary

java.util.EnumSet	getApplicability()
InvalidCharactersFinder	getInvalidCharacterFinder(UmlObject o)

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule.UmlObjectsWithBadCharacterInName

doValidate, getInvalidCharacterFinder

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule.AbstractRuleWithSubobjectsAndSkips

doValidate, getSubObjects, skipSubobjectValidation, skipValidation, validate

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule

createIssue, createIssue, createIssue, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from interface org.tanjakostic.jcleancim.validation.SimpleRule
validate

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule
getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Constructors

ClassesWithBadCharacterInName

public **ClassesWithBadCharacterInName**()

Methods

getApplicability

public java.util.EnumSet **getApplicability**()

getInvalidCharacterFinder

protected InvalidCharactersFinder **getInvalidCharacterFinder**(UmlObject o)

org.tanjakostic.jcleancim.validation
Class ClassValidator.CimClassesNameStartingWithLowerCase

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.validation.AbstractRule
      |
      +-
```
org.tanjakostic.jcleancim.validation.ClassValidator.CimClassesNameStartingWithLowerCase

All Implemented Interfaces:
SimpleRule, Rule

---

public static class **ClassValidator.CimClassesNameStartingWithLowerCase**
extends AbstractRule
implements Rule, SimpleRule

Constructor Summary

public	CimClassesNameStartingWithLowerCase()

Method Summary

java.util.EnumSet	getApplicability()
void	validate(UmlClass o, ModelIssues issues)

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule

createIssue, createIssue, createIssue, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from interface org.tanjakostic.jcleancim.validation.SimpleRule

validate

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

---

(continued from last page)

Constructors

CimClassesNameStartingWithLowerCase

```
public CimClassesNameStartingWithLowerCase()
```

Methods

getApplicability

```
public java.util.EnumSet getApplicability()
```

---

validate

```
public void validate(UmlClass o,
         ModelIssues issues)
```

org.tanjakostic.jcleancim.validation
Class ClassValidator.CimClassesNameShouldBeSingular

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.validation.AbstractRule
      |
      +-
```
org.tanjakostic.jcleancim.validation.ClassValidator.CimClassesNameShouldBeSingular

All Implemented Interfaces:
SimpleRule, Rule

---

public static class **ClassValidator.CimClassesNameShouldBeSingular**
extends AbstractRule
implements Rule, SimpleRule

Constructor Summary

public	CimClassesNameShouldBeSingular()

Method Summary

java.util.EnumSet	getApplicability()
void	validate(UmlClass o, ModelIssues issues)

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule

createIssue, createIssue, createIssue, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from interface org.tanjakostic.jcleancim.validation.SimpleRule

validate

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

---

(continued from last page)

Constructors

CimClassesNameShouldBeSingular

```
public CimClassesNameShouldBeSingular()
```

Methods

getApplicability

```
public java.util.EnumSet getApplicability()
```

---

validate

```
public void validate(UmlClass o,
         ModelIssues issues)
```

org.tanjakostic.jcleancim.validation
Class ClassValidator.Iec61850LNClassesInWrongGroup

```
java.lang.Object
  |-org.tanjakostic.jcleancim.validation.AbstractRule
    |-
```
org.tanjakostic.jcleancim.validation.ClassValidator.Iec61850LNClassesInWrongGroup

All Implemented Interfaces:
SimpleRule, Rule

---

public static class **ClassValidator.Iec61850LNClassesInWrongGroup**
extends AbstractRule
implements Rule, SimpleRule

Constructor Summary

public	Iec61850LNClassesInWrongGroup()

Method Summary

java.util.EnumSet	getApplicability()
void	validate(UmlClass o, ModelIssues issues)

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule

createIssue, createIssue, createIssue, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from interface org.tanjakostic.jcleancim.validation.SimpleRule

validate

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

---

(continued from last page)

Constructors

Iec61850LNClassesInWrongGroup

```
public Iec61850LNClassesInWrongGroup()
```

Methods

getApplicability

```
public java.util.EnumSet getApplicability()
```

---

validate

```
public void validate(UmlClass o,
         ModelIssues issues)
```

org.tanjakostic.jcleancim.validation
Class ClassValidator.Iec61850LNClassesMalformedName

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.validation.AbstractRule
      |
      +-
```
org.tanjakostic.jcleancim.validation.ClassValidator.Iec61850LNClassesMalformedName

All Implemented Interfaces:
SimpleRule, Rule

---

public static class **ClassValidator.Iec61850LNClassesMalformedName**
extends AbstractRule
implements Rule, SimpleRule

Constructor Summary

public	Iec61850LNClassesMalformedName()

Method Summary

java.util.EnumSet	getApplicability()
void	validate(UmlClass o, ModelIssues issues)

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule

createIssue, createIssue, createIssue, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from interface org.tanjakostic.jcleancim.validation.SimpleRule

validate

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

---

(continued from last page)

Constructors

Iec61850LNClassesMalformedName

```
public Iec61850LNClassesMalformedName()
```

Methods

getApplicability

```
public java.util.EnumSet getApplicability()
```

---

validate

```
public void validate(UmlClass o,
         ModelIssues issues)
```

org.tanjakostic.jcleancim.validation
Class ClassValidator.EnumClassesWithSomeCodesMissing

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.validation.AbstractRule
      |
      +-
```
org.tanjakostic.jcleancim.validation.ClassValidator.EnumClassesWithSomeCodesMissing

All Implemented Interfaces:
SimpleRule, Rule

---

public static class **ClassValidator.EnumClassesWithSomeCodesMissing**
extends AbstractRule
implements Rule, SimpleRule

Constructor Summary

public	EnumClassesWithSomeCodesMissing()

Method Summary

java.util.EnumSet	getApplicability()
void	validate(UmlClass o, ModelIssues issues)

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule

createIssue, createIssue, createIssue, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from interface org.tanjakostic.jcleancim.validation.SimpleRule

validate

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

---

(continued from last page)

Constructors

EnumClassesWithSomeCodesMissing

```
public EnumClassesWithSomeCodesMissing()
```

Methods

getApplicability

```
public java.util.EnumSet getApplicability()
```

---

validate

```
public void validate(UmlClass o,
         ModelIssues issues)
```

org.tanjakostic.jcleancim.validation
Class ClassValidator.EnumClassesWithDuplicateCodes

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.validation.AbstractRule
      |
      +-
```
org.tanjakostic.jcleancim.validation.ClassValidator.EnumClassesWithDuplicateCodes

All Implemented Interfaces:
SimpleRule, Rule

---

public static class **ClassValidator.EnumClassesWithDuplicateCodes**
extends AbstractRule
implements Rule, SimpleRule

Constructor Summary

public	EnumClassesWithDuplicateCodes()

Method Summary

java.util.EnumSet	getApplicability()
void	validate(UmlClass o, ModelIssues issues)

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule

createIssue, createIssue, createIssue, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from interface org.tanjakostic.jcleancim.validation.SimpleRule

validate

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

---

(continued from last page)

Constructors

EnumClassesWithDuplicateCodes

```
public EnumClassesWithDuplicateCodes()
```

Methods

getApplicability

```
public java.util.EnumSet getApplicability()
```

---

validate

```
public void validate(UmlClass o,
         ModelIssues issues)
```

org.tanjakostic.jcleancim.validation
Class ClassValidator.ClassesWithSameName

```
java.lang.Object
  |-org.tanjakostic.jcleancim.validation.AbstractRule
    |-org.tanjakostic.jcleancim.validation.ClassValidator.ClassesWithSameName
```

All Implemented Interfaces:
CrossRule, Rule

---

public static class **ClassValidator.ClassesWithSameName**
extends AbstractRule
implements Rule, CrossRule

Constructor Summary

public	ClassesWithSameName(java.util.Collection allClasses)

Method Summary

java.util.EnumSet	getApplicability()
java.util.Collection	getObjsToTestAgainst()
void	validate(java.util.List classes, ModelIssues issues)

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule

createIssue, createIssue, createIssue, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from interface org.tanjakostic.jcleancim.validation.CrossRule

getObjsToTestAgainst, validate

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Constructors

ClassesWithSameName

```
public ClassesWithSameName(java.util.Collection allClasses)
```

Methods

getApplicability

```
public java.util.EnumSet getApplicability()
```

getObjsToTestAgainst

```
public java.util.Collection getObjsToTestAgainst()
```

validate

```
public void validate(java.util.List classes,
         ModelIssues issues)
```

org.tanjakostic.jcleancim.validation
Class ClassValidator.CimClassesNeverUsedAsTypeForAttribute

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.validation.AbstractRule
      |
      +-
```

org.tanjakostic.jcleancim.validation.ClassValidator.CimClassesNeverUsedAsTypeForAttribute

All Implemented Interfaces:
CrossRule, Rule

---

public static class **ClassValidator.CimClassesNeverUsedAsTypeForAttribute**
extends AbstractRule
implements Rule, CrossRule

Constructor Summary

public	CimClassesNeverUsedAsTypeForAttribute(java.util.List scopedWrtUmlAttributes)

Method Summary

java.util.EnumSet	getApplicability()
java.util.Collection	getObjsToTestAgainst()
void	validate(java.util.List classes, ModelIssues issues)

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule

createIssue, createIssue, createIssue, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from interface org.tanjakostic.jcleancim.validation.CrossRule

getObjsToTestAgainst, validate

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

```
getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity,
logDiagnosis
```

Constructors

CimClassesNeverUsedAsTypeForAttribute

```
public CimClassesNeverUsedAsTypeForAttribute(java.util.List scopedWrtUmlAttributes)
```

Methods

getObjsToTestAgainst

```
public java.util.Collection getObjsToTestAgainst()
```

getApplicability

```
public java.util.EnumSet getApplicability()
```

validate

```
public void validate(java.util.List classes,
         ModelIssues issues)
```

org.tanjakostic.jcleancim.validation
Interface CrossRule

All Known Implementing Classes:
Iec61850ConditionLiteralsNeverUsedAsConstraints, Iec61850DOAttributesWithSameNameDifferentType, Iec61850DOAbbreviationLiteralsNeverUsedInDOName, Iec61850DOAbbreviationLiteralsDuplicateDescription, Iec61850DOAbbreviationLiteralsDuplicateName, CimClassesNeverUsedAsTypeForAttribute, ClassesWithSameName, PackagesWithSameName

public interface **CrossRule**
extends Rule

Rule that applies to a collection of `UmlObject`-s against a collection of `UmlObject`-s potentially of different type.
Parameters:
`T` - rule applies to collection of these `UmlObject`-s

Method Summary

abstract java.util.Collection	`getObjsToTestAgainst()` Returns list of objects against which `validate(List, ModelIssues)` works.
abstract void	`validate(java.util.List objs, ModelIssues toCollect)` Applies the validation criteria to a **non-empty** list of `objs`, and creates problems for invalid ones and adds them to `toCollect`.

Methods inherited from interface `org.tanjakostic.jcleancim.validation.Rule`

`getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis`

Methods

validate

```
public abstract void validate(java.util.List objs,
         ModelIssues toCollect)
```

Applies the validation criteria to a **non-empty** list of `objs`, and creates problems for invalid ones and adds them to `toCollect`.

getObjsToTestAgainst

```
public abstract java.util.Collection getObjsToTestAgainst()
```

Returns list of objects against which `validate(List, ModelIssues)` works.

org.tanjakostic.jcleancim.validation
Class DependencyValidator

```
java.lang.Object
  |-org.tanjakostic.jcleancim.validation.AbstractValidator
    |-org.tanjakostic.jcleancim.validation.DependencyValidator
```

public class **DependencyValidator**
extends AbstractValidator

Validates (hand-drawn) dependencies.

Nested Class Summary

class	DependencyValidator.DependenciesWithUnallowedDirection DependencyValidator.DependenciesWithUnallowedDirection
class	DependencyValidator.DependenciesWithUnallowedStereotype DependencyValidator.DependenciesWithUnallowedStereotype
class	DependencyValidator.DependenciesWithUnallowedTagNames DependencyValidator.DependenciesWithUnallowedTagNames

Method Summary

boolean	enabled()
java.util.List	getScopedUmlObjects()

Methods inherited from class `org.tanjakostic.jcleancim.validation.AbstractValidator`

addCrossRule, addSimpleRule, displayAllAvailableRuleNames, displayAvailableRuleNames, enabled, getAllCrossRules, getAllRules, getAllSimpleRules, getCfg, getCheckedCrossRules, getCheckedRules, getCheckedSimpleRules, getCollectedIssues, getScopedUmlObjects, validate

Methods inherited from class `java.lang.Object`

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods

enabled

public boolean **enabled**()

Returns whether the validation for this validator has been enabled (by configuration).

(continued from last page)

getScopedUmlObjects

```
public java.util.List getScopedUmlObjects()
```

Returns the elements retained for validation, for the configured scope.

org.tanjakostic.jcleancim.validation
Class DependencyValidator.DependenciesWithUnallowedStereotype

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.validation.AbstractRule
      |
      +-
org.tanjakostic.jcleancim.validation.AbstractRule.AbstractRuleWithSubobjectsAndSkips
          |
          +-
org.tanjakostic.jcleancim.validation.AbstractRule.UmlObjectsWithUnallowedStereotype
              |
              +-
```

org.tanjakostic.jcleancim.validation.DependencyValidator.DependenciesWithUnallowedStereotype

All Implemented Interfaces:
Rule, SimpleRule

---

public static class **DependencyValidator.DependenciesWithUnallowedStereotype**
extends AbstractRule.UmlObjectsWithUnallowedStereotype

Constructor Summary

public	DependenciesWithUnallowedStereotype()

Method Summary

java.util.EnumSet	getApplicability()

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule.UmlObjectsWithUnallowedStereotype

doValidate

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule.AbstractRuleWithSubobjectsAndSkips

doValidate, getSubObjects, skipSubobjectValidation, skipValidation, validate

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule

createIssue, createIssue, createIssue, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from interface `org.tanjakostic.jcleancim.validation.SimpleRule`
`validate`

Methods inherited from interface `org.tanjakostic.jcleancim.validation.Rule`
`getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis`

Constructors

DependenciesWithUnallowedStereotype

```
public DependenciesWithUnallowedStereotype()
```

Methods

getApplicability

```
public java.util.EnumSet getApplicability()
```

org.tanjakostic.jcleancim.validation
Class DependencyValidator.DependenciesWithUnallowedDirection

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.validation.AbstractRule
      |
      +-
```
org.tanjakostic.jcleancim.validation.DependencyValidator.DependenciesWithUnallowedDirection

All Implemented Interfaces:
SimpleRule, Rule

---

public static class **DependencyValidator.DependenciesWithUnallowedDirection**
extends AbstractRule
implements Rule, SimpleRule

Constructor Summary

public	DependenciesWithUnallowedDirection()

Method Summary

java.util.EnumSet	getApplicability()
void	validate(UmlDependency d, ModelIssues issues)

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule

createIssue, createIssue, createIssue, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from interface org.tanjakostic.jcleancim.validation.SimpleRule

validate

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Constructors

DependenciesWithUnallowedDirection

```
public DependenciesWithUnallowedDirection()
```

Methods

getApplicability

```
public java.util.EnumSet getApplicability()
```

validate

```
public void validate(UmlDependency d,
         ModelIssues issues)
```

org.tanjakostic.jcleancim.validation
Class DependencyValidator.DependenciesWithUnallowedTagNames

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.validation.AbstractRule
      |
      +-
org.tanjakostic.jcleancim.validation.AbstractRule.AbstractRuleWithSubobjectsAndSkips
          |
          +-
org.tanjakostic.jcleancim.validation.AbstractRule.UmlObjectsWithUnallowedTagNames
              |
              +-
```
org.tanjakostic.jcleancim.validation.DependencyValidator.DependenciesWithUnallowedTagNames

All Implemented Interfaces:
Rule, SimpleRule

---

public static class **DependencyValidator.DependenciesWithUnallowedTagNames**
extends AbstractRule.UmlObjectsWithUnallowedTagNames

Constructor Summary

public	DependenciesWithUnallowedTagNames()

Method Summary

java.util.EnumSet	getApplicability()

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule.UmlObjectsWithUnallowedTagNames

doValidate

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule.AbstractRuleWithSubobjectsAndSkips

doValidate, getSubObjects, skipSubobjectValidation, skipValidation, validate

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule

createIssue, createIssue, createIssue, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from interface `org.tanjakostic.jcleancim.validation.SimpleRule`
`validate`

Methods inherited from interface `org.tanjakostic.jcleancim.validation.Rule`
`getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis`

Constructors

DependenciesWithUnallowedTagNames

```
public DependenciesWithUnallowedTagNames()
```

Methods

getApplicability

```
public java.util.EnumSet getApplicability()
```

org.tanjakostic.jcleancim.validation
Class DiagramValidator

```
java.lang.Object
  +-org.tanjakostic.jcleancim.validation.AbstractValidator
    +-org.tanjakostic.jcleancim.validation.DiagramValidator
```

public class **DiagramValidator**
extends AbstractValidator

Validates diagrams.

Nested Class Summary

class	DiagramValidator.DiagramsMissingDoc DiagramValidator.DiagramsMissingDoc
class	DiagramValidator.DiagramsWithBadCharacterInName DiagramValidator.DiagramsWithBadCharacterInName
class	DiagramValidator.DiagramsWithBadDocEnd DiagramValidator.DiagramsWithBadDocEnd
class	DiagramValidator.DiagramsWithBadDocStart DiagramValidator.DiagramsWithBadDocStart
class	DiagramValidator.DiagramsWithBadOrientation DiagramValidator.DiagramsWithBadOrientation
class	DiagramValidator.DiagramsWithUnallowedStereotype DiagramValidator.DiagramsWithUnallowedStereotype

Method Summary

boolean	enabled()
java.util.List	getScopedUmlObjects()

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractValidator

addCrossRule, addSimpleRule, displayAllAvailableRuleNames, displayAvailableRuleNames, enabled, getAllCrossRules, getAllRules, getAllSimpleRules, getCfg, getCheckedCrossRules, getCheckedRules, getCheckedSimpleRules, getCollectedIssues, getScopedUmlObjects, validate

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

(continued from last page)

Methods

enabled

`public boolean **enabled**()`

Returns whether the validation for this validator has been enabled (by configuration).

getScopedUmlObjects

`public java.util.List **getScopedUmlObjects**()`

Returns the elements retained for validation, for the configured scope.

org.tanjakostic.jcleancim.validation
Class DiagramValidator.DiagramsWithBadOrientation

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.validation.AbstractRule
      |
      +-
```
org.tanjakostic.jcleancim.validation.DiagramValidator.DiagramsWithBadOrientation

All Implemented Interfaces:
SimpleRule, Rule

---

public static class **DiagramValidator.DiagramsWithBadOrientation**
extends AbstractRule
implements Rule, SimpleRule

Constructor Summary

public	DiagramsWithBadOrientation()

Method Summary

java.util.EnumSet	getApplicability()
void	validate(UmlDiagram d, ModelIssues issues)

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule

createIssue, createIssue, createIssue, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from interface org.tanjakostic.jcleancim.validation.SimpleRule

validate

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

---

(continued from last page)

Constructors

DiagramsWithBadOrientation

```
public DiagramsWithBadOrientation()
```

Methods

getApplicability

```
public java.util.EnumSet getApplicability()
```

---

validate

```
public void validate(UmlDiagram d,
         ModelIssues issues)
```

Matches if diagram orientation is not portrait.

org.tanjakostic.jcleancim.validation
Class DiagramValidator.DiagramsWithUnallowedStereotype

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.validation.AbstractRule
      |
      +-
org.tanjakostic.jcleancim.validation.AbstractRule.AbstractRuleWithSubobjectsAndSkips
          |
          +-
org.tanjakostic.jcleancim.validation.AbstractRule.UmlObjectsWithUnallowedStereotype
              |
              +-
```

org.tanjakostic.jcleancim.validation.DiagramValidator.DiagramsWithUnallowedStereotype

All Implemented Interfaces:
Rule, SimpleRule

---

public static class **DiagramValidator.DiagramsWithUnallowedStereotype**
extends AbstractRule.UmlObjectsWithUnallowedStereotype

Constructor Summary

public	DiagramsWithUnallowedStereotype()

Method Summary

java.util.EnumSet	getApplicability()

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule.UmlObjectsWithUnallowedStereotype

doValidate

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule.AbstractRuleWithSubobjectsAndSkips

doValidate, getSubObjects, skipSubobjectValidation, skipValidation, validate

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule

createIssue, createIssue, createIssue, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from interface `org.tanjakostic.jcleancim.validation.SimpleRule`
`validate`

Methods inherited from interface `org.tanjakostic.jcleancim.validation.Rule`
`getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis`

Constructors

DiagramsWithUnallowedStereotype

```
public DiagramsWithUnallowedStereotype()
```

Methods

getApplicability

```
public java.util.EnumSet getApplicability()
```

org.tanjakostic.jcleancim.validation
Class DiagramValidator.DiagramsMissingDoc

```
java.lang.Object
  |-org.tanjakostic.jcleancim.validation.AbstractRule
    |-
org.tanjakostic.jcleancim.validation.AbstractRule.AbstractRuleWithSubobjectsAndSkips
      |-org.tanjakostic.jcleancim.validation.AbstractRule.UmlObjectsMissingDoc
        |-org.tanjakostic.jcleancim.validation.DiagramValidator.DiagramsMissingDoc
```

All Implemented Interfaces:
Rule, SimpleRule

---

public static class **DiagramValidator.DiagramsMissingDoc**
extends AbstractRule.UmlObjectsMissingDoc

Constructor Summary

public	DiagramsMissingDoc()

Method Summary

java.util.EnumSet	getApplicability()

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule.UmlObjectsMissingDoc

doValidate

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule.AbstractRuleWithSubobjectsAndSkips

doValidate, getSubObjects, skipSubobjectValidation, skipValidation, validate

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule

createIssue, createIssue, createIssue, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from interface org.tanjakostic.jcleancim.validation.SimpleRule

validate

Methods inherited from interface `org.tanjakostic.jcleancim.validation.Rule`
`getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis`

Constructors

DiagramsMissingDoc

```
public DiagramsMissingDoc()
```

Methods

getApplicability

```
public java.util.EnumSet getApplicability()
```

org.tanjakostic.jcleancim.validation
Class DiagramValidator.DiagramsWithBadDocStart

```
java.lang.Object
  |-org.tanjakostic.jcleancim.validation.AbstractRule
    |-
org.tanjakostic.jcleancim.validation.AbstractRule.AbstractRuleWithSubobjectsAndSkips
      |-org.tanjakostic.jcleancim.validation.AbstractRule.UmlObjectsWithBadDocStart
        |-
org.tanjakostic.jcleancim.validation.DiagramValidator.DiagramsWithBadDocStart
```

All Implemented Interfaces:
Rule, SimpleRule

---

public static class **DiagramValidator.DiagramsWithBadDocStart**
extends AbstractRule.UmlObjectsWithBadDocStart

Constructor Summary

public	DiagramsWithBadDocStart()

Method Summary

java.util.EnumSet	getApplicability()

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule.UmlObjectsWithBadDocStart

doValidate

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule.AbstractRuleWithSubobjectsAndSkips

doValidate, getSubObjects, skipSubobjectValidation, skipValidation, validate

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule

createIssue, createIssue, createIssue, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from interface org.tanjakostic.jcleancim.validation.SimpleRule

validate

Methods inherited from interface `org.tanjakostic.jcleancim.validation.Rule`
`getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis`

Constructors

DiagramsWithBadDocStart

`public` **`DiagramsWithBadDocStart`**`()`

Methods

getApplicability

`public java.util.EnumSet` **`getApplicability`**`()`

org.tanjakostic.jcleancim.validation
Class DiagramValidator.DiagramsWithBadDocEnd

```
java.lang.Object
  |-org.tanjakostic.jcleancim.validation.AbstractRule
    |-
org.tanjakostic.jcleancim.validation.AbstractRule.AbstractRuleWithSubobjectsAndSkips
      |-org.tanjakostic.jcleancim.validation.AbstractRule.UmlObjectsWithBadDocEnd
        |-
org.tanjakostic.jcleancim.validation.DiagramValidator.DiagramsWithBadDocEnd
```

All Implemented Interfaces:
Rule, SimpleRule

---

public static class **DiagramValidator.DiagramsWithBadDocEnd**
extends AbstractRule.UmlObjectsWithBadDocEnd

Constructor Summary

public	DiagramsWithBadDocEnd()

Method Summary

java.util.EnumSet	getApplicability()

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule.UmlObjectsWithBadDocEnd

doValidate

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule.AbstractRuleWithSubobjectsAndSkips

doValidate, getSubObjects, skipSubobjectValidation, skipValidation, validate

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule

createIssue, createIssue, createIssue, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from interface org.tanjakostic.jcleancim.validation.SimpleRule

validate

Methods inherited from interface `org.tanjakostic.jcleancim.validation.Rule`
`getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis`

Constructors

DiagramsWithBadDocEnd

`public` **`DiagramsWithBadDocEnd`**`()`

Methods

getApplicability

`public java.util.EnumSet` **`getApplicability`**`()`

org.tanjakostic.jcleancim.validation
Class DiagramValidator.DiagramsWithBadCharacterInName

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.validation.AbstractRule
    |
    +-
org.tanjakostic.jcleancim.validation.AbstractRule.AbstractRuleWithSubobjectsAndSkips
      |
      +-
org.tanjakostic.jcleancim.validation.AbstractRule.UmlObjectsWithBadCharacterInName
        |
        +-
```
org.tanjakostic.jcleancim.validation.DiagramValidator.DiagramsWithBadCharacterInName

All Implemented Interfaces:
Rule, SimpleRule

---

public static class **DiagramValidator.DiagramsWithBadCharacterInName**
extends AbstractRule.UmlObjectsWithBadCharacterInName

Constructor Summary

public	DiagramsWithBadCharacterInName()

Method Summary

java.util.EnumSet	getApplicability()
InvalidCharactersFinder	getInvalidCharacterFinder(UmlObject o)
boolean	skipValidation(UmlDiagram o)

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule.UmlObjectsWithBadCharacterInName

doValidate, getInvalidCharacterFinder

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule.AbstractRuleWithSubobjectsAndSkips

doValidate, getSubObjects, skipSubobjectValidation, skipValidation, validate

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule

createIssue, createIssue, createIssue, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface `org.tanjakostic.jcleancim.validation.Rule`
`getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis`

Methods inherited from interface `org.tanjakostic.jcleancim.validation.SimpleRule`
`validate`

Methods inherited from interface `org.tanjakostic.jcleancim.validation.Rule`
`getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis`

Constructors

DiagramsWithBadCharacterInName

```
public DiagramsWithBadCharacterInName()
```

Methods

getApplicability

```
public java.util.EnumSet getApplicability()
```

skipValidation

```
protected boolean skipValidation(UmlDiagram o)
```

This default implementation returns false (no skipping); override if main object doesn't need validation.

getInvalidCharacterFinder

```
protected InvalidCharactersFinder getInvalidCharacterFinder(UmlObject o)
```

org.tanjakostic.jcleancim.validation
Class InvalidCharactersFinder

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.validation.InvalidCharactersFinder
```

public class **InvalidCharactersFinder**
extends java.lang.Object

Pattern compiled from the regular expression that will match characters invalid for the CIM tokens, and most of IEC 61850 tokens. So, if the matcher returns a match from this pattern, this will be the invalid character.

The valid token should start with a lower or upper case letter, and be followed by any number of lower or upper case letters or numbers.

Field Summary

public static final	NUM_UNDERSCORE_DASH_SPACE_COMMA
public static final	STRICT
public static final	STRICT_UNDERSCORE_DASH

Constructor Summary

public	InvalidCharactersFinder(java.lang.String regexExpression) Constructor.

Method Summary

java.util.List	findInvalidCharacters(java.lang.String input) Returns the list of characters in `input` that are invalid according to regular expression passed at creation of this instance, empty list if all characters are valid or `input` is null or empty string.
java.util.regex.Pattern	getCompiledRegexExpression()
java.lang.String	getRegexExpression()

Methods inherited from class `java.lang.Object`

`clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait`

Fields

(continued from last page)

STRICT

```
public static final org.tanjakostic.jcleancim.validation.InvalidCharactersFinder
STRICT
```

STRICT_UNDERSCORE_DASH

```
public static final org.tanjakostic.jcleancim.validation.InvalidCharactersFinder
STRICT_UNDERSCORE_DASH
```

NUM_UNDERSCORE_DASH_SPACE_COMMA

```
public static final org.tanjakostic.jcleancim.validation.InvalidCharactersFinder
NUM_UNDERSCORE_DASH_SPACE_COMMA
```

Constructors

InvalidCharactersFinder

```
public InvalidCharactersFinder(java.lang.String regexExpression)
```

Constructor.

Methods

getRegexExpression

```
public java.lang.String getRegexExpression()
```

getCompiledRegexExpression

```
public java.util.regex.Pattern getCompiledRegexExpression()
```

findInvalidCharacters

```
public java.util.List findInvalidCharacters(java.lang.String input)
```

Returns the list of characters in `input` that are invalid according to regular expression passed at creation of this instance, empty list if all characters are valid or `input` is null or empty string.

org.tanjakostic.jcleancim.validation
Class ModelIssue

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.validation.ModelIssue
```

public class **ModelIssue**
extends java.lang.Object

Model issue found during validation.

Constructor Summary

public	`ModelIssue(UmlObject subject, Rule rule)` Creates an issue without the evidence part and with toShortString() subject description.
public	`ModelIssue(UmlObject subject, Rule rule, java.lang.String subjectDescription)` Creates an issue without the evidence part.
public	`ModelIssue(UmlObject subject, Rule rule, java.lang.String subjectDescription, java.lang.String evidence, java.lang.String groupTag)` Constructor.

Method Summary

java.lang.String	`asCSV()` Returns the string representation suitable for comma-separated format.
static java.lang.String	`columnsAsCSV()` Returns the string representation of columns suitable for comma-separated format.
java.lang.String	`getCategory()`
java.lang.String	`getDiagnosisItem()` Returns the line of text with diagnosis as appropriate for logging; it will likely be deduced from `getGroupTag()`, `getEvidence()` and `getSubjectDescription()`.
java.lang.String	`getEvidence()` Returns the "proof of guilt"; may be null if obvious.
java.lang.String	`getGroupTag()` Returns potentially null/empty tag indicating relation with other issues of the same type.
java.lang.String	`getHowToFix()` Returns the suggestion on how to fix the problem.
java.lang.String	`getHypothesis()` Returns what the rule is enforcing (and what was violated).
java.lang.String	`getRuleName()` Returns the name of the rule that was violated, which resulted in this issue.

java.lang.String	getSeverity()
java.lang.String	getSubjectDescription() Returns the description of subject in this issue, sufficient to find it among all the objects in the model.
java.lang.String	getSubjectKind()
java.lang.String	getSubjectOwner()
java.lang.String	getSubjectQName()
java.lang.String	toString()

Methods inherited from class java.lang.Object
clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Constructors

ModelIssue

```
public ModelIssue(UmlObject subject,
                  Rule rule)
```

Creates an issue without the evidence part and with toShortString() subject description.

Parameters:

`subject` - non-null subject of this issue.
`rule` - non-null rule whole violation created this issue.

ModelIssue

```
public ModelIssue(UmlObject subject,
                  Rule rule,
                  java.lang.String subjectDescription)
```

Creates an issue without the evidence part.

Parameters:

`subject` - non-null subject of this issue.
`rule` - non-null rule whole violation created this issue.
`subjectDescription` - subject description; if null, using toShortString()

ModelIssue

```
public ModelIssue(UmlObject subject,
                  Rule rule,
                  java.lang.String subjectDescription,
                  java.lang.String evidence,
                  java.lang.String groupTag)
```

Constructor.

(continued from last page)

Parameters:

`subject` - non-null subject of this issue.
`rule` - non-null rule whose violation created this issue.
`subjectDescription` - subject description; if null, using toShortString().
`evidence` - (potentially null) evidence of the issue for subject.
`groupTag` - (potentially null) tag to indicate this issue is related to some other one; typically used in relation to duplicate names.

Methods

toString

`public java.lang.String` **`toString`**`()`

getSubjectOwner

`public java.lang.String` **`getSubjectOwner`**`()`

getSubjectQName

`public java.lang.String` **`getSubjectQName`**`()`

getSubjectKind

`public java.lang.String` **`getSubjectKind`**`()`

getCategory

`public java.lang.String` **`getCategory`**`()`

getSeverity

`public java.lang.String` **`getSeverity`**`()`

getRuleName

`public java.lang.String` **`getRuleName`**`()`

Returns the name of the rule that was violated, which resulted in this issue.

getHypothesis

`public java.lang.String` **`getHypothesis`**`()`

Returns what the rule is enforcing (and what was violated).

getHowToFix

`public java.lang.String **getHowToFix**()`

Returns the suggestion on how to fix the problem.

getSubjectDescription

`public java.lang.String **getSubjectDescription**()`

Returns the description of subject in this issue, sufficient to find it among all the objects in the model.

getEvidence

`public java.lang.String **getEvidence**()`

Returns the "proof of guilt"; may be null if obvious.

getGroupTag

`public java.lang.String **getGroupTag**()`

Returns potentially null/empty tag indicating relation with other issues of the same type.

getDiagnosisItem

`public java.lang.String **getDiagnosisItem**()`

Returns the line of text with diagnosis as appropriate for logging; it will likely be deduced from `getGroupTag()`, `getEvidence()` and `getSubjectDescription()`.

columnsAsCSV

`public static java.lang.String **columnsAsCSV**()`

Returns the string representation of columns suitable for comma-separated format.

asCSV

`public java.lang.String **asCSV**()`

Returns the string representation suitable for comma-separated format.

org.tanjakostic.jcleancim.validation
Class ModelIssues

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.validation.ModelIssues
```

public class **ModelIssues**
extends java.lang.Object

Constructor Summary

public	ModelIssues()

Method Summary

void	add(UmlObject subject, ModelIssue issue)
java.lang.String	asCSV()
java.util.List	getDiagnosisItems(java.lang.String ruleName)
java.util.List	getIssues()
java.util.Collection	getIssuesFor(UmlObject subject)
java.util.Collection	getSubjectsWithProblem(java.lang.String ruleName)

Methods inherited from class `java.lang.Object`

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Constructors

ModelIssues

public **ModelIssues**()

Methods

getIssues

public java.util.List **getIssues**()

getIssuesFor

```
public java.util.Collection getIssuesFor(UmlObject subject)
```

getSubjectsWithProblem

```
public java.util.Collection getSubjectsWithProblem(java.lang.String ruleName)
```

getDiagnosisItems

```
public java.util.List getDiagnosisItems(java.lang.String ruleName)
```

add

```
public void add(UmlObject subject,
         ModelIssue issue)
```

asCSV

```
public java.lang.String asCSV()
```

org.tanjakostic.jcleancim.validation
Class ModelValidator

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.validation.ModelValidator
```

public class **ModelValidator**
extends java.lang.Object

Creates validators per kind of UML element and allows to run the validation.

Field Summary

public static final	PROBLEMS_REPORT_PREFIX Value: **problemsReport-**

Constructor Summary

public	ModelValidator(UmlModel model)

Method Summary

void	logAllAvailableRuleNames(org.apache.log4j.Level level)
void	logAvailableRuleNamesWithCategoryAndSeverity(org.apache.log4j.Level level)
void	saveReport()
void	validate() Performs validation.

Methods inherited from class `java.lang.Object`

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Fields

PROBLEMS_REPORT_PREFIX

public static final java.lang.String **PROBLEMS_REPORT_PREFIX**

Constant value: **problemsReport-**

Constructors

(continued from last page)

ModelValidator

```
public ModelValidator(UmlModel model)
```

Methods

logAllAvailableRuleNames

```
public void logAllAvailableRuleNames(org.apache.log4j.Level level)
```

logAvailableRuleNamesWithCategoryAndSeverity

```
public void logAvailableRuleNamesWithCategoryAndSeverity(org.apache.log4j.Level level)
```

validate

```
public void validate()
```

Performs validation.

saveReport

```
public void saveReport()
```

org.tanjakostic.jcleancim.validation
Class OperationValidator

```
java.lang.Object
  |-org.tanjakostic.jcleancim.validation.AbstractValidator
    |-org.tanjakostic.jcleancim.validation.OperationValidator
```

public class **OperationValidator**
extends AbstractValidator

Validates operations.

Nested Class Summary

class	OperationValidator.OperationParametersMissingDoc OperationValidator.OperationParametersMissingDoc
class	OperationValidator.OperationParametersWithBadCharacterInName OperationValidator.OperationParametersWithBadCharacterInName
class	OperationValidator.OperationParametersWithBadDocEnd OperationValidator.OperationParametersWithBadDocEnd
class	OperationValidator.OperationParametersWithBadDocStart OperationValidator.OperationParametersWithBadDocStart
class	OperationValidator.OperationParametersWithUnallowedStereotype OperationValidator.OperationParametersWithUnallowedStereotype
class	OperationValidator.OperationParametersWithUnallowedTagNames OperationValidator.OperationParametersWithUnallowedTagNames
class	OperationValidator.OperationsMissingDoc OperationValidator.OperationsMissingDoc
class	OperationValidator.OperationsWithBadCharacterInName OperationValidator.OperationsWithBadCharacterInName
class	OperationValidator.OperationsWithBadDocEnd OperationValidator.OperationsWithBadDocEnd
class	OperationValidator.OperationsWithBadDocStart OperationValidator.OperationsWithBadDocStart
class	OperationValidator.OperationsWithInvalidArgTypeNull OperationValidator.OperationsWithInvalidArgTypeNull
class	OperationValidator.OperationsWithInvalidExcTypeNull OperationValidator.OperationsWithInvalidExcTypeNull
class	OperationValidator.OperationsWithInvalidReturnTypeNull OperationValidator.OperationsWithInvalidReturnTypeNull
class	OperationValidator.OperationsWithUnallowedStereotype OperationValidator.OperationsWithUnallowedStereotype

class	OperationValidator.OperationsWithUnallowedTagNames OperationValidator.OperationsWithUnallowedTagNames
class	OperationValidator.OperationsWithUpperCaseName OperationValidator.OperationsWithUpperCaseName

Method Summary

boolean	enabled()
java.util.List	getScopedUmlObjects()

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractValidator

addCrossRule, addSimpleRule, displayAllAvailableRuleNames, displayAvailableRuleNames, enabled, getAllCrossRules, getAllRules, getAllSimpleRules, getCfg, getCheckedCrossRules, getCheckedRules, getCheckedSimpleRules, getCollectedIssues, getScopedUmlObjects, validate

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods

enabled

public boolean **enabled**()

Returns whether the validation for this validator has been enabled (by configuration).

getScopedUmlObjects

public java.util.List **getScopedUmlObjects**()

Returns the elements retained for validation, for the configured scope.

org.tanjakostic.jcleancim.validation
Class OperationValidator.OperationsWithUpperCaseName

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.validation.AbstractRule
      |
      +-
```
org.tanjakostic.jcleancim.validation.OperationValidator.OperationsWithUpperCaseName

All Implemented Interfaces:
SimpleRule, Rule

---

public static class **OperationValidator.OperationsWithUpperCaseName**
extends AbstractRule
implements Rule, SimpleRule

Constructor Summary

public	OperationsWithUpperCaseName()

Method Summary

java.util.EnumSet	getApplicability()
void	validate(UmlOperation o, ModelIssues issues)

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule

createIssue, createIssue, createIssue, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from interface org.tanjakostic.jcleancim.validation.SimpleRule

validate

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

---

(continued from last page)

Constructors

OperationsWithUpperCaseName

```
public OperationsWithUpperCaseName()
```

Methods

getApplicability

```
public java.util.EnumSet getApplicability()
```

---

validate

```
public void validate(UmlOperation o,
         ModelIssues issues)
```

org.tanjakostic.jcleancim.validation
Class OperationValidator.OperationsWithUnallowedStereotype

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.validation.AbstractRule
      |
      +-
org.tanjakostic.jcleancim.validation.AbstractRule.AbstractRuleWithSubobjectsAndSkips
          |
          +-
org.tanjakostic.jcleancim.validation.AbstractRule.UmlObjectsWithUnallowedStereotype
              |
              +-
```
org.tanjakostic.jcleancim.validation.OperationValidator.OperationsWithUnallowedStereotype

All Implemented Interfaces:
Rule, SimpleRule

---

public static class **OperationValidator.OperationsWithUnallowedStereotype**
extends AbstractRule.UmlObjectsWithUnallowedStereotype

Constructor Summary

public	OperationsWithUnallowedStereotype()

Method Summary

java.util.EnumSet	getApplicability()

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule.UmlObjectsWithUnallowedStereotype

doValidate

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule.AbstractRuleWithSubobjectsAndSkips

doValidate, getSubObjects, skipSubobjectValidation, skipValidation, validate

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule

createIssue, createIssue, createIssue, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from interface `org.tanjakostic.jcleancim.validation.SimpleRule`
`validate`

Methods inherited from interface `org.tanjakostic.jcleancim.validation.Rule`
`getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis`

Constructors

OperationsWithUnallowedStereotype

```
public OperationsWithUnallowedStereotype()
```

Methods

getApplicability

```
public java.util.EnumSet getApplicability()
```

org.tanjakostic.jcleancim.validation
Class OperationValidator.OperationParametersWithUnallowedStereotype

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.validation.AbstractRule
      |
      +-
org.tanjakostic.jcleancim.validation.AbstractRule.AbstractRuleWithSubobjectsAndSkips
          |
          +-
org.tanjakostic.jcleancim.validation.AbstractRule.UmlObjectsWithUnallowedStereotype
              |
              +-
```
org.tanjakostic.jcleancim.validation.OperationValidator.OperationParametersWithUnallowedStereotype

All Implemented Interfaces:
Rule, SimpleRule

---

public static class **OperationValidator.OperationParametersWithUnallowedStereotype**
extends AbstractRule.UmlObjectsWithUnallowedStereotype

Constructor Summary

public	OperationParametersWithUnallowedStereotype()

Method Summary

java.util.EnumSet	getApplicability()
java.util.List	getSubObjects(UmlOperation op)

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule.UmlObjectsWithUnallowedStereotype

doValidate

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule.AbstractRuleWithSubobjectsAndSkips

doValidate, getSubObjects, skipSubobjectValidation, skipValidation, validate

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule

createIssue, createIssue, createIssue, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from interface org.tanjakostic.jcleancim.validation.SimpleRule
validate

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule
getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Constructors

OperationParametersWithUnallowedStereotype

public **OperationParametersWithUnallowedStereotype**()

Methods

getApplicability

public java.util.EnumSet **getApplicability**()

getSubObjects

protected java.util.List **getSubObjects**(UmlOperation op)

Used for validation of "sub-objects" (such as association ends and operation parameters) that do not have their own validators, but get validated through their container.

This default implementation returns empty list; subtypes having contained objects that need to be validated against this rule have to override this method.

org.tanjakostic.jcleancim.validation
Class OperationValidator.OperationsWithInvalidReturnTypeNull

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.validation.AbstractRule
      |
      +-
```
org.tanjakostic.jcleancim.validation.OperationValidator.OperationsWithInvalidReturnTypeNull

All Implemented Interfaces:
SimpleRule, Rule

---

public static class **OperationValidator.OperationsWithInvalidReturnTypeNull**
extends AbstractRule
implements Rule, SimpleRule

Constructor Summary

public	OperationsWithInvalidReturnTypeNull()

Method Summary

java.util.EnumSet	getApplicability()
void	validate(UmlOperation o, ModelIssues issues)

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule

createIssue, createIssue, createIssue, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from interface org.tanjakostic.jcleancim.validation.SimpleRule

validate

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Constructors

OperationsWithInvalidReturnTypeNull

```
public OperationsWithInvalidReturnTypeNull()
```

Methods

getApplicability

```
public java.util.EnumSet getApplicability()
```

validate

```
public void validate(UmlOperation o,
         ModelIssues issues)
```

org.tanjakostic.jcleancim.validation
Class OperationValidator.OperationsWithInvalidArgTypeNull

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.validation.AbstractRule
      |
      +-
```
org.tanjakostic.jcleancim.validation.OperationValidator.OperationsWithInvalidArgTypeNull

All Implemented Interfaces:
SimpleRule, Rule

---

public static class **OperationValidator.OperationsWithInvalidArgTypeNull**
extends AbstractRule
implements Rule, SimpleRule

Constructor Summary

public	OperationsWithInvalidArgTypeNull()

Method Summary

java.util.EnumSet	getApplicability()
void	validate(UmlOperation o, ModelIssues issues)

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule

createIssue, createIssue, createIssue, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from interface org.tanjakostic.jcleancim.validation.SimpleRule

validate

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Constructors

OperationsWithInvalidArgTypeNull

```
public OperationsWithInvalidArgTypeNull()
```

Methods

getApplicability

```
public java.util.EnumSet getApplicability()
```

validate

```
public void validate(UmlOperation o,
         ModelIssues issues)
```

org.tanjakostic.jcleancim.validation
Class OperationValidator.OperationsWithInvalidExcTypeNull

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.validation.AbstractRule
      |
      +-
```
org.tanjakostic.jcleancim.validation.OperationValidator.OperationsWithInvalidExcTypeNull

All Implemented Interfaces:
SimpleRule, Rule

---

public static class **OperationValidator.OperationsWithInvalidExcTypeNull**
extends AbstractRule
implements Rule, SimpleRule

Constructor Summary

public	OperationsWithInvalidExcTypeNull()

Method Summary

java.util.EnumSet	getApplicability()
void	validate(UmlOperation o, ModelIssues issues)

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule

createIssue, createIssue, createIssue, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from interface org.tanjakostic.jcleancim.validation.SimpleRule

validate

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Constructors

OperationsWithInvalidExcTypeNull

```
public OperationsWithInvalidExcTypeNull()
```

Methods

getApplicability

```
public java.util.EnumSet getApplicability()
```

validate

```
public void validate(UmlOperation o,
         ModelIssues issues)
```

org.tanjakostic.jcleancim.validation
Class OperationValidator.OperationsWithUnallowedTagNames

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.validation.AbstractRule
      |
      +-
org.tanjakostic.jcleancim.validation.AbstractRule.AbstractRuleWithSubobjectsAndSkips
          |
          +-
org.tanjakostic.jcleancim.validation.AbstractRule.UmlObjectsWithUnallowedTagNames
              |
              +-
```

org.tanjakostic.jcleancim.validation.OperationValidator.OperationsWithUnallowedTagNames

All Implemented Interfaces:
Rule, SimpleRule

---

public static class **OperationValidator.OperationsWithUnallowedTagNames**
extends AbstractRule.UmlObjectsWithUnallowedTagNames

Constructor Summary

public	OperationsWithUnallowedTagNames()

Method Summary

java.util.EnumSet	getApplicability()

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule.UmlObjectsWithUnallowedTagNames

doValidate

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule.AbstractRuleWithSubobjectsAndSkips

doValidate, getSubObjects, skipSubobjectValidation, skipValidation, validate

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule

createIssue, createIssue, createIssue, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from interface `org.tanjakostic.jcleancim.validation.SimpleRule`
`validate`

Methods inherited from interface `org.tanjakostic.jcleancim.validation.Rule`
`getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis`

Constructors

OperationsWithUnallowedTagNames

```
public OperationsWithUnallowedTagNames()
```

Methods

getApplicability

```
public java.util.EnumSet getApplicability()
```

org.tanjakostic.jcleancim.validation
Class OperationValidator.OperationParametersWithUnallowedTagNames

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.validation.AbstractRule
      |
      +-
org.tanjakostic.jcleancim.validation.AbstractRule.AbstractRuleWithSubobjectsAndSkips
          |
          +-
org.tanjakostic.jcleancim.validation.AbstractRule.UmlObjectsWithUnallowedTagNames
              |
              +-
```
org.tanjakostic.jcleancim.validation.OperationValidator.OperationParametersWithUnallowedTagNames

All Implemented Interfaces:
Rule, SimpleRule

---

public static class **OperationValidator.OperationParametersWithUnallowedTagNames**
extends AbstractRule.UmlObjectsWithUnallowedTagNames

Constructor Summary

public	OperationParametersWithUnallowedTagNames()

Method Summary

java.util.EnumSet	getApplicability()
java.util.List	getSubObjects(UmlOperation op)

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule.UmlObjectsWithUnallowedTagNames

doValidate

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule.AbstractRuleWithSubobjectsAndSkips

doValidate, getSubObjects, skipSubobjectValidation, skipValidation, validate

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule

createIssue, createIssue, createIssue, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from interface org.tanjakostic.jcleancim.validation.SimpleRule
validate

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule
getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Constructors

OperationParametersWithUnallowedTagNames

public **OperationParametersWithUnallowedTagNames**()

Methods

getApplicability

public java.util.EnumSet **getApplicability**()

getSubObjects

protected java.util.List **getSubObjects**(UmlOperation op)

Used for validation of "sub-objects" (such as association ends and operation parameters) that do not have their own validators, but get validated through their container.

This default implementation returns empty list; subtypes having contained objects that need to be validated against this rule have to override this method.

org.tanjakostic.jcleancim.validation
Class OperationValidator.OperationsMissingDoc

```
java.lang.Object
  |-org.tanjakostic.jcleancim.validation.AbstractRule
    |-
org.tanjakostic.jcleancim.validation.AbstractRule.AbstractRuleWithSubobjectsAndSkips
      |-org.tanjakostic.jcleancim.validation.AbstractRule.UmlObjectsMissingDoc
        |-
```

org.tanjakostic.jcleancim.validation.OperationValidator.OperationsMissingDoc

All Implemented Interfaces:
Rule, SimpleRule

---

public static class **OperationValidator.OperationsMissingDoc**
extends AbstractRule.UmlObjectsMissingDoc

Constructor Summary

public	OperationsMissingDoc()

Method Summary

java.util.EnumSet	getApplicability()

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule.UmlObjectsMissingDoc

doValidate

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule.AbstractRuleWithSubobjectsAndSkips

doValidate, getSubObjects, skipSubobjectValidation, skipValidation, validate

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule

createIssue, createIssue, createIssue, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from interface org.tanjakostic.jcleancim.validation.SimpleRule

validate

Methods inherited from interface `org.tanjakostic.jcleancim.validation.Rule`
`getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis`

Constructors

OperationsMissingDoc

```
public OperationsMissingDoc()
```

Methods

getApplicability

```
public java.util.EnumSet getApplicability()
```

org.tanjakostic.jcleancim.validation
Class OperationValidator.OperationParametersMissingDoc

```
java.lang.Object
  |-org.tanjakostic.jcleancim.validation.AbstractRule
    |-
org.tanjakostic.jcleancim.validation.AbstractRule.AbstractRuleWithSubobjectsAndSkips
      |-org.tanjakostic.jcleancim.validation.AbstractRule.UmlObjectsMissingDoc
        |-
org.tanjakostic.jcleancim.validation.OperationValidator.OperationParametersMissingDoc
```

All Implemented Interfaces:
Rule, SimpleRule

---

public static class **OperationValidator.OperationParametersMissingDoc**
extends AbstractRule.UmlObjectsMissingDoc

Constructor Summary

public	OperationParametersMissingDoc()

Method Summary

java.util.EnumSet	getApplicability()
java.util.List	getSubObjects(UmlOperation op)

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule.UmlObjectsMissingDoc

doValidate

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule.AbstractRuleWithSubobjectsAndSkips

doValidate, getSubObjects, skipSubobjectValidation, skipValidation, validate

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule

createIssue, createIssue, createIssue, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from interface org.tanjakostic.jcleancim.validation.SimpleRule
validate

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule
getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Constructors

OperationParametersMissingDoc

public **OperationParametersMissingDoc**()

Methods

getApplicability

public java.util.EnumSet **getApplicability**()

getSubObjects

protected java.util.List **getSubObjects**(UmlOperation op)

Used for validation of "sub-objects" (such as association ends and operation parameters) that do not have their own validators, but get validated through their container.

This default implementation returns empty list; subtypes having contained objects that need to be validated against this rule have to override this method.

org.tanjakostic.jcleancim.validation
Class OperationValidator.OperationsWithBadDocStart

```
java.lang.Object
  |-org.tanjakostic.jcleancim.validation.AbstractRule
    |-
org.tanjakostic.jcleancim.validation.AbstractRule.AbstractRuleWithSubobjectsAndSkips
      |-org.tanjakostic.jcleancim.validation.AbstractRule.UmlObjectsWithBadDocStart
        |-
```

org.tanjakostic.jcleancim.validation.OperationValidator.OperationsWithBadDocStart

All Implemented Interfaces:
Rule, SimpleRule

---

public static class **OperationValidator.OperationsWithBadDocStart**
extends AbstractRule.UmlObjectsWithBadDocStart

Constructor Summary

public	OperationsWithBadDocStart()

Method Summary

java.util.EnumSet	getApplicability()

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule.UmlObjectsWithBadDocStart

doValidate

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule.AbstractRuleWithSubobjectsAndSkips

doValidate, getSubObjects, skipSubobjectValidation, skipValidation, validate

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule

createIssue, createIssue, createIssue, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from interface org.tanjakostic.jcleancim.validation.SimpleRule

validate

Methods inherited from interface `org.tanjakostic.jcleancim.validation.Rule`
`getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis`

Constructors

OperationsWithBadDocStart

`public` **`OperationsWithBadDocStart`**`()`

Methods

getApplicability

`public java.util.EnumSet` **`getApplicability`**`()`

org.tanjakostic.jcleancim.validation
Class OperationValidator.OperationParametersWithBadDocStart

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.validation.AbstractRule
      |
      +-
org.tanjakostic.jcleancim.validation.AbstractRule.AbstractRuleWithSubobjectsAndSkips
          |
          +-org.tanjakostic.jcleancim.validation.AbstractRule.UmlObjectsWithBadDocStart
              |
              +-
org.tanjakostic.jcleancim.validation.OperationValidator.OperationParametersWithBadDocSt
art
```

All Implemented Interfaces:
Rule, SimpleRule

---

public static class **OperationValidator.OperationParametersWithBadDocStart**
extends AbstractRule.UmlObjectsWithBadDocStart

Constructor Summary

public	OperationParametersWithBadDocStart()

Method Summary

java.util.EnumSet	getApplicability()
java.util.List	getSubObjects(UmlOperation op)

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule.UmlObjectsWithBadDocStart

doValidate

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule.AbstractRuleWithSubobjectsAndSkips

doValidate, getSubObjects, skipSubobjectValidation, skipValidation, validate

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule

createIssue, createIssue, createIssue, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from interface org.tanjakostic.jcleancim.validation.SimpleRule
validate

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule
getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Constructors

OperationParametersWithBadDocStart

public **OperationParametersWithBadDocStart**()

Methods

getApplicability

public java.util.EnumSet **getApplicability**()

getSubObjects

protected java.util.List **getSubObjects**(UmlOperation op)

Used for validation of "sub-objects" (such as association ends and operation parameters) that do not have their own validators, but get validated through their container.

This default implementation returns empty list; subtypes having contained objects that need to be validated against this rule have to override this method.

org.tanjakostic.jcleancim.validation
Class OperationValidator.OperationsWithBadDocEnd

```
java.lang.Object
  |-org.tanjakostic.jcleancim.validation.AbstractRule
    |-
org.tanjakostic.jcleancim.validation.AbstractRule.AbstractRuleWithSubobjectsAndSkips
      |-org.tanjakostic.jcleancim.validation.AbstractRule.UmlObjectsWithBadDocEnd
        |-
org.tanjakostic.jcleancim.validation.OperationValidator.OperationsWithBadDocEnd
```

All Implemented Interfaces:
Rule, SimpleRule

---

public static class **OperationValidator.OperationsWithBadDocEnd**
extends AbstractRule.UmlObjectsWithBadDocEnd

Constructor Summary

public	OperationsWithBadDocEnd()

Method Summary

java.util.EnumSet	getApplicability()

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule.UmlObjectsWithBadDocEnd

doValidate

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule.AbstractRuleWithSubobjectsAndSkips

doValidate, getSubObjects, skipSubobjectValidation, skipValidation, validate

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule

createIssue, createIssue, createIssue, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from interface org.tanjakostic.jcleancim.validation.SimpleRule

validate

Methods inherited from interface `org.tanjakostic.jcleancim.validation.Rule`
`getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis`

Constructors

OperationsWithBadDocEnd

`public` **`OperationsWithBadDocEnd`**`()`

Methods

getApplicability

`public java.util.EnumSet` **`getApplicability`**`()`

org.tanjakostic.jcleancim.validation
Class OperationValidator.OperationParametersWithBadDocEnd

```
java.lang.Object
  |-org.tanjakostic.jcleancim.validation.AbstractRule
    |-
org.tanjakostic.jcleancim.validation.AbstractRule.AbstractRuleWithSubobjectsAndSkips
      |-org.tanjakostic.jcleancim.validation.AbstractRule.UmlObjectsWithBadDocEnd
        |-
org.tanjakostic.jcleancim.validation.OperationValidator.OperationParametersWithBadDocEnd
```

All Implemented Interfaces:
Rule, SimpleRule

---

public static class **OperationValidator.OperationParametersWithBadDocEnd**
extends AbstractRule.UmlObjectsWithBadDocEnd

Constructor Summary

public	OperationParametersWithBadDocEnd()

Method Summary

java.util.EnumSet	getApplicability()
java.util.List	getSubObjects(UmlOperation op)

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule.UmlObjectsWithBadDocEnd

doValidate

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule.AbstractRuleWithSubobjectsAndSkips

doValidate, getSubObjects, skipSubobjectValidation, skipValidation, validate

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule

createIssue, createIssue, createIssue, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from interface org.tanjakostic.jcleancim.validation.SimpleRule
validate

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule
getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Constructors

OperationParametersWithBadDocEnd

public **OperationParametersWithBadDocEnd**()

Methods

getApplicability

public java.util.EnumSet **getApplicability**()

getSubObjects

protected java.util.List **getSubObjects**(UmlOperation op)

Used for validation of "sub-objects" (such as association ends and operation parameters) that do not have their own validators, but get validated through their container.

This default implementation returns empty list; subtypes having contained objects that need to be validated against this rule have to override this method.

org.tanjakostic.jcleancim.validation
Class OperationValidator.OperationsWithBadCharacterInName

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.validation.AbstractRule
      |
      +-
org.tanjakostic.jcleancim.validation.AbstractRule.AbstractRuleWithSubobjectsAndSkips
          |
          +-
org.tanjakostic.jcleancim.validation.AbstractRule.UmlObjectsWithBadCharacterInName
              |
              +-
```
**org.tanjakostic.jcleancim.validation.OperationValidator.OperationsWithBadCharacterInNam
e**

All Implemented Interfaces:
Rule, SimpleRule

---

public static class **OperationValidator.OperationsWithBadCharacterInName**
extends AbstractRule.UmlObjectsWithBadCharacterInName

Constructor Summary

public	OperationsWithBadCharacterInName()

Method Summary

java.util.EnumSet	getApplicability()
InvalidCharactersFinder	getInvalidCharacterFinder(UmlObject o)

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule.UmlObjectsWithBadCharacterInName

doValidate, getInvalidCharacterFinder

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule.AbstractRuleWithSubobjectsAndSkips

doValidate, getSubObjects, skipSubobjectValidation, skipValidation, validate

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule

createIssue, createIssue, createIssue, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from interface org.tanjakostic.jcleancim.validation.SimpleRule
validate

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule
getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Constructors

OperationsWithBadCharacterInName

public **OperationsWithBadCharacterInName**()

Methods

getApplicability

public java.util.EnumSet **getApplicability**()

getInvalidCharacterFinder

protected InvalidCharactersFinder **getInvalidCharacterFinder**(UmlObject o)

org.tanjakostic.jcleancim.validation
Class OperationValidator.OperationParametersWithBadCharacterInName

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.validation.AbstractRule
      |
      +-
org.tanjakostic.jcleancim.validation.AbstractRule.AbstractRuleWithSubobjectsAndSkips
          |
          +-
org.tanjakostic.jcleancim.validation.AbstractRule.UmlObjectsWithBadCharacterInName
              |
              +-
```
org.tanjakostic.jcleancim.validation.OperationValidator.OperationParametersWithBadCharacterInName

All Implemented Interfaces:
Rule, SimpleRule

---

public static class **OperationValidator.OperationParametersWithBadCharacterInName**
extends AbstractRule.UmlObjectsWithBadCharacterInName

Constructor Summary

public	OperationParametersWithBadCharacterInName()

Method Summary

java.util.EnumSet	getApplicability()
InvalidCharactersFinder	getInvalidCharacterFinder(UmlObject o)
java.util.List	getSubObjects(UmlOperation op)

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule.UmlObjectsWithBadCharacterInName

doValidate, getInvalidCharacterFinder

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule.AbstractRuleWithSubobjectsAndSkips

doValidate, getSubObjects, skipSubobjectValidation, skipValidation, validate

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule

createIssue, createIssue, createIssue, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface `org.tanjakostic.jcleancim.validation.Rule`
`getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis`

Methods inherited from interface `org.tanjakostic.jcleancim.validation.SimpleRule`
`validate`

Methods inherited from interface `org.tanjakostic.jcleancim.validation.Rule`
`getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis`

Constructors

OperationParametersWithBadCharacterInName

`public OperationParametersWithBadCharacterInName()`

Methods

getApplicability

`public java.util.EnumSet getApplicability()`

getSubObjects

`protected java.util.List getSubObjects(UmlOperation op)`

Used for validation of "sub-objects" (such as association ends and operation parameters) that do not have their own validators, but get validated through their container.

This default implementation returns empty list; subtypes having contained objects that need to be validated against this rule have to override this method.

getInvalidCharacterFinder

`protected InvalidCharactersFinder getInvalidCharacterFinder(UmlObject o)`

org.tanjakostic.jcleancim.validation
Class PackageValidator

```
java.lang.Object
  +-org.tanjakostic.jcleancim.validation.AbstractValidator
    +-org.tanjakostic.jcleancim.validation.PackageValidator
```

public class **PackageValidator**
extends AbstractValidator

Validates packages.

Nested Class Summary

class	PackageValidator.Iec61850PackagesThatShouldHaveAliasAsTitle PackageValidator.Iec61850PackagesThatShouldHaveAliasAsTitle
class	PackageValidator.PackagesMissingDoc PackageValidator.PackagesMissingDoc
class	PackageValidator.PackagesTopLevelWithoutVersionClass PackageValidator.PackagesTopLevelWithoutVersionClass
class	PackageValidator.PackagesWithBadCharacterInName PackageValidator.PackagesWithBadCharacterInName
class	PackageValidator.PackagesWithBadDocEnd PackageValidator.PackagesWithBadDocEnd
class	PackageValidator.PackagesWithBadDocStart PackageValidator.PackagesWithBadDocStart
class	PackageValidator.PackagesWithSameName PackageValidator.PackagesWithSameName
class	PackageValidator.PackagesWithSelfDependency PackageValidator.PackagesWithSelfDependency
class	PackageValidator.PackagesWithUnallowedStereotype PackageValidator.PackagesWithUnallowedStereotype
class	PackageValidator.PackagesWithUnallowedTagNames PackageValidator.PackagesWithUnallowedTagNames
class	PackageValidator.PackageUnexpectedConnectors PackageValidator.PackageUnexpectedConnectors
class	PackageValidator.PackageUnexpectedElements PackageValidator.PackageUnexpectedElements

Constructor Summary

public	PackageValidator(Config cfg, java.util.Collection allPackages, ModelIssues issues)

Method Summary

boolean	enabled()
java.util.List	getScopedUmlObjects()

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractValidator

addCrossRule, addSimpleRule, displayAllAvailableRuleNames, displayAvailableRuleNames, enabled, getAllCrossRules, getAllRules, getAllSimpleRules, getCfg, getCheckedCrossRules, getCheckedRules, getCheckedSimpleRules, getCollectedIssues, getScopedUmlObjects, validate

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Constructors

PackageValidator

```
public PackageValidator(Config cfg,
                        java.util.Collection allPackages,
                        ModelIssues issues)
```

Methods

enabled

```
public boolean enabled()
```

Returns whether the validation for this validator has been enabled (by configuration).

getScopedUmlObjects

```
public java.util.List getScopedUmlObjects()
```

Returns the elements retained for validation, for the configured scope.

org.tanjakostic.jcleancim.validation
Class PackageValidator.PackageUnexpectedElements

```
java.lang.Object
  |-org.tanjakostic.jcleancim.validation.AbstractRule
    |-org.tanjakostic.jcleancim.validation.PackageValidator.PackageUnexpectedElements
```

All Implemented Interfaces:
SimpleRule, Rule

public static class **PackageValidator.PackageUnexpectedElements**
extends AbstractRule
implements Rule, SimpleRule

Constructor Summary

public	PackageUnexpectedElements()

Method Summary

java.util.EnumSet	getApplicability()
void	validate(UmlPackage o, ModelIssues issues)

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule

createIssue, createIssue, createIssue, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from interface org.tanjakostic.jcleancim.validation.SimpleRule

validate

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

(continued from last page)

Constructors

PackageUnexpectedElements

```
public PackageUnexpectedElements()
```

Methods

getApplicability

```
public java.util.EnumSet getApplicability()
```

---

validate

```
public void validate(UmlPackage o,
         ModelIssues issues)
```

org.tanjakostic.jcleancim.validation
Class PackageValidator.PackageUnexpectedConnectors

```
java.lang.Object
  |-org.tanjakostic.jcleancim.validation.AbstractRule
    |-
```
org.tanjakostic.jcleancim.validation.PackageValidator.PackageUnexpectedConnectors

All Implemented Interfaces:
SimpleRule, Rule

---

public static class **PackageValidator.PackageUnexpectedConnectors**
extends AbstractRule
implements Rule, SimpleRule

Constructor Summary

public	PackageUnexpectedConnectors()

Method Summary

java.util.EnumSet	getApplicability()
void	validate(UmlPackage o, ModelIssues issues)

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule

createIssue, createIssue, createIssue, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from interface org.tanjakostic.jcleancim.validation.SimpleRule

validate

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

---

(continued from last page)

Constructors

PackageUnexpectedConnectors

```
public PackageUnexpectedConnectors()
```

Methods

getApplicability

```
public java.util.EnumSet getApplicability()
```

---

validate

```
public void validate(UmlPackage o,
         ModelIssues issues)
```

org.tanjakostic.jcleancim.validation
Class PackageValidator.PackagesWithSelfDependency

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.validation.AbstractRule
      |
      +-
```
org.tanjakostic.jcleancim.validation.PackageValidator.PackagesWithSelfDependency

All Implemented Interfaces:
SimpleRule, Rule

---

public static class **PackageValidator.PackagesWithSelfDependency**
extends AbstractRule
implements Rule, SimpleRule

Constructor Summary

public	PackagesWithSelfDependency()

Method Summary

java.util.EnumSet	getApplicability()
void	validate(UmlPackage o, ModelIssues issues)

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule

createIssue, createIssue, createIssue, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from interface org.tanjakostic.jcleancim.validation.SimpleRule

validate

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

---

(continued from last page)

Constructors

PackagesWithSelfDependency

```
public PackagesWithSelfDependency()
```

Methods

getApplicability

```
public java.util.EnumSet getApplicability()
```

---

validate

```
public void validate(UmlPackage o,
         ModelIssues issues)
```

org.tanjakostic.jcleancim.validation
Class PackageValidator.PackagesWithUnallowedStereotype

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.validation.AbstractRule
      |
      +-org.tanjakostic.jcleancim.validation.AbstractRule.AbstractRuleWithSubobjectsAndSkips
          |
          +-org.tanjakostic.jcleancim.validation.AbstractRule.UmlObjectsWithUnallowedStereotype
              |
              +-org.tanjakostic.jcleancim.validation.PackageValidator.PackagesWithUnallowedStereotype
```

All Implemented Interfaces:
Rule, SimpleRule

---

public static class **PackageValidator.PackagesWithUnallowedStereotype**
extends AbstractRule.UmlObjectsWithUnallowedStereotype

Constructor Summary

public	PackagesWithUnallowedStereotype()

Method Summary

java.util.EnumSet	getApplicability()

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule.UmlObjectsWithUnallowedStereotype

doValidate

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule.AbstractRuleWithSubobjectsAndSkips

doValidate, getSubObjects, skipSubobjectValidation, skipValidation, validate

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule

createIssue, createIssue, createIssue, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from interface `org.tanjakostic.jcleancim.validation.SimpleRule`
`validate`

Methods inherited from interface `org.tanjakostic.jcleancim.validation.Rule`
`getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis`

Constructors

PackagesWithUnallowedStereotype

```
public PackagesWithUnallowedStereotype()
```

Methods

getApplicability

```
public java.util.EnumSet getApplicability()
```

org.tanjakostic.jcleancim.validation
Class PackageValidator.PackagesTopLevelWithoutVersionClass

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.validation.AbstractRule
      |
      +-
```
org.tanjakostic.jcleancim.validation.PackageValidator.PackagesTopLevelWithoutVersionClass

All Implemented Interfaces:
SimpleRule, Rule

---

public static class **PackageValidator.PackagesTopLevelWithoutVersionClass**
extends AbstractRule
implements Rule, SimpleRule

Constructor Summary

public	PackagesTopLevelWithoutVersionClass()

Method Summary

java.util.EnumSet	getApplicability()
void	validate(UmlPackage o, ModelIssues issues)

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule

createIssue, createIssue, createIssue, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from interface org.tanjakostic.jcleancim.validation.SimpleRule

validate

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Constructors

PackagesTopLevelWithoutVersionClass

```
public PackagesTopLevelWithoutVersionClass()
```

Methods

getApplicability

```
public java.util.EnumSet getApplicability()
```

validate

```
public void validate(UmlPackage o,
         ModelIssues issues)
```

org.tanjakostic.jcleancim.validation
Class PackageValidator.Iec61850PackagesThatShouldHaveAliasAsTitle

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.validation.AbstractRule
      |
      +-
```
org.tanjakostic.jcleancim.validation.PackageValidator.Iec61850PackagesThatShouldHaveAliasAsTitle

All Implemented Interfaces:
SimpleRule, Rule

---

public static class **PackageValidator.Iec61850PackagesThatShouldHaveAliasAsTitle**
extends AbstractRule
implements Rule, SimpleRule

Constructor Summary

public	Iec61850PackagesThatShouldHaveAliasAsTitle()

Method Summary

java.util.EnumSet	getApplicability()
void	validate(UmlPackage o, ModelIssues issues)

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule

createIssue, createIssue, createIssue, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from interface org.tanjakostic.jcleancim.validation.SimpleRule

validate

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Constructors

Iec61850PackagesThatShouldHaveAliasAsTitle

```
public Iec61850PackagesThatShouldHaveAliasAsTitle()
```

Methods

getApplicability

```
public java.util.EnumSet getApplicability()
```

validate

```
public void validate(UmlPackage o,
         ModelIssues issues)
```

org.tanjakostic.jcleancim.validation
Class PackageValidator.PackagesWithUnallowedTagNames

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.validation.AbstractRule
      |
      +-
org.tanjakostic.jcleancim.validation.AbstractRule.AbstractRuleWithSubobjectsAndSkips
          |
          +-
org.tanjakostic.jcleancim.validation.AbstractRule.UmlObjectsWithUnallowedTagNames
              |
              +-
```

org.tanjakostic.jcleancim.validation.PackageValidator.PackagesWithUnallowedTagNames

All Implemented Interfaces:
Rule, SimpleRule

---

public static class **PackageValidator.PackagesWithUnallowedTagNames**
extends AbstractRule.UmlObjectsWithUnallowedTagNames

Constructor Summary

public	PackagesWithUnallowedTagNames()

Method Summary

java.util.EnumSet	getApplicability()

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule.UmlObjectsWithUnallowedTagNames

doValidate

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule.AbstractRuleWithSubobjectsAndSkips

doValidate, getSubObjects, skipSubobjectValidation, skipValidation, validate

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule

createIssue, createIssue, createIssue, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from interface `org.tanjakostic.jcleancim.validation.SimpleRule`
`validate`

Methods inherited from interface `org.tanjakostic.jcleancim.validation.Rule`
`getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis`

Constructors

PackagesWithUnallowedTagNames

`public` **`PackagesWithUnallowedTagNames`**`()`

Methods

getApplicability

`public java.util.EnumSet` **`getApplicability`**`()`

org.tanjakostic.jcleancim.validation
Class PackageValidator.PackagesMissingDoc

```
java.lang.Object
  |-org.tanjakostic.jcleancim.validation.AbstractRule
    |-
org.tanjakostic.jcleancim.validation.AbstractRule.AbstractRuleWithSubobjectsAndSkips
      |-org.tanjakostic.jcleancim.validation.AbstractRule.UmlObjectsMissingDoc
        |-org.tanjakostic.jcleancim.validation.PackageValidator.PackagesMissingDoc
```

All Implemented Interfaces:
Rule, SimpleRule

---

public static class **PackageValidator.PackagesMissingDoc**
extends AbstractRule.UmlObjectsMissingDoc

Constructor Summary

public	PackagesMissingDoc()

Method Summary

java.util.EnumSet	getApplicability()

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule.UmlObjectsMissingDoc

doValidate

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule.AbstractRuleWithSubobjectsAndSkips

doValidate, getSubObjects, skipSubobjectValidation, skipValidation, validate

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule

createIssue, createIssue, createIssue, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from interface org.tanjakostic.jcleancim.validation.SimpleRule

validate

Methods inherited from interface `org.tanjakostic.jcleancim.validation.Rule`
`getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis`

Constructors

PackagesMissingDoc

`public` **`PackagesMissingDoc`**`()`

Methods

getApplicability

`public java.util.EnumSet` **`getApplicability`**`()`

org.tanjakostic.jcleancim.validation
Class PackageValidator.PackagesWithBadDocStart

```
java.lang.Object
  |-org.tanjakostic.jcleancim.validation.AbstractRule
    |-
org.tanjakostic.jcleancim.validation.AbstractRule.AbstractRuleWithSubobjectsAndSkips
      |-org.tanjakostic.jcleancim.validation.AbstractRule.UmlObjectsWithBadDocStart
        |-
org.tanjakostic.jcleancim.validation.PackageValidator.PackagesWithBadDocStart
```

All Implemented Interfaces:
Rule, SimpleRule

---

public static class **PackageValidator.PackagesWithBadDocStart**
extends AbstractRule.UmlObjectsWithBadDocStart

Constructor Summary

public	PackagesWithBadDocStart()

Method Summary

java.util.EnumSet	getApplicability()

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule.UmlObjectsWithBadDocStart

doValidate

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule.AbstractRuleWithSubobjectsAndSkips

doValidate, getSubObjects, skipSubobjectValidation, skipValidation, validate

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule

createIssue, createIssue, createIssue, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from interface org.tanjakostic.jcleancim.validation.SimpleRule

validate

Methods inherited from interface `org.tanjakostic.jcleancim.validation.Rule`
`getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis`

Constructors

PackagesWithBadDocStart

`public` **`PackagesWithBadDocStart`**`()`

Methods

getApplicability

`public java.util.EnumSet` **`getApplicability`**`()`

org.tanjakostic.jcleancim.validation
Class PackageValidator.PackagesWithBadDocEnd

```
java.lang.Object
  |-org.tanjakostic.jcleancim.validation.AbstractRule
    |-
org.tanjakostic.jcleancim.validation.AbstractRule.AbstractRuleWithSubobjectsAndSkips
      |-org.tanjakostic.jcleancim.validation.AbstractRule.UmlObjectsWithBadDocEnd
        |-
org.tanjakostic.jcleancim.validation.PackageValidator.PackagesWithBadDocEnd
```

All Implemented Interfaces:
Rule, SimpleRule

---

public static class **PackageValidator.PackagesWithBadDocEnd**
extends AbstractRule.UmlObjectsWithBadDocEnd

Constructor Summary

public	PackagesWithBadDocEnd()

Method Summary

java.util.EnumSet	getApplicability()

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule.UmlObjectsWithBadDocEnd

doValidate

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule.AbstractRuleWithSubobjectsAndSkips

doValidate, getSubObjects, skipSubobjectValidation, skipValidation, validate

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule

createIssue, createIssue, createIssue, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from interface org.tanjakostic.jcleancim.validation.SimpleRule

validate

Methods inherited from interface `org.tanjakostic.jcleancim.validation.Rule`
`getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis`

Constructors

PackagesWithBadDocEnd

`public` **`PackagesWithBadDocEnd`**`()`

Methods

getApplicability

`public java.util.EnumSet` **`getApplicability`**`()`

org.tanjakostic.jcleancim.validation
Class PackageValidator.PackagesWithBadCharacterInName

```
java.lang.Object
  |-org.tanjakostic.jcleancim.validation.AbstractRule
    |-
org.tanjakostic.jcleancim.validation.AbstractRule.AbstractRuleWithSubobjectsAndSkips
      |-
org.tanjakostic.jcleancim.validation.AbstractRule.UmlObjectsWithBadCharacterInName
        |-
org.tanjakostic.jcleancim.validation.PackageValidator.PackagesWithBadCharacterInName
```

All Implemented Interfaces:
Rule, SimpleRule

---

public static class **PackageValidator.PackagesWithBadCharacterInName**
extends AbstractRule.UmlObjectsWithBadCharacterInName

Constructor Summary

public	PackagesWithBadCharacterInName()

Method Summary

java.util.EnumSet	getApplicability()
InvalidCharactersFinder	getInvalidCharacterFinder(UmlObject o)

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule.UmlObjectsWithBadCharacterInName

doValidate, getInvalidCharacterFinder

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule.AbstractRuleWithSubobjectsAndSkips

doValidate, getSubObjects, skipSubobjectValidation, skipValidation, validate

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule

createIssue, createIssue, createIssue, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from interface org.tanjakostic.jcleancim.validation.SimpleRule
validate

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule
getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Constructors

PackagesWithBadCharacterInName

public **PackagesWithBadCharacterInName**()

Methods

getApplicability

public java.util.EnumSet **getApplicability**()

getInvalidCharacterFinder

protected InvalidCharactersFinder **getInvalidCharacterFinder**(UmlObject o)

org.tanjakostic.jcleancim.validation
Class PackageValidator.PackagesWithSameName

```
java.lang.Object
  |-org.tanjakostic.jcleancim.validation.AbstractRule
    |-org.tanjakostic.jcleancim.validation.PackageValidator.PackagesWithSameName
```

All Implemented Interfaces:
CrossRule, Rule

public static class **PackageValidator.PackagesWithSameName**
extends AbstractRule
implements Rule, CrossRule

Constructor Summary

public	PackagesWithSameName(java.util.Collection allPackages)

Method Summary

java.util.EnumSet	getApplicability()
java.util.Collection	getObjsToTestAgainst()
void	validate(java.util.List packages, ModelIssues issues)

Methods inherited from class org.tanjakostic.jcleancim.validation.AbstractRule

createIssue, createIssue, createIssue, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Methods inherited from interface org.tanjakostic.jcleancim.validation.CrossRule

getObjsToTestAgainst, validate

Methods inherited from interface org.tanjakostic.jcleancim.validation.Rule

getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis

Constructors

PackagesWithSameName

```
public PackagesWithSameName(java.util.Collection allPackages)
```

Methods

getApplicability

```
public java.util.EnumSet getApplicability()
```

getObjsToTestAgainst

```
public java.util.Collection getObjsToTestAgainst()
```

validate

```
public void validate(java.util.List packages,
         ModelIssues issues)
```

Ignores packages with name `UML.DetailedDiagrams` - this is reserved name, repeated on purpose in the model.

org.tanjakostic.jcleancim.validation
Interface SimpleRule

All Known Implementing Classes:

AbstractRuleWithSubobjectsAndSkips, CimAssociationEndsNameShouldBeSingular, CimAssociationEndsNameShouldBePlural, CimAssociationEndsNameStartingWithLowerCase, Iec61850AssociationsWithDifferentEndVisibility, Iec61850AssociationsThatShouldBePrivate, AssociationsWithWrongSource, AssociationsWithNoMultiplicity, AssociationsMissingInformativeStereotype, AssociationsWithName, AssociationsWithSameDocOnBothEnds, AssociationsWithDoc, AssociationsWithRoleBadDirection, AssociationsWithExplicitDirection, AttributesWithTypeFromUnallowedOwner, Iec61850DOAttributesNameStartingWithLowerCase, Iec61850AbbreviationLiteralsNameStartingWithLowerCase, CimAttributesNameShouldNotStartWithClassName, CimAttributesNameShouldBeSingular, CimAttributesNameStartingWithUpperCase, Iec61850DOAttributesWithNameMissingAbbreviation, AttributesWithInexistingEnumLiteralAsInitValue, Iec61850FCDAAttributesWithMissingConstraint, Iec61850DOAttributesWithTooLongName, CimAttributesWithFlagInName, Iec61850AttributesWithInexistingSibling, AttributesWhoseTypeIsInformative, CimAttributesThatShouldBeReplacedWithAssociation, AttributesThatAreEnumsInNonEnumeratedClass, AttributesThatAreConstNonStatic, CimAttributesThatAreNotStaticNonConstWithInitVal, AttributesThatAreStaticButNotConst, CimAttributesThatShouldBePublic, AttributesWithTypeIdMismatch, AttributesWithInvalidTypeString, AttributesWithInvalidTypeNull, CimAttributesThatShouldBeOptional, AttributesWithInvalidMultiplicity, EnumLiteralsWithoutEnumStereotype, EnumLiteralsWithSuperfluousType, EnumClassesWithDuplicateCodes, EnumClassesWithSomeCodesMissing, Iec61850LNClassesMalformedName, Iec61850LNClassesInWrongGroup, CimClassesNameShouldBeSingular, CimClassesNameStartingWithLowerCase, CimDatatypeClassesWithInvalidAttributes, Iec61850ClassesWithMissingCondIDTextInConstraints, Iec61850LNClassesWithSuperfluousConstraints, Iec61850ClassesWithInvalidConstraints, CimClassesNeverUsedInRelationships, Iec61850ClassesThatShouldHaveTaggedValuesForDocgen, Iec61850ClassesThatShouldHaveAliasAsTitle, ClassesThatShouldNotHaveNestingThroughAttribute, CimClassesThatShouldNotHaveExplicitDependencies, CimClassesThatShouldNotHaveOperations, CimClassesThatShouldNotBeAbstract, CimClassesUsedForAttributesButHaveSuperclasses, CimClassesUsedForAttributesButHaveSubclasses, CimClassesUsedForAttributesButHaveAssociations, CimClassesWithOldDatatypeStereotype, ClassesThatShouldNotBeAssociationClass, ClassesWithSuperclassesFromUnallowedOwner, ClassesWithMultipleSuperclasses, ClassesWithPersistentPropSet, ClassesWithRootPropSet, ClassesWithLeafPropSet, ClassesWithSelfDependency, ClassesWithDuplicateOwnOrInheritedAssociationEndNames, ClassesWithDuplicateInheritedAttributeNames, CimPrimitiveClassesWithIllegalOwner, CimPrimitiveClassesWithAttributes, ClassesWithSelfInheritance, EnumClassesWithBadName, ClassesWithQuestionableAttributeCount, ClassesWithUnexpectedConnectors, CimClassesWithUnexpectedElements, DependenciesWithUnallowedDirection, DiagramsWithBadOrientation, OperationsWithInvalidExcTypeNull, OperationsWithInvalidArgTypeNull, OperationsWithInvalidReturnTypeNull, OperationsWithUpperCaseName, Iec61850PackagesThatShouldHaveAliasAsTitle, PackagesTopLevelWithoutVersionClass, PackagesWithSelfDependency, PackageUnexpectedConnectors, PackageUnexpectedElements

---

public interface **SimpleRule**
extends Rule

Rule that applies to a single `UmlObject`.
Parameters:
`T`

Method Summary

`abstract void`	`validate(UmlObject obj, ModelIssues toCollect)` Applies the validation criteria to `obj`, and creates problems for invalid ones and adds them to `toCollect`.

Methods inherited from interface `org.tanjakostic.jcleancim.validation.Rule`

`getApplicability, getCategory, getHowToFix, getHypothesis, getLogLevel, getSeverity, logDiagnosis`

Methods

validate

```
public abstract void validate(UmlObject obj,
         ModelIssues toCollect)
```

Applies the validation criteria to `obj`, and creates problems for invalid ones and adds them to `toCollect`.

Package
org.tanjakostic.jcleancim.xml

org.tanjakostic.jcleancim.xml
Class AbstractConfiguredDOMBuilder

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.xml.AbstractConfiguredDOMBuilder
```

All Implemented Interfaces:
ConfiguredDOMBuilder

Direct Known Subclasses:
AbstractXsdValidatingDOMBuilder, WellformedDOMBuilder

---

public abstract class **AbstractConfiguredDOMBuilder**
extends java.lang.Object
implements ConfiguredDOMBuilder

Implementation for commons of all the DOM builders.

Nested Class Summary

class	AbstractConfiguredDOMBuilder.SaxErrorCollector AbstractConfiguredDOMBuilder.SaxErrorCollector

Constructor Summary

protected	AbstractConfiguredDOMBuilder(boolean builderFactorySetValidate)

Method Summary

javax.xml.parsers.DocumentBuilder	getDOMBuilder()
javax.xml.parsers.DocumentBuilderFactory	getDOMBuilderFactory()
org.xml.sax.ErrorHandler	getErrorHandler()
SaxErrorData	getParsingErrors()
javax.xml.validation.SchemaFactory	getSchemaFactory() This default implementation returns null; override to return configured schema factory.
org.w3c.dom.Document	readAndValidate(java.io.File xmlFile)
org.w3c.dom.Document	readAndValidate(org.xml.sax.InputSource source)
org.w3c.dom.Document	readAndValidate(XmlString xmlText)

Methods inherited from class `java.lang.Object`

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.xml.ConfiguredDOMBuilder
getParsingErrors, readAndValidate, readAndValidate, readAndValidate

Constructors

AbstractConfiguredDOMBuilder

```
protected AbstractConfiguredDOMBuilder(boolean builderFactorySetValidate)
```

Methods

getDOMBuilderFactory

```
protected final javax.xml.parsers.DocumentBuilderFactory getDOMBuilderFactory()
```

getDOMBuilder

```
protected final javax.xml.parsers.DocumentBuilder getDOMBuilder()
  throws javax.xml.parsers.ParserConfigurationException
```

getParsingErrors

```
public final SaxErrorData getParsingErrors()
```

readAndValidate

```
public final org.w3c.dom.Document readAndValidate(java.io.File xmlFile)
```

readAndValidate

```
public final org.w3c.dom.Document readAndValidate(XmlString xmlText)
```

readAndValidate

```
public final org.w3c.dom.Document readAndValidate(org.xml.sax.InputSource source)
```

(continued from last page)

getErrorHandler

```
protected final org.xml.sax.ErrorHandler getErrorHandler()
```

getSchemaFactory

```
protected javax.xml.validation.SchemaFactory getSchemaFactory()
```

This default implementation returns null; override to return configured schema factory.

org.tanjakostic.jcleancim.xml
Class AbstractConfiguredDOMBuilder.SaxErrorCollector

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.xml.AbstractConfiguredDOMBuilder.SaxErrorCollector
```

All Implemented Interfaces:
org.xml.sax.ErrorHandler

---

protected static class **AbstractConfiguredDOMBuilder.SaxErrorCollector**
extends java.lang.Object
implements org.xml.sax.ErrorHandler

Simple implementation of SAX error handler when validating against schema.

Method Summary

void	error(org.xml.sax.SAXParseException e)
void	fatalError(org.xml.sax.SAXParseException e)
SaxErrorData	getData()
boolean	isCollectAll()
void	warning(org.xml.sax.SAXParseException e)

Methods inherited from class `java.lang.Object`
clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface `org.xml.sax.ErrorHandler`
error, fatalError, warning

Methods

getData

public SaxErrorData **getData**()

---

isCollectAll

public boolean **isCollectAll**()

warning

```
public void warning(org.xml.sax.SAXParseException e)
```

error

```
public void error(org.xml.sax.SAXParseException e)
  throws org.xml.sax.SAXException
```

fatalError

```
public void fatalError(org.xml.sax.SAXParseException e)
  throws org.xml.sax.SAXException
```

org.tanjakostic.jcleancim.xml
Class AbstractXsdValidatingDOMBuilder

```
java.lang.Object
  |-org.tanjakostic.jcleancim.xml.AbstractConfiguredDOMBuilder
    |-org.tanjakostic.jcleancim.xml.AbstractXsdValidatingDOMBuilder
```

All Implemented Interfaces:
ConfiguredDOMBuilder

Direct Known Subclasses:
ExternalXsdValidatingDOMBuilder, InternalXsdValidatingDOMBuilder

---

public abstract class **AbstractXsdValidatingDOMBuilder**
extends AbstractConfiguredDOMBuilder

Common implementation for readers that validate against the XML schema, internal or external.

Field Summary

public static final	W3C_XML_SCHEMA Value: **http://www.w3.org/2001/XMLSchema**

Constructor Summary

protected	AbstractXsdValidatingDOMBuilder(java.io.InputStream externalSchema, boolean saxReaderSetValidate)

Method Summary

abstract void	configureBuilderFactoryWithSchema(javax.xml.parsers.DocumentBuilderFactory factory)
java.io.InputStream	getExternalSchema()
void	validate(org.w3c.dom.Document document) Validates existing DOM document against XML schema.

Methods inherited from class org.tanjakostic.jcleancim.xml.AbstractConfiguredDOMBuilder

getDOMBuilder, getDOMBuilderFactory, getErrorHandler, getParsingErrors, getSchemaFactory, readAndValidate, readAndValidate, readAndValidate

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.xml.ConfiguredDOMBuilder

getParsingErrors, readAndValidate, readAndValidate, readAndValidate

Fields

W3C_XML_SCHEMA

```
public static final java.lang.String W3C_XML_SCHEMA
```

Constant value: **http://www.w3.org/2001/XMLSchema**

Constructors

AbstractXsdValidatingDOMBuilder

```
protected AbstractXsdValidatingDOMBuilder(java.io.InputStream externalSchema,
                                          boolean saxReaderSetValidate)
```

Methods

configureBuilderFactoryWithSchema

```
protected abstract void
configureBuilderFactoryWithSchema(javax.xml.parsers.DocumentBuilderFactory factory)
  throws XmlParsingException
```

getExternalSchema

```
protected final java.io.InputStream getExternalSchema()
```

validate

```
public void validate(org.w3c.dom.Document document)
  throws XmlParsingException
```

Validates existing DOM `document` against XML schema.

Throws:
`XmlParsingException`

org.tanjakostic.jcleancim.xml
Interface ConfiguredDOMBuilder

All Known Implementing Classes:
AbstractConfiguredDOMBuilder

public interface **ConfiguredDOMBuilder**
extends

Configured DOM builder, containing potentially parser errors.

Implementations for configuring the readers using DOM are according to How to Validate XML using Java.

Method Summary

abstract SaxErrorData	getParsingErrors() Returns errors collected during parsing; may be empty but never null.
abstract org.w3c.dom.Document	readAndValidate(java.io.File xmlFile) Reads and validates xmlFile and returns it as DOM document.
abstract org.w3c.dom.Document	readAndValidate(org.xml.sax.InputSource source) Reads and validates source and returns it as DOM document.
abstract org.w3c.dom.Document	readAndValidate(XmlString xmlText) Reads and validates xmlText and returns it as DOM document.

Methods

getParsingErrors

```
public abstract SaxErrorData getParsingErrors()
```

Returns errors collected during parsing; may be empty but never null.

readAndValidate

```
public abstract org.w3c.dom.Document readAndValidate(java.io.File xmlFile)
  throws XmlParsingException
```

Reads and validates xmlFile and returns it as DOM document.

Throws:
XmlParsingException

readAndValidate

```
public abstract org.w3c.dom.Document readAndValidate(XmlString xmlText)
  throws XmlParsingException
```

Reads and validates xmlText and returns it as DOM document.

Throws:

(continued from last page)

`XmlParsingException`

readAndValidate

```
public abstract org.w3c.dom.Document readAndValidate(org.xml.sax.InputSource source)
  throws XmlParsingException
```

Reads and validates `source` and returns it as DOM document.

Throws:

`XmlParsingException`

org.tanjakostic.jcleancim.xml
Class ExternalXsdInternalDtdValidatingDOMBuilder

```
java.lang.Object
  |-org.tanjakostic.jcleancim.xml.AbstractConfiguredDOMBuilder
    |-org.tanjakostic.jcleancim.xml.AbstractXsdValidatingDOMBuilder
      |-org.tanjakostic.jcleancim.xml.ExternalXsdValidatingDOMBuilder
        |-org.tanjakostic.jcleancim.xml.ExternalXsdInternalDtdValidatingDOMBuilder
```

All Implemented Interfaces:
ConfiguredDOMBuilder

---

public class **ExternalXsdInternalDtdValidatingDOMBuilder**
extends ExternalXsdValidatingDOMBuilder

DOM builder configured to validate against both the external schema (specified programmatically, by the code) and the internal DTD (specified in the instance file through DOCTYPE).

Fields inherited from class `org.tanjakostic.jcleancim.xml.AbstractXsdValidatingDOMBuilder`
`W3C_XML_SCHEMA`

Constructor Summary

public	`ExternalXsdInternalDtdValidatingDOMBuilder(java.io.InputStream externalSchema)`

Methods inherited from class `org.tanjakostic.jcleancim.xml.ExternalXsdValidatingDOMBuilder`
`configureBuilderFactoryWithSchema`

Methods inherited from class `org.tanjakostic.jcleancim.xml.AbstractXsdValidatingDOMBuilder`
`configureBuilderFactoryWithSchema`, `getExternalSchema`, `validate`

Methods inherited from class `org.tanjakostic.jcleancim.xml.AbstractConfiguredDOMBuilder`
`getDOMBuilder`, `getDOMBuilderFactory`, `getErrorHandler`, `getParsingErrors`, `getSchemaFactory`, `readAndValidate`, `readAndValidate`, `readAndValidate`

Methods inherited from class `java.lang.Object`
`clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait`

Methods inherited from interface `org.tanjakostic.jcleancim.xml.ConfiguredDOMBuilder`
`getParsingErrors`, `readAndValidate`, `readAndValidate`, `readAndValidate`

Constructors

(continued from last page)

ExternalXsdInternalDtdValidatingDOMBuilder

```
public ExternalXsdInternalDtdValidatingDOMBuilder(java.io.InputStream externalSchema)
```

org.tanjakostic.jcleancim.xml
Class ExternalXsdValidatingDOMBuilder

```
java.lang.Object
  +-org.tanjakostic.jcleancim.xml.AbstractConfiguredDOMBuilder
    +-org.tanjakostic.jcleancim.xml.AbstractXsdValidatingDOMBuilder
      +-org.tanjakostic.jcleancim.xml.ExternalXsdValidatingDOMBuilder
```

All Implemented Interfaces:
ConfiguredDOMBuilder

Direct Known Subclasses:
ExternalXsdInternalDtdValidatingDOMBuilder

---

public class **ExternalXsdValidatingDOMBuilder**
extends AbstractXsdValidatingDOMBuilder

DOM builder configured to validate against the external schema (specified programmatically, by the constructor argument).

Fields inherited from class `org.tanjakostic.jcleancim.xml.AbstractXsdValidatingDOMBuilder`
`W3C_XML_SCHEMA`

Constructor Summary

public	`ExternalXsdValidatingDOMBuilder(java.io.InputStream externalSchema)` Constructor.
protected	`ExternalXsdValidatingDOMBuilder(java.io.InputStream externalSchema, boolean respectDtd)`

Method Summary

void	`configureBuilderFactoryWithSchema(javax.xml.parsers.DocumentBuilderFactory builderFactory)`

Methods inherited from class `org.tanjakostic.jcleancim.xml.AbstractXsdValidatingDOMBuilder`
`configureBuilderFactoryWithSchema`, `getExternalSchema`, `validate`

Methods inherited from class `org.tanjakostic.jcleancim.xml.AbstractConfiguredDOMBuilder`
`getDOMBuilder`, `getDOMBuilderFactory`, `getErrorHandler`, `getParsingErrors`, `getSchemaFactory`, `readAndValidate`, `readAndValidate`, `readAndValidate`

Methods inherited from class `java.lang.Object`
`clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait`

Methods inherited from interface `org.tanjakostic.jcleancim.xml.ConfiguredDOMBuilder`

getParsingErrors, readAndValidate, readAndValidate, readAndValidate

Constructors

ExternalXsdValidatingDOMBuilder

```
public ExternalXsdValidatingDOMBuilder(java.io.InputStream externalSchema)
```

Constructor.

Parameters:

`externalSchema` - non-null external schema as input stream.

ExternalXsdValidatingDOMBuilder

```
protected ExternalXsdValidatingDOMBuilder(java.io.InputStream externalSchema,
                                          boolean respectDtd)
```

Methods

configureBuilderFactoryWithSchema

```
protected void
configureBuilderFactoryWithSchema(javax.xml.parsers.DocumentBuilderFactory
builderFactory)
  throws XmlParsingException
```

org.tanjakostic.jcleancim.xml
Class InternalDtdValidatingDOMBuilder

```
java.lang.Object
  |-org.tanjakostic.jcleancim.xml.AbstractConfiguredDOMBuilder
    |-org.tanjakostic.jcleancim.xml.WellformedDOMBuilder
      |-org.tanjakostic.jcleancim.xml.InternalDtdValidatingDOMBuilder
```

All Implemented Interfaces:
ConfiguredDOMBuilder

---

public class **InternalDtdValidatingDOMBuilder**
extends WellformedDOMBuilder

SAX reader configured to validate against the internal DTD (specified in the instance file through DOCTYPE).

Constructor Summary

public	InternalDtdValidatingDOMBuilder()

Methods inherited from class org.tanjakostic.jcleancim.xml.WellformedDOMBuilder
emptyDocument

Methods inherited from class org.tanjakostic.jcleancim.xml.AbstractConfiguredDOMBuilder
getDOMBuilder, getDOMBuilderFactory, getErrorHandler, getParsingErrors, getSchemaFactory, readAndValidate, readAndValidate, readAndValidate

Methods inherited from class java.lang.Object
clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.xml.ConfiguredDOMBuilder
getParsingErrors, readAndValidate, readAndValidate, readAndValidate

---

Constructors

InternalDtdValidatingDOMBuilder

public **InternalDtdValidatingDOMBuilder**()

org.tanjakostic.jcleancim.xml
Class InternalXsdValidatingDOMBuilder

```
java.lang.Object
  +-org.tanjakostic.jcleancim.xml.AbstractConfiguredDOMBuilder
    +-org.tanjakostic.jcleancim.xml.AbstractXsdValidatingDOMBuilder
      +-org.tanjakostic.jcleancim.xml.InternalXsdValidatingDOMBuilder
```

All Implemented Interfaces:
ConfiguredDOMBuilder

---

public class **InternalXsdValidatingDOMBuilder**
extends AbstractXsdValidatingDOMBuilder

SAX reader configured to validate against the internal schema (specified in the instance file through schema location).

Fields inherited from class `org.tanjakostic.jcleancim.xml.AbstractXsdValidatingDOMBuilder`
`W3C_XML_SCHEMA`

Constructor Summary

public	`InternalXsdValidatingDOMBuilder()`

Method Summary

void	`configureBuilderFactoryWithSchema(javax.xml.parsers.DocumentBuilderFactory builderFactory)`

Methods inherited from class `org.tanjakostic.jcleancim.xml.AbstractXsdValidatingDOMBuilder`
`configureBuilderFactoryWithSchema`, `getExternalSchema`, `validate`

Methods inherited from class `org.tanjakostic.jcleancim.xml.AbstractConfiguredDOMBuilder`
`getDOMBuilder`, `getDOMBuilderFactory`, `getErrorHandler`, `getParsingErrors`, `getSchemaFactory`, `readAndValidate`, `readAndValidate`, `readAndValidate`

Methods inherited from class `java.lang.Object`
`clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait`

Methods inherited from interface `org.tanjakostic.jcleancim.xml.ConfiguredDOMBuilder`
`getParsingErrors`, `readAndValidate`, `readAndValidate`, `readAndValidate`

Constructors

(continued from last page)

InternalXsdValidatingDOMBuilder

```
public InternalXsdValidatingDOMBuilder()
```

Methods

configureBuilderFactoryWithSchema

```
protected void
configureBuilderFactoryWithSchema(javax.xml.parsers.DocumentBuilderFactory
builderFactory)
```

org.tanjakostic.jcleancim.xml
Class JaxpHelper

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.xml.JaxpHelper
```

public class **JaxpHelper**
extends java.lang.Object

Field Summary

public static final	INDENT Value: **2**

Method Summary

static org.w3c.dom.Element	addCDATA(org.w3c.dom.Element el, java.lang.String cdata, org.w3c.dom.Document document) Adds CDATA section to el and returns modified el.
static void	addNamespace(org.w3c.dom.Document dom, XmlNs ns) Adds the namespace binding definition to DOM dom (to define multiple namespace bindings on the root element).
static org.w3c.dom.Element	addQSubElement(org.w3c.dom.Element el, java.lang.String qname, org.w3c.dom.Document document) Adds sub-element to el and returns modified el.
static XmlString	asPrettyXml(org.w3c.dom.Node documentOrElement, java.io.File xmlFile) Returns documentOrElement as pretty-print XML string and saves it to xmlFile if not null.
static XmlString	asPrettyXml(XmlString xmlText, java.io.File xmlFile) Returns xmlText as pretty-print string and saves to xmlFile if not null.
static XmlString	asXml(org.w3c.dom.Node documentOrElement, java.io.File xmlFile) Returns documentOrElement as XML string and saves it to xmlFile if not null.
static javax.xml.xpath.XPathExpression	compileXpath(java.lang.String xpathExpression, XmlNs[] namespaces) Returns compiled xpath expression that recognises all namespaces.
static org.w3c.dom.Document	createDocumentWithRoot(java.lang.String comment, java.lang.String rootName, XmlNs ns) Returns a document with rootName element.
static org.w3c.dom.Element	createQRoot(java.lang.String qname, org.w3c.dom.Document document) Creates root element, adds it to the document and returns that new root element.
static org.w3c.dom.Element	createQSubElement(org.w3c.dom.Element el, java.lang.String qname, org.w3c.dom.Document document) Creates sub-element, adds it to el and returns that new sub-element.

static javax.xml.xpath.XPath	`createXpath(XmlNs[] namespaces)` Returns the xpath instance that recognises all `namespaces`.
static java.util.List	`getAttributes(org.w3c.dom.Element el)` Returns all attributes of `el`, potentially empty list.
static org.w3c.dom.Element	`getFirstNamedSubElement(org.w3c.dom.Element el, java.lang.String name)` Returns first element under `el` with the `name` if found, null otherwise.
static org.w3c.dom.Attr	`getNamedAttribute(org.w3c.dom.Element el, java.lang.String name)` Returns attribute of `el` with the `name` if found, null otherwise.
static java.util.List	`getNamedSubElements(org.w3c.dom.Element el, java.lang.String name)` Returns elements under `el` with the `name` if found, empty list otherwise.
static java.util.List	`getSubElements(org.w3c.dom.Element el)` Returns all elements under `el`, potentially empty list.
static org.w3c.dom.Document	`parseAsDocument(XmlString xmlText)` Parses `xmlText` and returns DOM document containing it on success, null on failure or if `xmlText` is null.
static org.w3c.dom.DocumentFragment	`parseAsFragment(XmlString xmlText)` Parses `xmlText` and returns DOM document fragment on success, null on failure or if `xmlText` is null.
static java.util.List	`selectElements(java.lang.String xpathExpression, java.lang.Object ctx, XmlNs[] namespaces)`
static org.w3c.dom.Node	`selectNode(java.lang.String xpathExpression, java.lang.Object ctx, XmlNs[] namespaces)`
static java.util.List	`selectNodes(java.lang.String xpathExpression, java.lang.Object ctx, XmlNs[] namespaces)`

Methods inherited from class `java.lang.Object`
`clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait`

Fields

INDENT

```
public static final java.lang.String INDENT
```

Constant value: **2**

Methods

createDocumentWithRoot

```
public static org.w3c.dom.Document createDocumentWithRoot(java.lang.String comment,
        java.lang.String rootName,
        XmlNs ns)
```

(continued from last page)

Returns a document with `rootName` element. If `ns` is not null, the root element is qualified with the namespace, and specifies the binding prefix/URI for that namespace.

Parameters:

`comment` - (potentially null or empty) document comment.
`rootName` - non-null, non-empty name of the root element.
`ns` - (potentially null) namespace definition.

addNamespace

```
public static void addNamespace(org.w3c.dom.Document dom,
         XmlNs ns)
```

Adds the namespace binding definition to DOM `dom` (to define multiple namespace bindings on the root element).

Parameters:

`dom` - non-null element to which to add the namespace binding.
`ns` - non-null namespace.

createQRoot

```
public static org.w3c.dom.Element createQRoot(java.lang.String qname,
         org.w3c.dom.Document document)
```

Creates root element, adds it to the document and returns that new root element.

createQSubElement

```
public static org.w3c.dom.Element createQSubElement(org.w3c.dom.Element el,
         java.lang.String qname,
         org.w3c.dom.Document document)
```

Creates sub-element, adds it to `el` and returns that new sub-element.

addQSubElement

```
public static org.w3c.dom.Element addQSubElement(org.w3c.dom.Element el,
         java.lang.String qname,
         org.w3c.dom.Document document)
```

Adds sub-element to `el` and returns modified `el`.

addCDATA

```
public static org.w3c.dom.Element addCDATA(org.w3c.dom.Element el,
         java.lang.String cdata,
         org.w3c.dom.Document document)
```

Adds CDATA section to `el` and returns modified `el`.

createXpath

```
public static javax.xml.xpath.XPath createXpath(XmlNs[] namespaces)
```

Returns the xpath instance that recognises all `namespaces`.

(continued from last page)

compileXpath

```
public static javax.xml.xpath.XPathExpression compileXpath(java.lang.String
xpathExpression,
         XmlNs[] namespaces)
```

Returns compiled xpath expression that recognises all `namespaces`.

selectNode

```
public static org.w3c.dom.Node selectNode(java.lang.String xpathExpression,
         java.lang.Object ctx,
         XmlNs[] namespaces)
```

selectNodes

```
public static java.util.List selectNodes(java.lang.String xpathExpression,
         java.lang.Object ctx,
         XmlNs[] namespaces)
```

selectElements

```
public static java.util.List selectElements(java.lang.String xpathExpression,
         java.lang.Object ctx,
         XmlNs[] namespaces)
```

parseAsDocument

```
public static org.w3c.dom.Document parseAsDocument(XmlString xmlText)
```

Parses `xmlText` and returns DOM document containing it on success, null on failure or if `xmlText` is null. Use this method when you require fully validated document.

parseAsFragment

```
public static org.w3c.dom.DocumentFragment parseAsFragment(XmlString xmlText)
```

Parses `xmlText` and returns DOM document fragment on success, null on failure or if `xmlText` is null. Use this method when you require fully validated document fragment.

asPrettyXml

```
public static XmlString asPrettyXml(org.w3c.dom.Node documentOrElement,
         java.io.File xmlFile)
```

Returns `documentOrElement` as pretty-print XML string and saves it to `xmlFile` if not null.

asXml

```
public static XmlString asXml(org.w3c.dom.Node documentOrElement,
         java.io.File xmlFile)
```

Returns `documentOrElement` as XML string and saves it to `xmlFile` if not null.

asPrettyXml

```
public static XmlString asPrettyXml(XmlString xmlText,
         java.io.File xmlFile)
```

Returns `xmlText` as pretty-print string and saves to `xmlFile` if not null.

getSubElements

```
public static java.util.List getSubElements(org.w3c.dom.Element el)
```

Returns all elements under `el`, potentially empty list.

getNamedSubElements

```
public static java.util.List getNamedSubElements(org.w3c.dom.Element el,
         java.lang.String name)
```

Returns elements under `el` with the `name` if found, empty list otherwise.

getFirstNamedSubElement

```
public static org.w3c.dom.Element getFirstNamedSubElement(org.w3c.dom.Element el,
         java.lang.String name)
```

Returns first element under `el` with the `name` if found, null otherwise.

getAttributes

```
public static java.util.List getAttributes(org.w3c.dom.Element el)
```

Returns all attributes of `el`, potentially empty list.

getNamedAttribute

```
public static org.w3c.dom.Attr getNamedAttribute(org.w3c.dom.Element el,
         java.lang.String name)
```

Returns attribute of `el` with the `name` if found, null otherwise.

org.tanjakostic.jcleancim.xml
Class NamespaceCache

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.xml.NamespaceCache
```

All Implemented Interfaces:
javax.xml.namespace.NamespaceContext

public class **NamespaceCache**
extends java.lang.Object
implements javax.xml.namespace.NamespaceContext

Adapted from Read the namespaces from the document and cache them

Constructor Summary

public	NamespaceCache(org.w3c.dom.Document document, boolean rootOnly) Constructor parses the document and stores all namespaces it can find.
public	NamespaceCache(XmlNs[] nsMappings) Constructor initialises the cache from non-null, non-empty nsMappings.

Method Summary

void	addMapping(java.lang.String prefix, java.lang.String uri) Adds mapping for prefix and uri to the cache.
java.util.List	getAllXmlNs()
java.lang.String	getNamespaceURI(java.lang.String prefix)
java.lang.String	getPrefix(java.lang.String namespaceURI)
java.util.Iterator	getPrefixes(java.lang.String namespaceURI)
XmlNs	getXmlNs(java.lang.String uri) Returns namespace instance if uri has been bound to a prefix, null otherwise.
java.lang.String	toString()

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface javax.xml.namespace.NamespaceContext

getNamespaceURI, getPrefix, getPrefixes

(continued from last page)

Constructors

NamespaceCache

```
public NamespaceCache(org.w3c.dom.Document document,
                      boolean rootOnly)
```

Constructor parses the document and stores all namespaces it can find.

Parameters:

`document` - source document
`rootOnly` - restriction of the search to enhance performance; only namespaces in the root are stored.

NamespaceCache

```
public NamespaceCache(XmlNs[] nsMappings)
```

Constructor initialises the cache from non-null, non-empty `nsMappings`.

Methods

addMapping

```
public void addMapping(java.lang.String prefix,
         java.lang.String uri)
```

Adds mapping for `prefix` and `uri` to the cache.

getXmlNs

```
public XmlNs getXmlNs(java.lang.String uri)
```

Returns namespace instance if `uri` has been bound to a prefix, null otherwise.

getAllXmlNs

```
public java.util.List getAllXmlNs()
```

toString

```
public java.lang.String toString()
```

getNamespaceURI

```
public java.lang.String getNamespaceURI(java.lang.String prefix)
```

Method called by XPath; returns the default namespace, if the prefix is null or "".

getPrefix

```
public java.lang.String getPrefix(java.lang.String namespaceURI)
```

(continued from last page)

getPrefixes

```
public java.util.Iterator getPrefixes(java.lang.String namespaceURI)
```

org.tanjakostic.jcleancim.xml
Class SaxErrorData

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.xml.SaxErrorData
```

public class **SaxErrorData**
extends java.lang.Object

Simple storage for XML validation warnings and errors.

Constructor Summary

public	SaxErrorData()

Method Summary

void	addError(java.lang.String error)
void	addFatal(java.lang.String fatal)
void	addWarn(java.lang.String warn)
java.util.List	getAll()
java.util.List	getErrors()
java.util.List	getFatals()
java.util.List	getWarns()
boolean	hasErrorOrFatal()
boolean	isEmpty() Returns true if neither of warning, error or fatal has been stored.
void	reset() If you are using this instance multiple times, ensure you always first reset it.
java.lang.String	toString()

Methods inherited from class `java.lang.Object`

`clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait`

Constructors

(continued from last page)

SaxErrorData

```
public SaxErrorData()
```

Methods

addWarn

```
public void addWarn(java.lang.String warn)
```

getWarns

```
public java.util.List getWarns()
```

addError

```
public void addError(java.lang.String error)
```

getErrors

```
public java.util.List getErrors()
```

addFatal

```
public void addFatal(java.lang.String fatal)
```

getFatals

```
public java.util.List getFatals()
```

isEmpty

```
public boolean isEmpty()
```

Returns true if neither of warning, error or fatal has been stored.

hasErrorOrFatal

```
public boolean hasErrorOrFatal()
```

(continued from last page)

getAll

```
public java.util.List getAll()
```

reset

```
public void reset()
```

If you are using this instance multiple times, ensure you always first reset it.

toString

```
public java.lang.String toString()
```

org.tanjakostic.jcleancim.xml
Class WellformedDOM

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.xml.WellformedDOM
```

Direct Known Subclasses:
XmlSchemaDOM, XmlDocument

public abstract class **WellformedDOM**
extends java.lang.Object

Abstract class as a supertype for a DOM document that will not use validation, such as for XML schema or a simple XML where we don't care about the validation.

Constructor Summary

protected	WellformedDOM(java.lang.String filePath, XmlString content)

Method Summary

java.io.InputStream	asInputStream()
XmlString	asXmlString()
org.w3c.dom.Document	getDocument() Returns DOM document.
java.io.File	getFile()
NamespaceCache	getNsCache()

Methods inherited from class `java.lang.Object`
clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Constructors

WellformedDOM

```
protected WellformedDOM(java.lang.String filePath,
                        XmlString content)
```

Methods

(continued from last page)

getFile

```
public java.io.File getFile()
```

asXmlString

```
public XmlString asXmlString()
```

asInputStream

```
public java.io.InputStream asInputStream()
```

getDocument

```
public org.w3c.dom.Document getDocument()
```

Returns DOM document.

getNsCache

```
public NamespaceCache getNsCache()
```

org.tanjakostic.jcleancim.xml
Class WellformedDOMBuilder

```
java.lang.Object
  |-org.tanjakostic.jcleancim.xml.AbstractConfiguredDOMBuilder
    |-org.tanjakostic.jcleancim.xml.WellformedDOMBuilder
```

All Implemented Interfaces:
ConfiguredDOMBuilder

Direct Known Subclasses:
InternalDtdValidatingDOMBuilder

---

public class **WellformedDOMBuilder**
extends AbstractConfiguredDOMBuilder

SAX reader configured to check wellformed-ness only.

Constructor Summary

public	WellformedDOMBuilder()
protected	WellformedDOMBuilder(boolean respectDtd)

Method Summary

org.w3c.dom.Document	emptyDocument()

Methods inherited from class org.tanjakostic.jcleancim.xml.AbstractConfiguredDOMBuilder
getDOMBuilder, getDOMBuilderFactory, getErrorHandler, getParsingErrors, getSchemaFactory, readAndValidate, readAndValidate, readAndValidate

Methods inherited from class java.lang.Object
clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Methods inherited from interface org.tanjakostic.jcleancim.xml.ConfiguredDOMBuilder
getParsingErrors, readAndValidate, readAndValidate, readAndValidate

---

Constructors

WellformedDOMBuilder

public **WellformedDOMBuilder**()

WellformedDOMBuilder

```
protected WellformedDOMBuilder(boolean respectDtd)
```

Methods

emptyDocument

```
public org.w3c.dom.Document emptyDocument()
```

org.tanjakostic.jcleancim.xml
Class XmlException

```
java.lang.Object
  +-java.lang.Throwable
      +-java.lang.Exception
          +-java.lang.RuntimeException
              +-org.tanjakostic.jcleancim.xml.XmlException
```

All Implemented Interfaces:
java.io.Serializable

---

public class **XmlException**
extends java.lang.RuntimeException

Wrapper for any XML-related exceptions not involving validation.

Constructor Summary

public	XmlException(java.lang.String message)
public	XmlException(java.lang.Throwable cause)
public	XmlException(java.lang.String message, java.lang.Throwable cause)

Methods inherited from class `java.lang.Throwable`
addSuppressed, fillInStackTrace, getCause, getLocalizedMessage, getMessage, getStackTrace, getSuppressed, initCause, printStackTrace, printStackTrace, printStackTrace, setStackTrace, toString

Methods inherited from class `java.lang.Object`
clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

---

Constructors

XmlException

public **XmlException**(java.lang.String message)

XmlException

public **XmlException**(java.lang.Throwable cause)

XmlException

```
public XmlException(java.lang.String message,
                    java.lang.Throwable cause)
```

org.tanjakostic.jcleancim.xml
Class XmlInstanceDOM

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.xml.XmlInstanceDOM
```

public abstract class **XmlInstanceDOM**
extends java.lang.Object

Common implementation for all XML instance documents.

Constructor Summary

protected	XmlInstanceDOM(java.lang.String comment, java.lang.String instancePath, java.lang.String schemaPath, java.lang.String rootTag) FIXME: test
protected	XmlInstanceDOM(java.lang.String comment, java.lang.String instancePath, java.lang.String schemaPath) Constructs this instance with empty qualified root element, and with reference to schema found in schemaPath; the root element tag is deduced from the schema.
protected	XmlInstanceDOM(java.lang.String comment, java.lang.String instancePath, XmlSchemaDOM schema, java.lang.String rootTag) FIXME: test
protected	XmlInstanceDOM(java.lang.String comment, java.lang.String instancePath, XmlSchemaDOM schema) Constructs this instance with empty qualified root element, and with schema, which potentially does not have the file representation (e.g., it may have been created from XML text, and does not exist as a file); the root element tag is deduced from the schema.

Method Summary

org.w3c.dom.Element	addCDATA(org.w3c.dom.Element el, java.lang.String cdata) Adds CDATA section to el and returns modified el.
org.w3c.dom.Element	addSubElement(org.w3c.dom.Element el, java.lang.String qname) Adds sub-element to el and returns modified el.
org.w3c.dom.Element	createSubElement(org.w3c.dom.Element el, java.lang.String name) Creates new element (by qualifying its name with the target namespace prefix), adds it under el and returns that new element.
org.w3c.dom.Element	createSubElementUnderRoot(java.lang.String name) Creates new element (by qualifying its name with the target namespace prefix), adds it under document root and returns that new element.
org.w3c.dom.Document	getDocument()
java.io.File	getInstanceFile()
java.lang.String	getPrettyXml()

org.w3c.dom.Element	getRoot()
XmlSchemaDOM	getSchema()
XmlNs	getTargetNs()
java.lang.String	qname(java.lang.String name)
void	save()
SaxErrorData	validate() If initialised with an external schema, validates this instance document against that schema.

Methods inherited from class `java.lang.Object`
`clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait`

Constructors

XmlInstanceDOM

```
protected XmlInstanceDOM(java.lang.String comment,
                         java.lang.String instancePath,
                         java.lang.String schemaPath,
                         java.lang.String rootTag)
```

FIXME: test

Constructs this instance with empty qualified root element, and with reference to schema found in `schemaPath`.

Parameters:

`comment` - (potentially null or empty) document comment.
`instancePath` - path where this document can be saved as file.
`schemaPath` - path where the schema can be found as file.
`rootTag` - root element name

Throws:

XmlParsingException

XmlInstanceDOM

```
protected XmlInstanceDOM(java.lang.String comment,
                         java.lang.String instancePath,
                         java.lang.String schemaPath)
```

Constructs this instance with empty qualified root element, and with reference to schema found in `schemaPath`; the root element tag is deduced from the schema.

Parameters:

`comment` - (potentially null or empty) document comment.
`instancePath` - path where this document can be saved as file.
`schemaPath` - path where the schema can be found as file.

Throws:

(continued from last page)

XmlParsingException

XmlInstanceDOM

```
protected XmlInstanceDOM(java.lang.String comment,
                         java.lang.String instancePath,
                         XmlSchemaDOM schema,
                         java.lang.String rootTag)
```

FIXME: test

Constructs this instance with empty qualified root element, and with `schema`, which potentially does not have the file representation (e.g., it may have been created from XML text, and does not exist as a file). This one is useful for testing.

Parameters:

`comment`
`instancePath`
`schema`
`rootTag`

Throws:

XmlParsingException

XmlInstanceDOM

```
protected XmlInstanceDOM(java.lang.String comment,
                         java.lang.String instancePath,
                         XmlSchemaDOM schema)
```

Constructs this instance with empty qualified root element, and with `schema`, which potentially does not have the file representation (e.g., it may have been created from XML text, and does not exist as a file); the root element tag is deduced from the schema. This one is useful for testing.

Parameters:

`comment`
`instancePath`
`schema`

Throws:

XmlParsingException

Methods

validate

```
public SaxErrorData validate()
```

If initialised with an external schema, validates this instance document against that schema. Otherwise, validates against the schema that may be specified in the instance document as schema location.

Throws:

XmlParsingException

save

```
public final void save()
```

(continued from last page)

qname

```
public final java.lang.String qname(java.lang.String name)
```

getPrettyXml

```
public final java.lang.String getPrettyXml()
```

getInstanceFile

```
public java.io.File getInstanceFile()
```

getSchema

```
public XmlSchemaDOM getSchema()
```

getTargetNs

```
public XmlNs getTargetNs()
```

getRoot

```
public org.w3c.dom.Element getRoot()
```

getDocument

```
protected org.w3c.dom.Document getDocument()
```

createSubElementUnderRoot

```
public org.w3c.dom.Element createSubElementUnderRoot(java.lang.String name)
```

Creates new element (by qualifying its `name` with the target namespace prefix), adds it under document root and returns that new element.

createSubElement

```
public org.w3c.dom.Element createSubElement(org.w3c.dom.Element el,
         java.lang.String name)
```

Creates new element (by qualifying its `name` with the target namespace prefix), adds it under `el` and returns that new element.

(continued from last page)

addSubElement

```
public org.w3c.dom.Element addSubElement(org.w3c.dom.Element el,
         java.lang.String qname)
```

Adds sub-element to `el` and returns modified `el`.

addCDATA

```
public org.w3c.dom.Element addCDATA(org.w3c.dom.Element el,
         java.lang.String cdata)
```

Adds CDATA section to `el` and returns modified `el`.

org.tanjakostic.jcleancim.xml
Class XmlNs

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.xml.XmlNs
```

Direct Known Subclasses:
XmlNamespace

public class **XmlNs**
extends java.lang.Object

Namespace mappings for known and unknown namespaces.

Field Summary

public static final	FRAG_SEP Value: **#**
public static final	xsi

Constructor Summary

public	XmlNs(java.lang.String prefix, java.lang.String uri)
public	XmlNs(java.lang.String prefix, java.lang.String uri, NamespaceCache cache)

Method Summary

java.lang.String	getPrefix() Returns the prefix used for this namespace (e.g., "rdf").
java.lang.String	getUri() Returns URI of this namespace, as found in the root element (e.g., "http://...#").
java.lang.String	getUriWithoutFragmentSeparator() Returns getUri() with fragment separator trimmed out.
java.lang.String	qName(java.lang.String name) Returns qualified name for name with this instance's prefix as qualifier (e.g., "rdf:name" or "name" if namespace prefix null or empty).
java.lang.String	toString()

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Fields

xsi

```
public static final org.tanjakostic.jcleancim.xml.XmlNs xsi
```

FRAG_SEP

```
public static final java.lang.String FRAG_SEP
```

Constant value: **#**

Constructors

XmlNs

```
public XmlNs(java.lang.String prefix,
             java.lang.String uri)
```

XmlNs

```
public XmlNs(java.lang.String prefix,
             java.lang.String uri,
             NamespaceCache cache)
```

Methods

getPrefix

```
public final java.lang.String getPrefix()
```

Returns the prefix used for this namespace (e.g., "rdf").

qName

```
public java.lang.String qName(java.lang.String name)
```

Returns qualified name for `name` with this instance's prefix as qualifier (e.g., "rdf:name" or "name" if namespace prefix null or empty).

Parameters:

`name` - name to qualify.

getUri

```
public final java.lang.String getUri()
```

Returns URI of this namespace, as found in the root element (e.g., "http://...#").

(continued from last page)

getUriWithoutFragmentSeparator

`public final java.lang.String` **`getUriWithoutFragmentSeparator`**`()`

Returns `getUri()` with fragment separator trimmed out.

toString

`public java.lang.String` **`toString`**`()`

org.tanjakostic.jcleancim.xml
Class XmlParsingException

```
java.lang.Object
  |-java.lang.Throwable
    |-java.lang.Exception
      |-java.lang.RuntimeException
        |-org.tanjakostic.jcleancim.xml.XmlParsingException
```

All Implemented Interfaces:
java.io.Serializable

---

public class **XmlParsingException**
extends java.lang.RuntimeException

Wrapper for the underlying parsing/validation exceptions. The first fatal or error condition is available from `getErrorData()`.

Constructor Summary

public	`XmlParsingException(java.lang.String message, SaxErrorData errorData, java.lang.Throwable cause)` Constructor - carries concrete XML validation errors that can be manipulated.

Method Summary

`SaxErrorData`	`getErrorData()`

Methods inherited from class `java.lang.Throwable`

`addSuppressed, fillInStackTrace, getCause, getLocalizedMessage, getMessage, getStackTrace, getSuppressed, initCause, printStackTrace, printStackTrace, printStackTrace, setStackTrace, toString`

Methods inherited from class `java.lang.Object`

`clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait`

Constructors

XmlParsingException

```
public XmlParsingException(java.lang.String message,
                           SaxErrorData errorData,
                           java.lang.Throwable cause)
```

Constructor - carries concrete XML validation errors that can be manipulated.

Methods

(continued from last page)

getErrorData

```
public SaxErrorData getErrorData()
```

org.tanjakostic.jcleancim.xml
Class XmlSchemaDOM

```
java.lang.Object
  |-org.tanjakostic.jcleancim.xml.WellformedDOM
    |-org.tanjakostic.jcleancim.xml.XmlSchemaDOM
```

public class **XmlSchemaDOM**
extends WellformedDOM

Representaton of an XML schema document; provides some helper methods to facilitate creating instance XML documents compliant with this schema.

Constructor Summary

public	XmlSchemaDOM(java.lang.String schemaPath) Constructs schema document from the file schemaPath.
public	XmlSchemaDOM(XmlString schemaContent) Constructs schema document from the string content schemaContent; WellformedDOM.getFile() will return null.

Method Summary

java.lang.String	getRootTag()
XmlNs	getTargetNs()
SaxErrorData	validate(XmlInstanceDOM instanceDOM) Validates instanceDOM against this schema DOM and returns parsing errors.
static SaxErrorData	validate(XmlInstanceDOM instance, XmlSchemaDOM schema) Validates instance against schema and returns parsing errors.

Methods inherited from class org.tanjakostic.jcleancim.xml.WellformedDOM

asInputStream, asXmlString, getDocument, getFile, getNsCache

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Constructors

XmlSchemaDOM

public **XmlSchemaDOM**(java.lang.String schemaPath)

Constructs schema document from the file schemaPath.

(continued from last page)

Parameters:
`schemaPath` - non-null valid path of the schema.

XmlSchemaDOM

```
public XmlSchemaDOM(XmlString schemaContent)
```

Constructs schema document from the string content `schemaContent`; `WellformedDOM.getFile()` will return null.

Parameters:
`schemaContent` - non-null, non-empty schema content as XML string.

Methods

getRootTag

```
public java.lang.String getRootTag()
```

getTargetNs

```
public XmlNs getTargetNs()
```

validate

```
public SaxErrorData validate(XmlInstanceDOM instanceDOM)
```

Validates `instanceDOM` against this schema DOM and returns parsing errors.

validate

```
public static SaxErrorData validate(XmlInstanceDOM instance,
         XmlSchemaDOM schema)
```

Validates `instance` against `schema` and returns parsing errors.

org.tanjakostic.jcleancim.xml
Class XmlString

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.xml.XmlString
```

public class **XmlString**
extends java.lang.Object

Wrapper for a java string; allows us to use consistent approach for instantiation from XML string in case there are classes that require a single non-XML string argument in constructor.

Constructor Summary

public	XmlString(java.lang.String s)

Method Summary

boolean	equals(java.lang.Object obj)
int	hashCode()
java.lang.String	toString()

Methods inherited from class java.lang.Object
clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Constructors

XmlString

public **XmlString**(java.lang.String s)

Methods

toString

public java.lang.String **toString**()

hashCode

public int **hashCode**()

(continued from last page)

equals

```
public boolean equals(java.lang.Object obj)
```

org.tanjakostic.jcleancim.xml
Class XmlUtil

```
java.lang.Object
  |
  +-org.tanjakostic.jcleancim.xml.XmlUtil
```

public class **XmlUtil**
extends java.lang.Object

Field Summary

public static final	ENCODING Value: **UTF-8**

Method Summary

static org.xml.sax.InputSource	xmlAsInputSource(XmlString xmlText) Returns input source with #ENCODING encoding for XML content.
static java.io.InputStream	xmlAsInputStream(java.io.File xmlFile, boolean isOnClasspath) Returns input stream for file.
static java.io.InputStream	xmlAsInputStream(java.lang.String xmlFileUri, boolean isOnClasspath) Returns input stream for file path, on classpath or not.
static java.io.InputStream	xmlAsInputStream(XmlString xmlText) Returns input stream for XML content.

Methods inherited from class `java.lang.Object`

`clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait`

Fields

ENCODING

`public static final java.lang.String` **`ENCODING`**

Constant value: **UTF-8**

Methods

(continued from last page)

xmlAsInputStream

```
public static java.io.InputStream xmlAsInputStream(java.io.File xmlFile,
          boolean isOnClasspath)
  throws XmlParsingException,
          ResourceNotOnClasspathException
```

Returns input stream for file.

Parameters:

`xmlFile` - non-null file.
`isOnClasspath` - whether to search for `xmlFileUri` on the classpath.

Throws:

`ResourceNotOnClasspathException`
`XmlParsingException`

xmlAsInputStream

```
public static java.io.InputStream xmlAsInputStream(java.lang.String xmlFileUri,
          boolean isOnClasspath)
  throws XmlException,
          ResourceNotOnClasspathException
```

Returns input stream for file path, on classpath or not.

Parameters:

`xmlFileUri` - non-null, non-empty file URI.
`isOnClasspath` - whether to search for `xmlFileUri` on the classpath.

Throws:

`ResourceNotOnClasspathException`
`XmlParsingException`

xmlAsInputStream

```
public static java.io.InputStream xmlAsInputStream(XmlString xmlText)
```

Returns input stream for XML content.

Parameters:

`xmlText`

Throws:

`XmlException`

xmlAsInputSource

```
public static org.xml.sax.InputSource xmlAsInputSource(XmlString xmlText)
```

Returns input source with #ENCODING encoding for XML content.

Parameters:

`xmlText`

Throws:

`XmlParsingException`