
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)

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 [Placeholders](#).

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.

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 decrease 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:

- 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 decrease 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

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:

```

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 every 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.

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 successfully 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.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 crosscheck 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.
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:

[ApplicationException](#)

(continued on next page)

(continued from last page)

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

generateDoc

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

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

Config	getCfg() Returns the configuration, containing also diagram export options.
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 [Config](#) **getCfg()**

Returns the configuration, containing also diagram export options.

saveToFile

public 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

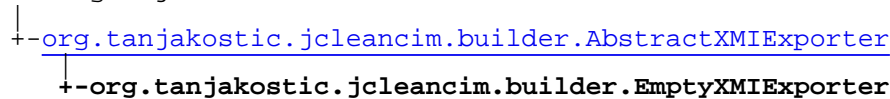
```
protected XMIExporter createXMIExporter()
```

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

org.tanjakostic.jcleancim.builder

Class EmptyXMIExporter

java.lang.Object



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

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

Methods

getCfg

public [Config](#) **getCfg()**

Returns configuration.

build

public [UmlModel](#) **build()**
throws [ApplicationException](#)

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

getDiagramExporter

public [DiagramExporter](#) **getDiagramExporter()**

getXMIExporter

public [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

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

Methods

build

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

Returns:
FIXME

build

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

Parameters:
model

Returns:
FIXME

getObjData

```
public UmlObjectData getObjData()
```

Returns:
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

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

Methods

getCfg

public [Config](#) **getCfg**()

Returns the configuration, containing also XMI export options.

exportToXMIs

public 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)
UmlClass	build () This default implementation ; 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	<u>collectDiagrams</u> (java.lang.Object itemsSrc)
abstract java.util.List	<u>collectOperations</u> (java.lang.Object itemsSrc)
abstract java.util.List	<u>collectTaggedValues</u> (java.lang.Object itemsSrc)
abstract <u>AssociationBuilder</u>	<u>createAssociation</u> (java.lang.Object item, <u>ClassBuilder</u> source, <u>ClassBuilder</u> target, <u>EaModelBuilder</u> model, <u>EaHelper</u> eaHelper)
abstract <u>AttributeBuilder</u>	<u>createAttribute</u> (java.lang.Object item, <u>EaHelper</u> eaHelper)
abstract <u>DependencyBuilder</u>	<u>createDependency</u> (java.lang.Object item, <u>EaModelBuilder</u> model, <u>ClassBuilder</u> source, <u>ClassBuilder</u> target, <u>EaHelper</u> eaHelper)
abstract <u>DiagramBuilder</u>	<u>createDiagram</u> (java.lang.Object item, <u>EaHelper</u> eaHelper)
abstract <u>ClassBuilder</u>	<u>createEmbeddedClass</u> (java.lang.Object item, <u>EaHelper</u> 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 <u>OperationBuilder</u>	<u>createOperation</u> (java.lang.Object item, <u>EaHelper</u> eaHelper)
abstract <u>SkippedBuilder</u>	<u>createSkippedConnector</u> (java.lang.Object item, <u>EaModelBuilder</u> model, <u>EaHelper</u> eaHelper)
abstract <u>SkippedBuilder</u>	<u>createSkippedElement</u> (java.lang.Object item, <u>EaModelBuilder</u> model, <u>EaHelper</u> eaHelper)
void	<u>doBuild</u> ()
abstract java.util.Map	<u>eaConnectorIDsToFields</u> (java.lang.Object item)
abstract java.lang.String	<u>fetchConnectorType</u> (java.lang.Object item)
java.util.List	<u>getAssociationSourceEndClasses</u> () For associations where I'm on the target end, returns the list of source end classes.
java.util.List	<u>getAssociationTargetEndClasses</u> () For associations where I'm on the source end, returns the list of target end classes.
java.util.List	<u>getAttributes</u> ()
java.util.Map	<u>getConstraints</u> ()
<u>PackageBuilder</u>	<u>getContainingPackage</u> ()
java.util.List	<u>getDependencyAfferentClasses</u> () Returns classes that depend on me through explicit UML dependency in the model.
java.util.List	<u>getDependencyEfferentClasses</u> () 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	<u>isAbstract()</u>
boolean	<u>isAssociationClass()</u>
static boolean	<u>isClassOrEaInterface</u> (java.lang.String eaType) Returns whether EA type is an EA class or an EA interface.
boolean	<u>isEaInterface()</u>
boolean	<u>isEaLeafPropSet()</u>
boolean	<u>isEaPersistentPropSet()</u>
boolean	<u>isEaRootPropSet()</u>
boolean	<u>isEnumeratedType()</u>
boolean	<u>isSelfDependent()</u>
boolean	<u>isSelfInherited()</u>

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)
```

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()
```

(continued from last page)

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()
```

isAssociationClass

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

(continued from last page)

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)
```

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()
```

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.

(continued from last page)

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	ensureDependenciesOfEndClassesInitialised () 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()
```

ensureDependenciesOfEndClassesInitialised

```
public final void ensureDependenciesOfEndClassesInitialised()
```

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
protected static final	ATTR_CONSTR_TAGS
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
protected static final	ATTR_TGVAL_TAGS
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
protected static final	CLASS_CONSTR_TAGS
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
protected 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
protected static final	CONN_TAGS_OUT
public static final	CONN_TGVAL_NAME Value: Property
protected static final	CONN_TGVAL_OWNER_ID Value: ElementID
protected static final	CONN_TGVAL_TAGS
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
protected 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_PKG_ID Value: Package_ID
protected static final	DIA_POS Value: TPos
public static final	DIA_STEREO Value: Stereotype
protected static final	DIA_TAGS
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
protected static final	ELEM_TGVAL_TAGS
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
protected static final	OP_TGVAL_TAGS
public static final	OP_TGVAL_VALUE Value: VALUE
protected static final	PACKAGE_AS_ELEM_TAGS these are defined only on elements (i.e., package type in elements table - sic!)
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
protected static final	PACKAGE_TAGS these are proper to package table; see more in PACKAGE_AS_ELEM_TAGS below
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
protected static final	ROLE_TGVAL_TAGS
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_TAGS

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) Extracts 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**

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**

XREF_TAGS

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

PACKAGE_OWNER_ID

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

(continued from last page)

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
```

Constant value: **TPos**

PACKAGE_TAGS

```
protected static final java.lang.String PACKAGE_TAGS
```

these are proper to package table; see more in PACKAGE_AS_ELEM_TAGS below

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**

(continued from last page)

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**

DIA_TAGS

```
protected static final java.lang.String DIA_TAGS
```

ELEM_ID

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

Constant value: **Object_ID**

ELEM_NAME

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

(continued from last page)

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**

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**

(continued from last page)

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
```

PACKAGE_AS_ELEM_TAGS

```
protected static final java.lang.String PACKAGE_AS_ELEM_TAGS
```

these are defined only on elements (i.e., package type in elements table - sic!)

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**

CLASS_CONSTR_TAGS

```
protected static final java.lang.String CLASS_CONSTR_TAGS
```

ELEM_TGVAL_NAME

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

Constant value: **Property**

ELEM_TGVAL_VALUE

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

Constant value: **Value**

ELEM_TGVAL_TAGS

```
protected static final java.lang.String ELEM_TGVAL_TAGS
```

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**

ATTR_STATIC

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

(continued from last page)

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
```

Constant value: **Property**

(continued from last page)

ATTR_TGVAL_VALUE

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

Constant value: **VALUE**

ATTR_TGVAL_TAGS

```
protected static final java.lang.String ATTR_TGVAL_TAGS
```

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**

ATTR_CONSTR_TAGS

```
protected static final java.lang.String ATTR_CONSTR_TAGS
```

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
```

(continued from last page)

Constant value: **Style**

OP_NOTE

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

Constant value: **Notes**

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**

(continued from last page)

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
```

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**

OP_TGVAL_TAGS

```
protected static final java.lang.String OP_TGVAL_TAGS
```

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
```

(continued from last page)

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**

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**

(continued from last page)

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
```

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
```

(continued from last page)

Constant value: **DestRole**

CONN_FROM_STYLE

```
protected static final java.lang.String CONN_FROM_STYLE
```

Constant value: **SourceStyle**

CONN_TO_STYLE

```
protected static final java.lang.String CONN_TO_STYLE
```

Constant value: **DestStyle**

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
```

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**

(continued from last page)

CONN_TAGS_OUT

```
protected static final java.lang.String CONN_TAGS_OUT
```

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**

CONN_TGVAL_TAGS

```
protected static final java.lang.String CONN_TGVAL_TAGS
```

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
```

(continued from last page)

Constant value: **Notes**

ROLE_TGVAL_TAGS

```
protected static final java.lang.String ROLE_TGVAL_TAGS
```

Constructors

EA

```
protected EA()
```

Methods

extractStereotypes

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

Extracts 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.

extractNavigability

```
public 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.

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	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 ()
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

(continued from last page)

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()
```

createModelPackage

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

(continued from last page)

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)

findClass

public final [ClassBuilder](#) **findClass**(java.lang.Integer typeId)

(continued from last page)

findElementTypeAndName

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

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)
```

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

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

Methods

select

```
public 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

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

Methods

sqlResultAsXml

public 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	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).

Parameters:

selector - accesses EA data

(continued from last page)

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)
```

(continued from last page)

Returns type and name (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)
```

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.

(continued from last page)

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

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

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)
```

(continued from last page)

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)
```

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)
```

collectPackageElements

```
protected abstract java.util.List collectPackageElements(java.lang.Object itemsSrcP)
```

(continued from last page)

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()
```

getDepth

```
public final int getDepth()
```

(continued from last page)

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()
```

getChildPackages

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

(continued from last page)

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.

getClassUids

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

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)
```

(continued from last page)

This default implementation 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 implementation 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	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](#), [findElementTypeAndName](#), [getFirstRoot](#), [getLogSubtitleEndPopulateBuilders](#), [getLogSubtitleStartPopulateBuilders](#), [getModels](#), [getTables](#), [initRepoAndGetVersion](#), [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()
```

(continued from last page)

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)
```

findElementTypeAndName

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

(continued from last page)

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](#)

[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	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](#), [findElementTypeAndName](#), [getFirstRoot](#), [getLogSubtitleEndPopulateBuilders](#), [getLogSubtitleStartPopulateBuilders](#), [getModels](#), [getTables](#), [initRepoAndGetVersion](#), [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)
```

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)
```

fetchPackageGuid

```
protected java.lang.String fetchPackageGuid(org.sparx.Package inData)
```

(continued from last page)

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

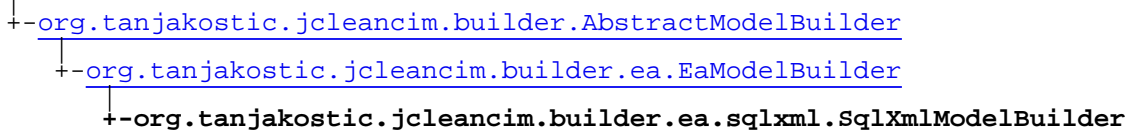
```
public static java.util.List eaToJavaList(org.sparx.Collection eaCollection)
```

Package

**org.tanjakostic.jcleancim.builder.ea.sqlx
ml**

org.tanjakostic.jcleancim.builder.ea.sqlxml Class SqlXmlModelBuilder

java.lang.Object



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	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](#), [findElementTypeAndName](#), [getFirstRoot](#), [getLogSubtitleEndPopulateBuilders](#), [getLogSubtitleStartPopulateBuilders](#), [getModels](#), [getTables](#), [initRepoAndGetVersion](#), [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)
```

(continued from last page)

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)
```

findElementTypeAndName

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

fetchPackageGuid

```
protected java.lang.String fetchPackageGuid(java.util.Map inData)
```

(continued from last page)

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 template 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 seamless 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_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:
 - 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 Access 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 validation 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 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".

At present, there are only 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.

Document generation properties (both MS Word and 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 the next release).

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 outputting 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_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".

MS Word document generation properties

- (started implementation in 01v08, not complete yet) 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.
- 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_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_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 containing "ERROR" indicates that all the placeholders are OK.
- Property `#KEY_DOCGEN_WORD_PRINT_HTML`, if set "true", will allow to respect markup formatting in the documentation of elements in the UML repository when generating MS Word document. 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 formatted documentation, but it is heavily used in IEC61850 model (e.g., for formulae). Consider that enabling formatted output results in ~1.5 time longer MS Word document generation! (TODO: In this release of jCleanCim, this functionality is still not working properly, so you will not be happy with the output; this should be fixed in some of coming releases.)

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).

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_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_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_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_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_PRINT_HTML Enable formatted documentation of UML model elements be formatted same in the generated MS Word document = "true" (default = "false", "", null). Value: docgen.word.printHtml
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_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_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_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.relpPath
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_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.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 <code>#KEY_DOCGEN_WORD_SAVE_REOPEN_EVERY</code> .
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.
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	<u>getPicsDirAbsPath()</u> Returns absolute path string for directory where to export images from the model.
java.util.Map	<u>getProfileFiles()</u>
java.lang.String	<u>getProfilesRelpath()</u>
java.lang.String	<u>getValidationIec61850Package72Top()</u>
java.lang.String	<u>getValidationIec61850PackageFc()</u>
java.lang.String	<u>getValidationIec61850PackageLnMaps()</u>
java.lang.String	<u>getValidationIec61850PackageMetaModel()</u>
java.lang.String	<u>getValidationIec61850PackagePresCond()</u>
java.util.Collection	<u>getValidationIec61850Packages72()</u>
java.util.Collection	<u>getValidationIec61850Packages73()</u>
java.util.Collection	<u>getValidationIec61850Packages74()</u>
java.util.Collection	<u>getValidationIec61850PackagesBasic()</u>
java.util.Collection	<u>getValidationIec61850PackagesCdc()</u>
java.util.Collection	<u>getValidationIec61850PackagesDa()</u>
java.util.Collection	<u>getValidationIec61850PackagesDoAbbr()</u>
java.util.Collection	<u>getValidationIec61850PackagesDocgen()</u> Returns the union of <u>getValidationIec61850Packages72()</u> , <u>getValidationIec61850Packages73()</u> and <u>getValidationIec61850Packages74()</u> .
java.util.Collection	<u>getValidationIec61850PackagesEnumsXml()</u>
java.util.Collection	<u>getValidationIec61850PackagesExtTitle()</u> Returns the union of <u>getValidationIec61850PackagesLn()</u> , <u>getValidationIec61850PackagesCdc()</u> and .
java.util.Collection	<u>getValidationIec61850PackagesLn()</u>
java.lang.String	<u>getValidationIec61850PackageTrgOp()</u>
java.util.Collection	<u>getValidationPackagesDataIndex()</u>
java.util.Collection	<u>getValidationRulesOff()</u>

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	isDocgenWordAnalysePlaceholders() Returns whether to only analyse placeholders in output MS Word document, without replacing them with the full content.
boolean	isDocgenWordIntroToFigureBefore()
boolean	isDocgenWordPrintHtml() Returns whether to respect markup (present in UML descriptions) in output MS Word document.
boolean	isDocgenWordUseDocFormat()
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**

(continued from last page)

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.relp**

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
```

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**

(continued from last page)

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
```

(continued from last page)

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**

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
```

(continued from last page)

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).

(continued from last page)

Constant value: **statistics.on**

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**

(continued from last page)

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_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_PRINT_HTML

```
public static final java.lang.String KEY_DOCGEN_WORD_PRINT_HTML
```

Enable formatted documentation of UML model elements be formatted same in the generated MS Word document = "true" (default = "false", "", null).
Constant value: **docgen.word.printHtml**

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_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**

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**

(continued from last page)

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
```

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
```

(continued from last page)

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**

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**

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.

(continued from last page)

modelName - 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

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 elapsed 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.

(continued from last page)

isXmiexportOn

```
public boolean isXmiexportOn()
```

Returns whether export to XMI is enabled; applicable only if the source of the model is an .eap file.

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()
```

(continued from last page)

isValidationDiagramsOn

```
public boolean isValidationDiagramsOn()
```

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()
```

(continued from last page)

getValidationIec61850PackagesEnumsXml

```
public java.util.Collection getValidationIec61850PackagesEnumsXml()
```

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()
```

(continued from last page)

getValidationIec61850PackagesDoAbbr

```
public java.util.Collection getValidationIec61850PackagesDoAbbr()
```

getValidationIec61850PackageLnMaps

```
public java.lang.String getValidationIec61850PackageLnMaps()
```

isStatisticsOn

```
public boolean isStatisticsOn()
```

isStatisticsCimIgnoreIdObjectInheritance

```
public boolean isStatisticsCimIgnoreIdObjectInheritance()
```

isStatisticsCimIgnoreDomainClassAttributes

```
public boolean isStatisticsCimIgnoreDomainClassAttributes()
```

isProfilesCrosscheckOn

```
public boolean isProfilesCrosscheckOn()
```

isDocgenOn

```
public boolean isDocgenOn()
```

isDocgenWordUseDocFormat

```
public boolean isDocgenWordUseDocFormat()
```

isDocgenWordIntroToFigureBefore

```
public boolean isDocgenWordIntroToFigureBefore()
```

(continued from last page)

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.

isDocgenWordPrintHtml

```
public boolean isDocgenWordPrintHtml()
```

Returns whether to respect markup (present in UML descriptions) in output MS Word document.

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()
```

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.

(continued from last page)

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 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 Pure CIM domain.
public static final	IEC61850 Pure IEC 61850 domain.

Method Summary

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

Fields

CIM

public static final org.tanjakostic.jcleancim.common.Nature **CIM**

(continued from last page)

Pure 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)
```

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 two standard model packages (one that contains CIM top-level packages, and one that contains non-CIM 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	<u>getAllowedOtherEndOwners()</u> Returns the allowed dependencies of this owner, as per IEC TC57 top-level package dependencies rules.
java.lang.String	<u>getAppDomain()</u>
<u>Nature</u>	<u>getNature()</u>
static <u>OwningWg</u>	<u>getOwnerForTopPackage()</u> (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	<u>getReservedTopPackageNames()</u> Returns reserved (standard) top package names.
java.lang.String	<u>getTopPackageName()</u>
boolean	<u>involvedIn()</u> (<u>OwningWg</u> oneEnd, <u>OwningWg</u> otherEnd) Returns whether one of the arguments has this owner.
static <u>OwningWg</u>	<u>valueOf()</u> (java.lang.String name)
static <u>OwningWg[]</u>	<u>values()</u>

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
```

(continued from last page)

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()
```

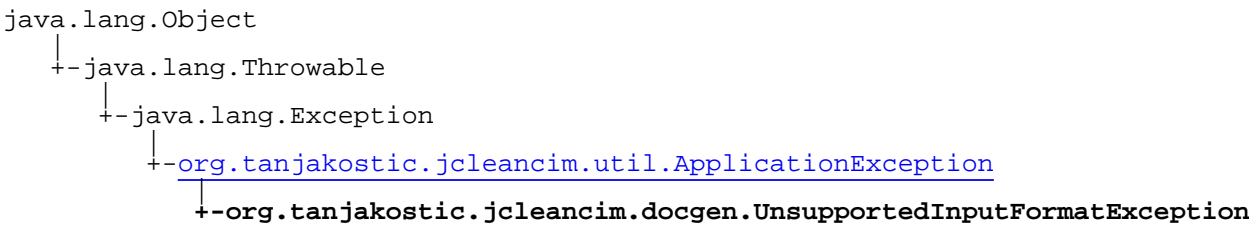
Returns the suffix, together with .xmi extension, as used for file name.

Package

org.tanjakostic.jcleancim.docgen

org.tanjakostic.jcleancim.docgen

Class UnsupportedInputFormatException



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

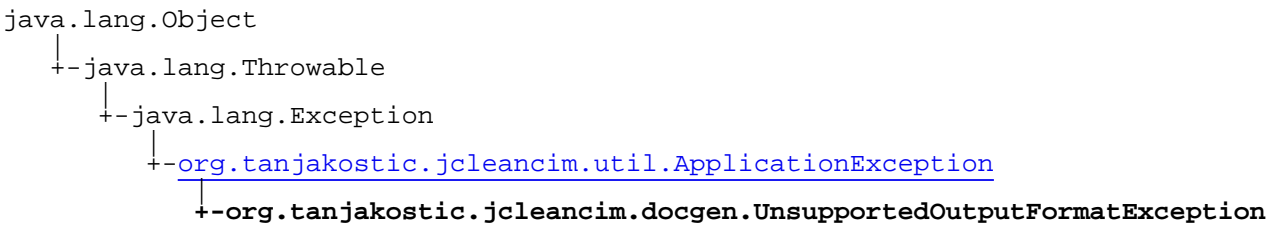
```
public UnsupportedInputFormatException(java.lang.String message)
```

UnsupportedInputFormatException

```
public UnsupportedInputFormatException(java.lang.Throwable cause)
```

org.tanjakostic.jcleancim.docgen

Class UnsupportedOutputFormatException



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 "Sci". 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	<code>isSpecial()</code> Returns whether this group is somehow special.
java.lang.String	<code>putCell(java.lang.String key, java.lang.String value)</code>
java.lang.String	<code>putCellNonEmpty(java.lang.String key, java.lang.String value)</code>

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

Interface ClassDoc

All Superinterfaces:

[ObjectDoc](#), [RawData](#)

public interface **ClassDoc**
extends [ObjectDoc](#)

Data required for documentation of classes.

Here the layout you may use:

```
getHeadingText()
getDescription()
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), alias, name Value: %s%s (%s)
public static final	CLASS_TITLE_PREFIX_FMT_CIM Value: (%s)
public static final	CLASS_TITLE_PREFIX_FMT_IEC61850 Value: <<%s>>
public static final	CONDITIONS_TEXT Text to append to description if conditions need to be explicitly printed. Value: Conditions:
public static final	LN_HEADING_FORMAT (deprecated, abstract), alias, nbsp, name Value: %sLN: %s%sName: %s
public static final	NO_ALIAS_HEADING_FORMAT (deprecated, abstract), name, (kindLabel) Value: %s%s%s
public static final	OTHER_WITH_ALIAS_HEADING_FORMAT (deprecated, abstract), alias, name, (stereotype) Value: %s%s (%s%s)

public static final	PRIM_DA_HEADING_FORMAT (deprecated, abstract), alias, name, kindLabel Value: <code>%s%s (%s %s)</code>
---------------------	--

Method Summary

PropertiesDoc	getAssocEndsDoc() Returns the documentation for all the association ends on the 'other' side of this class, suitable to be printed as a table.
PropertiesDoc	getAttributesDoc() Returns the documentation for all the attributes of this class, suitable to be printed as a table.
java.util.List	getDiagramDocs() Returns the documentation for all the diagrams of this class.
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](#)

[getDescription](#), [getDocgenCfg](#), [getHeadingText](#)

Methods inherited from interface [org.tanjakostic.jcleancim.docgen.collector.RawData](#)

[copyCell](#), [copyNonEmptyCell](#), [getCell](#), [getCells](#), [hasKey](#), [putCell](#), [putCellNonEmpty](#)

Fields

CLASS_TITLE_PREFIX_FMT_CIM

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

Constant value: `(%s)`

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), name, (kindLabel)
 Constant value: `%s%s%s`

LN_HEADING_FORMAT

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

(deprecated, abstract), alias, nbsp, name

(continued from last page)

Constant value: **%SLN: %S%SName: %S**

PRIM_DA_HEADING_FORMAT

```
public static final java.lang.String PRIM_DA_HEADING_FORMAT
```

(deprecated, abstract), 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), 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), alias, name, (stereotype)

Constant value: **%S%S (%S%S)**

CONDITIONS_TEXT

```
public static final java.lang.String CONDITIONS_TEXT
```

Text to append to description if conditions need to be explicitly printed.

Constant value: **Conditions:**

Methods

getAttributesDoc

```
public 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 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 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.

getDiagramDocs

```
public 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

java.util.List	getData()
java.lang.String	getEnd()
java.lang.String	getStart()
java.lang.String	toXml(boolean prettyPrint)

Methods

getStart

public java.lang.String **getStart()**

getData

public java.util.List **getData()**

getEnd

public java.lang.String **getEnd()**

toXml

public 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

java.util.List	_getSkippedInfPackageQNames() Returns qualified names of all the skipped packages.
boolean	addSkippedInformativePackage (java.lang.String qName) Must be called by every newly created PackageDoc that is to be skipped.
void	addToFlattened (PackageDoc packageDoc) Must be called by every newly created PackageDoc that need not be skipped.
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.
void	collect (UmlModel model) Collects recursively documentation from UML model packages, their sub-packages, etc.
DocgenConfig	getDocgenCfg() Returns configuration according to which the documentation is collected for generation.
FixedFormDocumentation	getFixedFormDocumentation() Returns documentation per package, indexed by nature.
FreeFormDocumentation	getFreeFormDocumentation() 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.
boolean	isFromUml() Returns whether this collector has been created from a UML model (as opposed to pure API calls).

Methods

collect

public void **collect**([UmlModel](#) model)

Collects recursively documentation from UML model packages, their sub-packages, etc.

Parameters:

(continued from last page)

model - UML model.

Throws:

`UnsupportedOperationException` - if this instance has not been created with an underlying UML model.

getFreeFormDocumentation

```
public FreeFormDocumentation getFreeFormDocumentation()
```

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 FixedFormDocumentation getFixedFormDocumentation()
```

Returns documentation per package, indexed by nature. This presentation is convenient for nature-dependent selection of packages.

addToFlattened

```
public void addToFlattened(PackageDoc packageDoc)
```

Must be called by every newly created [PackageDoc](#) that need not be skipped.

Parameters:

`packageDoc` - package documentation to retain.

addToScoped

```
public 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 boolean addSkippedInformativePackage(java.lang.String qName)
```

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.

_getSkippedInfPackageQNames

```
public java.util.List _getSkippedInfPackageQNames()
```

Returns qualified names of all the skipped packages.

getDocgenCfg

```
public DocgenConfig getDocgenCfg()
```

(continued from last page)

Returns configuration according to which the documentation is collected for generation.

isFromUml

```
public 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	includeInhFromMetamodel
public final	includeNonPublic
public final	keepHtml
public final	lnMapPckName
public final	lnPckNames
public final	owners
public final	presCondPckName
public final	trgOpPckName

public final

[writeUmlTypes](#)

Constructor Summary

public	DocgenConfig (boolean printHtml, boolean collectForWord, boolean includeInf, boolean includeNonPublic) Constructor for CIM-like simple printing; for XML all CIM owners are included.
public	DocgenConfig (boolean printHtml, boolean collectForWord, boolean includeInf, boolean includeNonPublic, 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

keepHtml

```
public final boolean keepHtml
```

includeInf

```
public final boolean includeInf
```

includeNonPublic

```
public final boolean includeNonPublic
```

owners

```
public final java.util.EnumSet owners
```

includeInhFromMetamodel

```
public final boolean includeInhFromMetamodel
```

(continued from last page)

writeUmlTypes

```
public final boolean writeUmlTypes
```

docgenPckNames

```
public final java.util.Collection docgenPckNames
```

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
```

(continued from last page)

InPckNames

```
public final java.util.Collection InPckNames
```

cdcPckNames

```
public final java.util.Collection cdcPckNames
```

daPckNames

```
public final java.util.Collection daPckNames
```

basicPckNames

```
public final java.util.Collection basicPckNames
```

Constructors

DocgenConfig

```
public DocgenConfig(boolean printHtml,  
                    boolean collectForWord,  
                    boolean includeInf,  
                    boolean includeNonPublic)
```

Constructor for CIM-like simple printing; for XML all CIM owners are included.

Parameters:

```
printHtml  
includeInf  
includeNonPublic
```

DocgenConfig

```
public DocgenConfig(boolean printHtml,  
                    boolean collectForWord,  
                    boolean includeInf,  
                    boolean includeNonPublic,  
                    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)
```


(continued from last page)

Constructor for IEC61850 printing.

DocgenConfig

```
public DocgenConfig(Config cfg)
```

Constructor.

Parameters:

cfg

org.tanjakostic.jcleancim.docgen.collector Class DocIDHelper

java.lang.Object

└--org.tanjakostic.jcleancim.docgen.collector.DocIDHelper

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

FIXME: Have to somehow deal with checks on uniqueness (without using UUIDs)...

Method Summary

static java.lang.String	createDocId (java.lang.String prefix, java.lang.String ending)
static java.lang.String	createDocId (UmlObject o, java.lang.String ending) FIXME: We currently use internal EA identifier to avoid name clashes...

Methods inherited from class java.lang.Object

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

Methods

createDocId

public static java.lang.String **createDocId**([UmlObject](#) o,
java.lang.String ending)

FIXME: We currently use internal EA identifier to avoid name clashes...

createDocId

public static java.lang.String **createDocId**(java.lang.String prefix,
java.lang.String ending)

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

AGSpec	getAttrGroupSpec() Returns non-null instance if this is EntryDoc.Kind.groupSubhead , null otherwise.
FormatInfo	getFormatInfo() Returns format information about the formattable cell at index <i>j</i> , null if this entry has no formattable cell, or if the formattable cell is actually not formatted.
EntryDoc.Kind	getKind() Returns the kind of this entry; useful for formatting.
java.lang.String[]	getValues() Returns array of values for columns (i.e., row content).
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 java.lang.String[] getValues()
```

Returns array of values for columns (i.e., row content).

getKind

```
public EntryDoc.Kind getKind()
```

Returns the kind of this entry; useful for formatting.

getAttrGroupSpec

```
public AGSpec getAttrGroupSpec()
```

Returns non-null instance if this is [EntryDoc.Kind.groupSubhead](#), null otherwise.

getFormatInfo

```
public 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.

toCsv

```
public 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 shading 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 shading 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 shading 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.

Method Summary

java.lang.String	getCaptionText() Returns text that describes the caption.
java.io.File	getFigureFile() Returns file with the figure.
java.lang.String	getIntroText() Returns text that will introduce the table.

Methods inherited from interface [org.tanjakostic.jcleancim.docgen.collector.ObjectDoc](#)

[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 java.lang.String getIntroText()
```

Returns text that will introduce the table. E.g. "Figure 23 {introText}".

getFigureFile

```
public java.io.File getFigureFile()
```

Returns file with the figure.

getCaptionText

```
public 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 limited 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 name (like for MS Word documentation). The scope of retained packages is limited according to configuration.

Constructor Summary

public	FreeFormDocumentation (ModelFinder modelFinder, java.util.Map packageDocs) Constructor.
--------	---

Method Summary

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,
java.util.Map packageDocs)

Constructor.

Parameters:

modelFinder - model facade; if null, most of placeholders will have error.

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.

Methods

(continued from last page)

getModelFinder

```
public ModelFinder getModelFinder()
```

getPackageDocs

```
public java.util.Map getPackageDocs()
```

getPackageDoc

```
public PackageDoc getPackageDoc(java.lang.String name)
```

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

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

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.
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.
java.lang.String	findQualifiedClassName (java.lang.String packageName, java.lang.String className, java.lang.String separator) Returns the packageNameseparatorclassName of the first class and package combination when found, null otherwise.

Methods

findAttributeValue

```
public 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 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.

findQualifiedClassName

```
public java.lang.String findQualifiedClassName(java.lang.String packageName,
      java.lang.String className,
      java.lang.String separator)
```

Returns the packageNameseparatorclassName of the first class and package combination 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.

Method Summary

TextDescription	getDescription() Returns description of this model element.
DocgenConfig	getDocgenCfg() Returns document generation specific configuration.
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](#)

Methods

getDocgenCfg

public [DocgenConfig](#) **getDocgenCfg()**

Returns document generation specific configuration.

getHeadingText

public java.lang.String **getHeadingText()**

Returns text to be used for chapter heading for this model element; may be empty.

getDescription

public [TextDescription](#) **getDescription()**

Returns description of this model element.

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 [Placeholder.Kind.PACKAGE](#):

```
getHeadingText()
  getGenHeadingText()
    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 [Placeholder.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 [Placeholder.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 [Placeholder.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 [Placeholder.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 [Placeholder.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 [Placeholder.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 mappings in place of [Placeholder.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
---------------------	---

Method Summary

PropertiesDoc	getAbbrPackageDoc()
java.util.List	getChildPackageDocs() Returns documentation for all the child packages of this package.
java.util.List	getClassDocs() Returns documentation for all the classes in this package.
PropertiesDoc	getDataIndexDoc()
PackageScl	getEnumsPackageScl()
PropertiesDoc	getFcPackageDoc()
java.util.List	getFigureDocs() Returns documentation for all the figures in this package.
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.
PropertiesDoc	getLnMapPackageDoc()
java.lang.String	getModelName() Returns name of the model package (the one with nature) to which it belongs.
NamespaceInfo	getNamespaceInfo() Returns name space information if it is defined for the package, null otherwise.
Nature	getNature() Returns nature of the package.
java.lang.String	getPackageName() Returns the name of the package.
PropertiesDoc	getPresCondPackageDoc()
PropertiesDoc	getTrgOpPackageDoc()

Methods inherited from interface [org.tanjakostic.jcleancim.docgen.collector.ObjectDoc](#)

[getDescription](#), [getDocgenCfg](#), [getHeadingText](#)

Methods inherited from interface [org.tanjakostic.jcleancim.docgen.collector.RawData](#)

[copyCell](#), [copyNonEmptyCell](#), [getCell](#), [getCells](#), [hasKey](#), [putCell](#), [putCellNonEmpty](#)

(continued from last page)

Fields

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**

Methods

getPackageName

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

Returns name of the model package (the one with nature) to which it belongs.

getNature

```
public Nature getNature()
```

Returns nature of the package.

getGenHeadingText

```
public 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.

getFigureDocs

```
public java.util.List getFigureDocs()
```

Returns documentation for all the figures in this package.

getClassDocs

```
public java.util.List getClassDocs()
```

Returns documentation for all the classes in this package.

(continued from last page)

getChildPackageDocs

```
public java.util.List getChildPackageDocs()
```

Returns documentation for all the child packages of this package.

getDataIndexDoc

```
public PropertiesDoc getDataIndexDoc()
```

getLnMapPackageDoc

```
public PropertiesDoc getLnMapPackageDoc()
```

getPresCondPackageDoc

```
public PropertiesDoc getPresCondPackageDoc()
```

getFcPackageDoc

```
public PropertiesDoc getFcPackageDoc()
```

getTrgOpPackageDoc

```
public PropertiesDoc getTrgOpPackageDoc()
```

getAbbrPackageDoc

```
public PropertiesDoc getAbbrPackageDoc()
```

getEnumsPackageScl

```
public 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

java.util.List	getClassScls() Class as XML strings.
java.lang.String	getHeadingText() Returns text that can be used as heading of a chapter.
boolean	notEmpty() Use this one to see whether to create XML for this package at all.
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 boolean **notEmpty()**

(continued from last page)

Use this one to see whether to create XML for this package at all.

getHeadingText

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

Returns text that can be used as heading of a chapter.

getClassScIs

```
public java.util.List getClassScIs()
```

Class as XML strings.

toXml

```
public java.lang.String toXml(boolean prettyPrint)
```

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 :
---------------------	--

Method Summary

java.lang.String	getCaptionText() Returns text that describes the caption (e.g., "Table 23 - [captionText]").
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.
int	getColumnCount() Returns number of columns for the table; must be same as TableSpec.colCount() from the columns spec returned by getTableSpec() .
java.util.List	getDataEntryDocs() Returns (unmodifiable list of) data entries only.

<code>java.util.List</code>	<code>getEntryDocs()</code> Returns (unmodifiable list of) all entries.
<code>TextDescription.TextKind[]</code>	<code>getFormats()</code> Returns array of applied formattings for the formattable cells in every row i.
<code>int</code>	<code>getHeadingEntriesCount()</code> Returns the number of header entries (<code>EntryDoc.Kind.tableName</code> or <code>EntryDoc.Kind.columnLabels</code>) in this instance.
<code>java.lang.String</code>	<code>getIntroText()</code> Returns text that will introduce the table (e.g.
<code>int</code>	<code>getRowCount()</code> Returns number of rows for the table.
<code>EntryDoc.Kind[]</code>	<code>getRowKinds()</code> Returns kinds for rows from <code>getCellValues()</code> .
<code>java.lang.String</code>	<code>getTableName()</code> Returns the name of the table; this will be the first heading row, when non-null.
<code>TableSpec</code>	<code>getTableSpec()</code> Returns columns specification.
<code>boolean</code>	<code>notEmpty()</code> Returns whether there are any non-heading entries (<code>EntryDoc.Kind.groupSubhead</code> or <code>EntryDoc.Kind.data</code>) in this instance.

Methods inherited from interface [org.tanjakostic.jcleancim.docgen.collector.ObjectDoc](#)

[`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

notEmpty

```
public 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.

(continued from last page)

getIntroText

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

Returns text that will introduce the table (e.g. "Table 23 [**introText**]").

getCaptionText

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

Returns text that describes the caption (e.g., "Table 23 - [**captionText**]").

getHeadingEntriesCount

```
public int getHeadingEntriesCount()
```

Returns the number of header entries ([EntryDoc.Kind.tableName](#) or [EntryDoc.Kind.columnLabels](#)) in this instance.

getEntryDocs

```
public java.util.List getEntryDocs()
```

Returns (unmodifiable list of) all entries.

getDataEntryDocs

```
public java.util.List getDataEntryDocs()
```

Returns (unmodifiable list of) data entries only.

getTableName

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

Returns the name of the table; this will be the first heading row, when non-null.

getTableSpec

```
public 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 int getRowCount()
```

Returns number of rows for the table.

getColumnCount

```
public int getColumnCount()
```

Returns number of columns for the table; must be same as [TableSpec.colCount\(\)](#) from the columns spec returned by [getTableSpec\(\)](#).

(continued from last page)

getCellValues

```
public 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 EntryDoc.Kind\[\] getRowKinds()
```

Returns kinds for rows from [getCellValues\(\)](#).

getFormats

```
public 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.

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

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.
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.
java.lang.String	getCell (java.lang.String key) Returns value for key, null if key does not exist or if key is null.
java.util.Map	getCells () Returns potentially empty map of key/value pairs.
boolean	hasKey (java.lang.String key) If true, key is present.
java.lang.String	putCell (java.lang.String key, java.lang.String value) Adds the value for key.
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 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 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 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 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 boolean hasKey(java.lang.String key)
```

If true, key is present.

getCells

```
public java.util.Map getCells()
```

Returns potentially empty map of key/value pairs.

getCell

```
public 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_aliasID Value: aliasID
public static final	A_cdcId Value: cdcId
public static final	A_deducedTypeText Value: deducedTypeText
public static final	A_descID Value: descID
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_kind Value: kind
public static final	A_literalVal Value: literalVal
public static final	A_lns Value: lns
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_rsName Value: rsName
public static final	A_signature Value: signature
public static final	A_text Value: text
public static final	A_textID Value: textID
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_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_CDCAAttributesTable Value: CDCAAttributesTable

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_aliasID

```
public static final java.lang.String A_aliasID
```

Constant value: **aliasID**

A_descID

```
public static final java.lang.String A_descID
```

Constant value: **descID**

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**

(continued from last page)

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_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**

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_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
```

(continued from last page)

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**

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**

E_AbbreviationsTable

```
public static final java.lang.String E_AbbreviationsTable
```

Constant value: **AbbreviationsTable**

(continued from last page)

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
```

Constant value: **LiteralsTable**

E_OperationsTable

```
public static final java.lang.String E_OperationsTable
```

(continued from last page)

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**

CAT_doNull

```
public static final java.lang.String CAT_doNull
```

Constant value: **doNull**

(continued from last page)

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
```

Constant value: **instTag**

LOC_isSpecial

```
public static final java.lang.String LOC_isSpecial
```

(continued from last page)

Constant value: **isspecial**

Package

**org.tanjakostic.jcleancim.docgen.collecto
r.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.

Constructor Summary

protected	AbstractObjectDoc (DocgenConfig docgenCfg, UmlObject o, java.lang.String what, TextDescription description, TextDescription htmlDescription, boolean ignoreDesc, java.lang.String headingText) "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)
java.lang.String	createDocId (UmlObject o, java.lang.String ending)
void	filterClasses (UmlPackage p, java.util.Collection retainedNatives)
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	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](#)

[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)
```

"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).

(continued from last page)

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.

Methods

log

```
protected static void log(org.apache.log4j.Logger logger,  
    java.lang.String message)
```

createDocId

```
protected final java.lang.String createDocId(UmlObject o,  
    java.lang.String ending)
```

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.

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

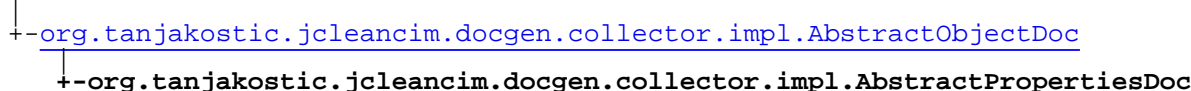
```
public final java.util.Map getCells()
```

getCell

```
public final java.lang.String getCell(java.lang.String key)
```

org.tanjakostic.jcleancim.docgen.collector.impl Class AbstractPropertiesDoc

java.lang.Object



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.PropertiesDoc](#)

[INHERITED_FROM](#)

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) 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) Same as AbstractPropertiesDoc(DocgenConfig, UmlObject, String, String, String, TableSpec, String) , 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	getCaptionText ()
java.lang.String[][]	getCellValues ()
int	getColumnCount ()
java.util.List	getDataEntryDocs ()
static java.lang.String	getDeprecatedTextAsPrefix (UmlObject o) Returns empty string if attribute is not deprecated, otherwise formatted text ending with white space.
java.util.List	getEntryDocs ()
TextDescription.TextKind ind[]	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](#), [filterClasses](#), [getCell](#), [getCells](#), [getDescription](#), [getDocgenCfg](#), [getHeadingText](#), [hasKey](#), [log](#), [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](#)

[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](#)

[getCaptionText](#), [getCellValues](#), [getColumnCount](#), [getDataEntryDocs](#), [getEntryDocs](#), [getFormats](#), [getHeadingEntriesCount](#), [getIntroText](#), [getRowCount](#), [getRowKinds](#), [getTableName](#), [getTableSpec](#), [notEmpty](#)

Methods inherited from interface [org.tanjakostic.jcleancim.docgen.collector.ObjectDoc](#)

[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)
```

(continued from last page)

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
```

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)
```

Same as [AbstractPropertiesDoc\(DocgenConfig, UmlObject, String, String, String, TableSpec, String\)](#), 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
ignoreDesc
headingText
introText
captionText
colSpec
tableName
```

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
```

(continued from last page)

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.

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)
```

(continued from last page)

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 attribute 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()
```

(continued from last page)

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()
```

getRowKinds

```
public final EntryDoc.Kind\[\] getRowKinds()
```

getFormats

```
public TextDescription.TextKind\[\] getFormats()
```

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	AbstractPropertiesDoc.CellText (TextDescription text) Creates an instance with null formatting info (all formatting will be ignored).
protected	AbstractPropertiesDoc.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

AbstractPropertiesDoc.CellText

protected **AbstractPropertiesDoc.CellText**([TextDescription](#) text)

(continued from last page)

Creates an instance with null formatting info (all formatting will be ignored).

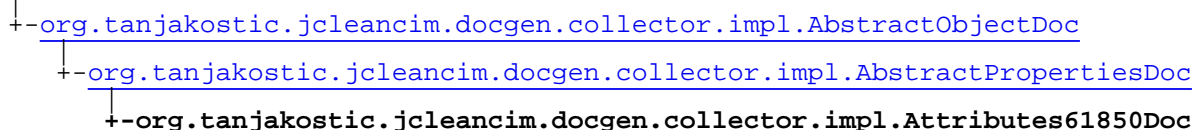
AbstractPropertiesDoc.CellText

```
protected AbstractPropertiesDoc.CellText( java.lang.Integer fmtIdx,  
                                         TextDescription textDesc)
```

Constructor.

org.tanjakostic.jcleancim.docgen.collector.impl Class Attributes61850Doc

java.lang.Object



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.PropertiesDoc](#)

[INHERITED_FROM](#)

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) Constructor.
-----------	--

Method Summary

void	deduceTypeText (UmlClass mmType, RawData outRawData, boolean showToENUM, boolean trimPrimitivePrefix, boolean all) Fills outRawData with value for org.tanjakostic.jcleancim.docgen.collector.WAX#A_deducedTypeText; and org.tanjakostic.jcleancim.docgen.collector.WAX#A_type and org.tanjakostic.jcleancim.docgen.collector.WAX#A_typeKind keys if enabled (all=true) and where they exist.
------	--

static void	fillPresenceConditionAndArgs (RawData entry, PresenceCondition pc, boolean isInherited, java.lang.String context) Fills appropriately raw data 'presCond', 'presCondArgs', 'presCondArgsID' and 'cond'.
static void	fillPresenceConditionAndArgs (RawData entry, PresenceCondition pc, boolean isInherited, java.lang.String context, boolean isDerivedStats) Fills appropriately raw data 'presCond', 'presCondArgs', 'presCondArgsID' and 'cond' if isDerivedStats=false, otherwise 'dsPresCond', 'dsPresCondArgs', 'dsPresCondArgsID' and 'dsCond'.
java.util.Collection	filterGroups (java.util.Collection groups) Returns non-empty groups, retained from groups after applying the filters set in configuration.

Methods inherited from class[org.tanjakostic.jcleancim.docgen.collector.impl.AbstractPropertiesDoc](#)

[addEntry](#), [deduceCellText](#), [deduceCellText](#), [filterAssociationEnds](#), [filterAttributes](#), [filterOperations](#), [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](#), [filterClasses](#), [getCell](#), [getCells](#), [getDescription](#), [getDocgenCfg](#), [getHeadingText](#), [hasKey](#), [log](#), [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](#)

[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](#)

[getCaptionText](#), [getCellValues](#), [getColumnCount](#), [getDataEntryDocs](#), [getEntryDocs](#), [getFormats](#), [getHeadingEntriesCount](#), [getIntroText](#), [getRowCount](#), [getRowKinds](#), [getTableName](#), [getTableSpec](#), [notEmpty](#)

Methods inherited from interface [org.tanjakostic.jcleancim.docgen.collector.ObjectDoc](#)

[getDescription](#), [getDocgenCfg](#), [getHeadingText](#)

Methods inherited from interface [org.tanjakostic.jcleancim.docgen.collector.RawData](#)

[copyCell](#), [copyNonEmptyCell](#), [getCell](#), [getCells](#), [hasKey](#), [putCell](#), [putCellNonEmpty](#)

(continued from last page)

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)
```

Constructor.

Methods

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 void deduceTypeText(UmlClass mmType,
                              RawData outRawData,
                              boolean showToENUM,
                              boolean trimPrimitivePrefix,
                              boolean all)
```

Fills outRawData with value for org.tanjakostic.jcleancim.docgen.collector.WAX#A_deducedTypeText; and org.tanjakostic.jcleancim.docgen.collector.WAX#A_type and org.tanjakostic.jcleancim.docgen.collector.WAX#A_typeKind keys if enabled (all=true) and where they exist.

Parameters:

mmType

(continued from last page)

outRawData - in/out argument, filled here.

showTorENUM - set this one to true when writing index for LNs and the attribute is a transient CDC; or when printing actual CDC tables and the attribute is some kind of enum.

trimPrimitivePrefix - set this one to true when dealing with an FCDA as type, for both CDC tables and CDC attributes index - because we search for actual underlying types for e.g. enums and composed DAs, for primitive/structured DAs it's the "real" primitive type that gets picked (with UML#PREF_P_/UML#PREF_S_), and we don't want to display that one in 7-3.

all - if true, will set all potential fields; otherwise, will set only

org.tanjakostic.jcleancim.docgen.collector.WAX#A_deducedTypeText.

fillPresenceConditionAndArgs

```
protected static void fillPresenceConditionAndArgs(RawData entry,  
    PresenceCondition pc,  
    boolean isInherited,  
    java.lang.String context)
```

Fills appropriately raw data 'presCond', 'presCondArgs', 'presCondArgsID' and 'cond'.

fillPresenceConditionAndArgs

```
protected static void fillPresenceConditionAndArgs(RawData entry,  
    PresenceCondition pc,  
    boolean isInherited,  
    java.lang.String context,  
    boolean isDerivedStats)
```

Fills appropriately raw data 'presCond', 'presCondArgs', 'presCondArgsID' and 'cond' if isDerivedStats=false, otherwise 'dsPresCond', 'dsPresCondArgs', 'dsPresCondArgsID' and '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

java.util.List	_getSkippedInfPackageQNames ()
boolean	addSkippedInformativePackage (java.lang.String qName)
void	addToFlattened (PackageDoc packageDoc)
void	addToScoped (PackageDoc packageDoc)
void	collect (UmlModel model)
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](#)

[_getSkippedInfPackageQNames](#), [addSkippedInformativePackage](#), [addToFlattened](#), [addToScoped](#), [collect](#), [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)

addToScoped

```
public void addToScoped(PackageDoc packageDoc)
```

addSkippedInformativePackage

```
public boolean addSkippedInformativePackage(java.lang.String qName)
```

getDocgenCfg

```
public DocgenConfig getDocgenCfg()
```

_getSkippedInfPackageQNames

```
public java.util.List _getSkippedInfPackageQNames()
```

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 (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[] values) Creates regular entry with data without any formatting (also, no new line character).
AGSpec	getAttrGroupSpec ()
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](#), [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.

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[] values)  
    throws java.lang.IllegalArgumentException
```

Creates regular entry with data without any formatting (also, no new line character).

Parameters:

values - non-empty array of non-null values.

Throws:

IllegalArgumentException - if values is empty.

createData

```
public static EntryDocImpl createData(FormatInfo formatInfo,  
        java.lang.String[] values)  
    throws java.lang.IllegalArgumentException
```

Creates regular entry with data.

Parameters:

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()
```

getValues

```
public final java.lang.String[] getValues()
```

(continued from last page)

getKind

```
public EntryDoc.Kind getKind()
```

getAttrGroupSpec

```
public AGSpec getAttrGroupSpec()
```

getFormatInfo

```
public final FormatInfo getFormatInfo()
```

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)
```

putCellNonEmpty

```
public java.lang.String putCellNonEmpty(java.lang.String key,  
                                         java.lang.String value)
```

hasKey

```
public boolean hasKey(java.lang.String key)
```

(continued from last page)

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.FigureDoc](#)

[CAPTION_TEXT_FORMAT](#), [INTRO_TEXT_FORMAT](#)

Constructor Summary

public	FigureDocImpl (DocgenConfig docgenCfg, UmlDiagram d) Constructor.
public	FigureDocImpl (DocgenConfig docgenCfg, UmlDiagram d, TextDescription description, TextDescription htmlDescription, java.io.File pic, java.lang.String showsWhat, java.lang.String caption) 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](#), [filterClasses](#), [getCell](#), [getCells](#), [getDescription](#), [getDocgenCfg](#), [getHeadingText](#), [hasKey](#), [log](#), [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](#)

[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](#)

[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)
```

Constructor.

Parameters:

docgenCfg
d

FigureDocImpl

```
public FigureDocImpl(DocgenConfig docgenCfg,
                    UmlDiagram d,
                    TextDescription description,
                    TextDescription htmlDescription,
                    java.io.File pic,
                    java.lang.String showsWhat,
                    java.lang.String caption)
```

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.

Methods

getIntroText

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

(continued from last page)

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 retrieve 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.io.File	findDiagramFile (java.lang.String containerName, java.lang.String diagramName)
java.lang.String	findQualifiedClassName (java.lang.String packageName, java.lang.String className, java.lang.String separator)

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](#), [findDiagramFile](#), [findQualifiedClassName](#)

Constructors

ModelFinderImpl

```
public ModelFinderImpl(UmlModel model)
```

Constructor.

(continued from last page)

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)
```

findQualifiedClassName

```
public java.lang.String findQualifiedClassName(java.lang.String packageName,  
                                                java.lang.String className,  
                                                java.lang.String separator)
```

Package

**org.tanjakostic.jcleancim.docgen.collecto
r.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.

Nested Class Summary

class	Caption.CaptionKind Caption.CaptionKind
-------	--

Constructor Summary

public	Caption (Caption.CaptionKind kind, Range range) Constructor.
--------	--

Method Summary

Caption.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**([Caption.CaptionKind](#) kind, [Range](#) range)

Constructor.

Parameters:
kind

(continued from last page)

Methods

getKind

```
public final Caption.CaptionKind getKind()
```

getRange

```
public Range getRange()
```

getStyle

```
public Style getStyle()
```

toString

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

org.tanjakostic.jcleancim.docgen.writer Class **Caption.CaptionKind**

java.lang.Object

└- java.lang.Enum

└- org.tanjakostic.jcleancim.docgen.writer.Caption.CaptionKind

All Implemented Interfaces:

java.io.Serializable, java.lang.Comparable

public static final class **Caption.CaptionKind**
extends java.lang.Enum

Field Summary

public static final	Figure
public static final	Table

Method Summary

Style	getStyle()
static boolean	looksLikeFigureCaption (java.lang.String styleName, java.lang.String text)
static boolean	looksLikeTableCaption (java.lang.String styleName, java.lang.String text)
static Caption.CaptionKind	valueOf (java.lang.String name)
static Caption.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.Caption.CaptionKind Figure
```

Table

```
public static final org.tanjakostic.jcleancim.docgen.writer.Caption.CaptionKind Table
```

Methods

values

```
public static Caption.CaptionKind\[\] values()
```

valueOf

```
public static Caption.CaptionKind valueOf(java.lang.String name)
```

getStyle

```
public Style getStyle()
```

looksLikeTableCaption

```
public static boolean looksLikeTableCaption(java.lang.String styleName,  
                                             java.lang.String text)
```

looksLikeFigureCaption

```
public static boolean looksLikeFigureCaption(java.lang.String styleName,  
                                              java.lang.String text)
```

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

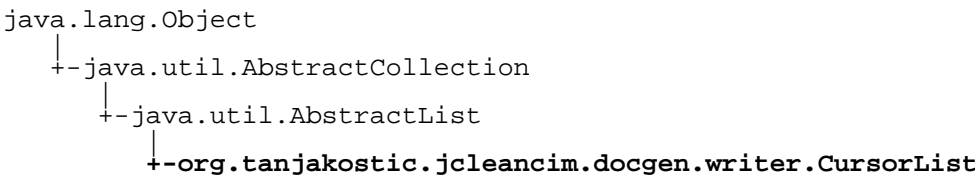
```
public Range getRange()
```

toString

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


org.tanjakostic.jcleancim.docgen.writer

Class CursorList



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(Caption.CaptionKind kind, Cursor cursor) Returns the number to be used in caption for the cursor; must be called.
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`, `remove`, `removeAll`, `retainAll`, `size`, `toArray`, `toArray`

Methods inherited from interface `java.lang.Iterable`

`iterator`

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`, `retainAll`, `set`, `size`, `subList`, `toArray`, `toArray`

Methods inherited from interface `java.util.Collection`

`add`, `addAll`, `clear`, `contains`, `containsAll`, `equals`, `hashCode`, `isEmpty`, `iterator`, `remove`, `removeAll`, `retainAll`, `size`, `toArray`, `toArray`

Methods inherited from interface `java.lang.Iterable`

`iterator`

Constructors

CursorList

```
public CursorList()
```

Methods

updateRanges

```
public CursorList updateRanges(CursorList cursors,  
    Range range)
```

snapshotIndexes

```
public java.util.List snapshotIndexes()
```

(continued from last page)

getReplacementFailures

```
public java.util.List getReplacementFailures()
```

captionAdded

```
public int captionAdded(Caption.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)
```

toString

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

org.tanjakostic.jcleancim.docgen.writer Class Placeholder

```
java.lang.Object
```

```
└--org.tanjakostic.jcleancim.docgen.writer.Placeholder
```

```
public class Placeholder
extends java.lang.Object
```

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
startUmlAttribute.{className}.{attributeName}.endUml
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
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 enclosed 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 [Placeholder.Kind.UNSUPPORTED](#) always returns non-null error that you may want to use to replace the placeholder to indicate failure. Instances of kind [Placeholder.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 ([Placeholder.Kind.PRES_CONDITIONS](#), [Placeholder.Kind.ABBREVIATIONS](#), [Placeholder.Kind.SCL_ENUMS](#), [Placeholder.Kind.PACKAGE](#), [Placeholder.Kind.LNMAP_PACKAGE](#), [Placeholder.Kind.DATA_INDEX](#), [Placeholder.Kind.FCS](#), [Placeholder.Kind.TRGOPS](#)) or
- both tokens ([Placeholder.Kind.ATTRIBUTE](#), [Placeholder.Kind.DIAGRAM](#)).

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:

- [Placeholder.Kind.PACKAGE](#),
- [Placeholder.Kind.PRES_CONDITIONS](#),
- [Placeholder.Kind.FCS](#),
- [Placeholder.Kind.TRGOPS](#),
- [Placeholder.Kind.SCL_ENUMS](#),
- [Placeholder.Kind.LNMAP_PACKAGE](#), and,
- [Placeholder.Kind.DATA_INDEX](#).

Nested Class Summary

class	Placeholder.Kind Placeholder.Kind
-------	--

Field Summary

public static final	MS_PATTERN
public static final	PH_LETTERS Value: [ADFLTPS]

Constructor Summary

public	Placeholder (java.lang.String text)
public	Placeholder (java.lang.String text, 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.
java.lang.String	getErrorText () Returns null if there are no parsing errors, the error message otherwise.
int	getFigureCount () Returns the last figure caption index by the current end of the range.
int	getFigureCountBefore () Returns number of figures (i.e., found captions starting with "Figure") before this placeholder.
java.lang.String	getFirstToken ()
Placeholder.Kind	getKind ()
java.lang.String	getReplacedText () Returns the replacement text, as set by the user with setReplacedText(String) .

java.lang.String	getSecondToken()
static java.util.List	getSupportedFormats()
int	getTableCount() Returns the last table caption index by the current end of the range.
int	getTableCountBefore() Returns number of tables (i.e., found captions starting with "Table") before this placeholder.
java.lang.String	getText() Returns the placeholder text.
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()
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

PH_LETTERS

```
public static final java.lang.String PH_LETTERS
```

Constant value: **[ADFLTPS]**

MS_PATTERN

```
public static final java.lang.String MS_PATTERN
```

Constructors

Placeholder

```
public Placeholder(java.lang.String text)
```

Placeholder

```
public Placeholder(java.lang.String text,
                  int figureCountBefore,
                  int tableCountBefore)
```

Methods

getSupportedFormats

```
public static java.util.List getSupportedFormats()
```

getText

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

Returns the placeholder text.

getKind

```
public Placeholder.Kind getKind()
```

getFirstToken

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

getSecondToken

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

getErrorMessage

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

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).

setReplacedText

```
public void setReplacedText(java.lang.String replacedText)
```

(continued from last page)

Use this setter to track progress and status of placeholder replacement.

getFigureCountBefore

```
public int getFigureCountBefore()
```

Returns number of figures (i.e., found captions starting with "Figure") before this placeholder.

getTableCountBefore

```
public int getTableCountBefore()
```

Returns number of tables (i.e., found captions starting with "Table") before this placeholder.

getReplacedText

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

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.

(continued from last page)

toString

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

org.tanjakostic.jcleancim.docgen.writer Class Placeholder.Kind

java.lang.Object

└- java.lang.Enum

└- org.tanjakostic.jcleancim.docgen.writer.Placeholder.Kind

All Implemented Interfaces:

java.io.Serializable, java.lang.Comparable

public static final class **Placeholder.Kind**
extends java.lang.Enum

Field Summary

public static final	ABBREVIATIONS
public static final	ATTRIBUTE
public static final	DATA_INDEX
public static final	DIAGRAM
public static final	FCS
public static final	FILE
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

Method Summary

static boolean	isForHeading (Placeholder.Kind kind) Returns whether kind is expected to be in the heading paragraph.
static Placeholder.Kind	valueOf (java.lang.String name)

<code>static Placeholder.Kind[]</code>	<code>values()</code>
--	---------------------------------------

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.writer.Placeholder.Kind FILE`

DIAGRAM

`public static final org.tanjakostic.jcleancim.docgen.writer.Placeholder.Kind DIAGRAM`

ATTRIBUTE

`public static final org.tanjakostic.jcleancim.docgen.writer.Placeholder.Kind ATTRIBUTE`

PRES_CONDITIONS

`public static final org.tanjakostic.jcleancim.docgen.writer.Placeholder.Kind
PRES_CONDITIONS`

FCS

`public static final org.tanjakostic.jcleancim.docgen.writer.Placeholder.Kind FCS`

TRGOPS

`public static final org.tanjakostic.jcleancim.docgen.writer.Placeholder.Kind TRGOPS`

ABBREVIATIONS

```
public static final org.tanjakostic.jcleancim.docgen.writer.Placeholder.Kind  
ABBREVIATIONS
```

SCL_ENUMS

```
public static final org.tanjakostic.jcleancim.docgen.writer.Placeholder.Kind SCL_ENUMS
```

PACKAGE

```
public static final org.tanjakostic.jcleancim.docgen.writer.Placeholder.Kind PACKAGE
```

DATA_INDEX

```
public static final org.tanjakostic.jcleancim.docgen.writer.Placeholder.Kind  
DATA_INDEX
```

LNMAP_PACKAGE

```
public static final org.tanjakostic.jcleancim.docgen.writer.Placeholder.Kind  
LNMAP_PACKAGE
```

UNSUPPORTED

```
public static final org.tanjakostic.jcleancim.docgen.writer.Placeholder.Kind  
UNSUPPORTED
```

Methods

values

```
public static Placeholder.Kind\[\] values()
```

valueOf

```
public static Placeholder.Kind valueOf(java.lang.String name)
```

isForHeading

```
public static boolean isForHeading(Placeholder.Kind kind)
```

(continued from last page)

Returns whether `kind` is expected to be in the heading paragraph.

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

int	getEnd()
java.lang.Object	getObject()
int	getStart()
java.lang.String	getText()
void	setEnd(int idx)
void	setStart(int idx)
void	setStartEnd(int idxStart, int idxEnd)
void	setText(java.lang.String newText)

Methods

getStart

```
public int getStart()
```

setStart

```
public void setStart(int idx)
```

getEnd

```
public int getEnd()
```

(continued from last page)

setEnd

```
public void setEnd(int idx)
```

setStartEnd

```
public void setStartEnd(int idxStart,  
                        int idxEnd)
```

getText

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

setText

```
public void setText(java.lang.String newText)
```

getObject

```
public 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.

FIXME: Application needs to initialise the name ([initialiseFromExisting\(Map\)](#)) so that [getName\(\)](#) returns desired style name if that style is available in the document, otherwise the default style name.

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 final	para
public static final	tabcapt
public static final	tabcell

<code>public static final</code>	<code>tabhead</code>
<code>public static final</code>	<code>TOC_PREFIX</code> Value: TOC

Method Summary

<code>java.util.List</code>	<code>getAllKnownIds()</code>
<code>java.util.List</code>	<code>getAllKnownNames()</code>
<code>java.lang.String</code>	<code>getBuiltinName()</code>
<code>java.lang.String</code>	<code>getBuiltinStyleId()</code> For .docx only.
<code>static Style</code>	<code>getHeadingStyle(int outlineLevel)</code>
<code>java.lang.String</code>	<code>getName()</code> Returns usable style name (both .doc and .docs).
<code>java.lang.String</code>	<code>getStyleId()</code> Returns style ID (only .docs).
<code>java.lang.String</code>	<code>getWantedName()</code>
<code>java.lang.String</code>	<code>getWantedStyleId()</code> For .docx only.
<code>static void</code>	<code>initialiseFromExisting(java.util.Map idPerName)</code> Initialises predefined enum literals with the actual data (e.g., read from an opened file).
<code>static boolean</code>	<code>isTOC(java.lang.String styleName)</code>
<code>static boolean</code>	<code>usableForFigureCaption(java.lang.String styleNameOrId)</code>
<code>static boolean</code>	<code>usableForTableCaption(java.lang.String styleNameOrId)</code>
<code>static Style</code>	<code>valueOf(java.lang.String name)</code>
<code>static Style[]</code>	<code>values()</code>

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
```

h1

```
public static final org.tanjakostic.jcleancim.docgen.writer.Style h1
```

h2

```
public static final org.tanjakostic.jcleancim.docgen.writer.Style h2
```

(continued from last page)

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
```

h8

```
public static final org.tanjakostic.jcleancim.docgen.writer.Style h8
```

h9

```
public static final org.tanjakostic.jcleancim.docgen.writer.Style h9
```

TOC_PREFIX

```
public static final java.lang.String TOC_PREFIX
```

Constant value: **TOC**

Methods

values

```
public static Style\[\] values()
```

(continued from last page)

valueOf

```
public static Style valueOf(java.lang.String name)
```

initialiseFromExisting

```
public static void initialiseFromExisting(java.util.Map idPerName)
```

Initialises predefined enum literals with the actual data (e.g., read from an opened file).

isTOC

```
public static boolean isTOC(java.lang.String styleName)
```

getHeadingStyle

```
public static Style getHeadingStyle(int outlineLevel)
```

usableForFigureCaption

```
public static boolean usableForFigureCaption(java.lang.String styleNameOrId)
```

usableForTableCaption

```
public static boolean usableForTableCaption(java.lang.String styleNameOrId)
```

getWantedName

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

getWantedStyleId

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

For .docx only.

getBuiltinName

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

getBuiltinStyleId

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

(continued from last page)

For .docx only.

getAllKnownNames

```
public java.util.List getAllKnownNames()
```

getAllKnownIds

```
public java.util.List getAllKnownIds()
```

getName

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

Returns usable style name (both .doc and .docs).

getStyleId

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

Returns style ID (only .docs).

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

java.util.Map	getDocumentMetadata() Returns (potentially empty) custom document properties.
WriterInput	getInput() Returns input used for writing.
java.lang.String	getInputFileNames() Return names of one or more input files used by this writer.
java.lang.String	getOutputFileNames() Return names of one or more output files created by this writer.
java.util.Set	getSupportedFormats() Returns the set of supported formats, as file extensions; e.g., ".doc", ".xml".
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 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 java.lang.String getInputFileNames()
```

Return names of one or more input files used by this writer.

getOutputFileNames

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

Return names of one or more output files created by this writer.

getSupportedFormats

```
public java.util.Set getSupportedFormats()
```

Returns the set of supported formats, as file extensions; e.g., ".doc", ".xml".

getDocumentMetadata

```
public 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 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; 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) You'll always use this one for regular text and tables.
void	clearUndoCache ()
Cursor	closeAndReopenDoc (CursorList cursors, Cursor currentCursor)
Caption	createCaption (Caption.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()
CursorList	initCursorsFromPlaceholders(java.util.List figCaptionRanges, java.util.List tabCaptionRanges)
void	initDocgenOptimisationOptions()
boolean	isInTOC(Range range)
static double	pointForPerc(int perc)
void	scanAndPostprocess(WordHelper.PostProcessor pp)
void	setDocgenOptimisationOptions()
java.lang.String	toString()
void	unsetDocgenOptimisationOptions()
void	write()
Cursor	writeAbbrTable(Cursor initCursor, PackageDoc packageDoc)
Cursor	writeClass(Cursor initCursor, ClassDoc doc)
Cursor	writeDataIndex(Cursor initCursor, PackageDoc packageDoc)
Cursor	writeDiagram(Cursor cursor, FigureDoc 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](#), [writeClass](#), [writeDataIndex](#), [writeDiagram](#), [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](#), [collectCaptions](#), [createCaption](#), [createCursor](#), [createPatternFinder](#), [createRange](#), [createWordApp](#), [duplicateRange](#), [exitAppAndSaveDocument](#), [getCursors](#), [getCustomDocProperties](#), [getDocumentAsRange](#), [getExistingStyleNames](#), [getRangeParagraphCount](#), [getRangeParagraphOutlineLevel](#), [getRangeParagraphStyleName](#), [getWordAppName](#), [getWordAppVersion](#), [initCursorsFromPlaceholders](#), [initDocgenOptimisationOptions](#), [insertFigure](#), [insertFigureCaption](#), [insertFigureRef](#), [insertTable](#), [insertTableCaption](#), [insertTableRef](#), [isInTOC](#), [isRangeWithTable](#), [openDoc](#), [prependNewLine](#), [prependText](#), [scanAndPostprocess](#), [setCustomDocProperties](#), [setDocgenOptimisationOptions](#), [unsetDocgenOptimisationOptions](#), [updateFields](#), [updateTablesOf](#)

Constructors

AbstractWordWriter

protected **AbstractWordWriter**([WordWriterInput](#) input)

Constructor; copies input template into output file that will be filled with this writer.

Parameters:

(continued from last page)

input

Methods

createWordApp

```
public void createWordApp()
```

This default implementation does nothing.

exitAppAndSaveDocument

```
public void exitAppAndSaveDocument()
```

This default implementation does nothing.

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.

pointForPerc

```
protected static double pointForPerc(int perc)
```

clearUndoCache

```
public void clearUndoCache()
```

This default implementation does nothing.

(continued from last page)

scanAndPostprocess

```
public final void scanAndPostprocess(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(Caption.CaptionKind figure,  
    Range range)
```

initCursorsFromPlaceholders

```
public CursorList initCursorsFromPlaceholders(java.util.List figCaptionRanges,  
    java.util.List tabCaptionRanges)
```

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.

(continued from last page)

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)
```

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)
```

(continued from last page)

writeClass

```
public Cursor writeClass(Cursor initCursor,  
    ClassDoc doc)
```

writeProperties

```
public Cursor writeProperties(Cursor initCursor,  
    PropertiesDoc doc)
```

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

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	appendTextInNewParagraphWithStyle (Range range, TextDescription newText, Style style)
java.lang.String	appendTextWithStyle (Range range, java.lang.String newText, Style style)
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."
Cursor	closeAndReopenDoc (CursorList cursors, Cursor currentCursor)

void	closeDoc() Closes and saves the MS Word document.
void	collapseRangeToEnd() (Range range)
java.util.Map	collectCaptions()
Caption	createCaption() (Caption.CaptionKind figure, Range range)
Cursor	createCursor() (Placeholder ph, Range limited)
WordPatternFinder	createPatternFinder()
Range	createRange() (java.lang.Object object)
void	createWordApp() Where applicable, launches (and caches) the MS Word application.
Range	duplicateRange() (Range range) FIXME: could go to Range?
void	exitAppAndSaveDocument() Saves MS Word document (and where applicable, exits MS Word application).
CursorList	getCursors()
java.util.Map	getCustomDocProperties()
Range	getDocumentAsRange()
java.util.Map	getExistingStyleNames() Returns names of existing styles.
int	getRangeParagraphCount() (Range range)
int	getRangeParagraphOutlineLevel() (Range range, int paraIdx)
java.lang.String	getRangeParagraphStyleName() (Range range, int paraIdx)
java.lang.String	getWordAppName() Returns the MS Word application name.
java.lang.String	getWordAppVersion() Returns the MS Word application version.
CursorList	initCursorsFromPlaceholders() (java.util.List figCaptionRanges, java.util.List tabCaptionRanges)
void	initDocgenOptimisationOptions() Where applicable, initialises and stores MS Word application options (speed of doc generation) to original values.
void	insertFigure() (Range range, java.io.File pic)

void	insertFigureCaption (Range range, java.lang.String captionText, java.lang.String logMsg)
void	insertFigureRef (Range insertPointRange, int figNumber)
int	insertTable (Range range, PropertiesDoc doc, Style tabhead)
void	insertTableCaption (Range range, java.lang.String captionText, java.lang.String logMsg)
void	insertTableRef (Range insertPointRange, int tabNumber)
boolean	isInTOC (Range range)
boolean	isRangeWithTable (java.lang.Object range)
void	openDoc () Creates (and caches) the MS Word document.
void	prependNewLine (Range range)
void	prependText (Range range, java.lang.String newText) Prepends text; prepended paragraph will have the same style as the one in range.
void	scanAndPostprocess (WordHelper.PostProcessor pp)
void	setCustomDocProperties (java.util.Map newCustomProps)
void	setDocgenOptimisationOptions () Where applicable, sets MS Word application options to speed performance of doc generation.
void	unsetDocgenOptimisationOptions () Where applicable, brings back the original MS Word application options to values stored with the call to initDocgenOptimisationOptions() .
void	updateFields ()
void	updateTablesOf (java.lang.String what)

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
```

(continued from last page)

Number of points for 1cm (from vba doc).
Constant value: **28.35**

Methods

scanAndPostprocess

```
public void scanAndPostprocess(WordHelper.PostProcessor pp)
```

createWordApp

```
public void createWordApp()
```

Where applicable, launches (and caches) the MS Word application.

getWordAppName

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

Returns the MS Word application name.

getWordAppVersion

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

Returns the MS Word application version.

openDoc

```
public void openDoc()  
    throws java.io.IOException
```

Creates (and caches) the MS Word document.

closeDoc

```
public void closeDoc()  
    throws java.io.IOException
```

Closes and saves the MS Word document.

exitAppAndSaveDocument

```
public void exitAppAndSaveDocument()  
    throws java.io.IOException
```

Saves MS Word document (and where applicable, exits MS Word application).

initDocgenOptimisationOptions

```
public void initDocgenOptimisationOptions()
```

Where applicable, initialises and stores MS Word application options (speed of doc generation) to original values.

(continued from last page)

setDocgenOptimisationOptions

```
public void setDocgenOptimisationOptions()
```

Where applicable, sets MS Word application options to speed performance of doc generation.

unsetDocgenOptimisationOptions

```
public void unsetDocgenOptimisationOptions()
```

Where applicable, brings back the original MS Word application options to values stored with the call to [initDocgenOptimisationOptions\(\)](#).

getExistingStyleNames

```
public java.util.Map getExistingStyleNames()
```

Returns names of existing styles. Key is the name, and the value is the style identifier. Depending on implementation, the two values may be the same.

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)
```

getCursors

```
public CursorList getCursors()
```

isInTOC

```
public boolean isInTOC(Range range)
```

(continued from last page)

closeAndReopenDoc

```
public Cursor closeAndReopenDoc(CursorList cursors,  
    Cursor currentCursor)
```

collectCaptions

```
public java.util.Map collectCaptions()
```

createRange

```
public Range createRange(java.lang.Object object)
```

createCursor

```
public Cursor createCursor(Placeholder ph,  
    Range limited)
```

createCaption

```
public Caption createCaption(Caption.CaptionKind figure,  
    Range range)
```

initCursorsFromPlaceholders

```
public CursorList initCursorsFromPlaceholders(java.util.List figCaptionRanges,  
    java.util.List tabCaptionRanges)
```

createPatternFinder

```
public WordPatternFinder createPatternFinder()
```

clearUndoCache

```
public 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 Range getDocumentAsRange()
```

duplicateRange

```
public Range duplicateRange(Range range)
```

FIXME: could go to Range?

prependText

```
public 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?

getRangeParagraphOutlineLevel

```
public int getRangeParagraphOutlineLevel(Range range,  
    int paraIdx)
```

appendTextInNewParagraphWithStyle

```
public java.lang.String appendTextInNewParagraphWithStyle(Range range,  
    TextDescription newText,  
    Style style)
```

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 void insertFigureCaption(Range range,  
    java.lang.String captionText,  
    java.lang.String logMsg)
```

insertTableCaption

```
public void insertTableCaption(Range range,  
    java.lang.String captionText,  
    java.lang.String logMsg)
```

insertFigureRef

```
public void insertFigureRef(Range insertPointRange,  
    int figNumber)
```

insertTableRef

```
public void insertTableRef(Range insertPointRange,  
    int tabNumber)
```

insertFigure

```
public void insertFigure(Range range,  
    java.io.File pic)
```

insertTable

```
public int insertTable(Range range,  
    PropertiesDoc doc,  
    Style tabhead)
```

getRangeParagraphStyleName

```
public java.lang.String getRangeParagraphStyleName(Range range,  
    int paraIdx)
```

getRangeParagraphCount

```
public int getRangeParagraphCount(Range range)
```

prependNewLine

```
public void prependNewLine(Range range)
```

(continued from last page)

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)
```

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

void	postProcess()
------	-------------------------------

Methods

postProcess

public void **postProcess()**

org.tanjakostic.jcleancim.docgen.writer.word Interface WordPatternFinder

public interface **WordPatternFinder**
extends

Method Summary

Range	getRange()
boolean	hasMore()

Methods

getRange

public [Range](#) **getRange()**

hasMore

public 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

boolean	applyCloseReopen() Returns whether to apply close/reopen hack (may be needed for performance reasons).
Cursor	writeAbbrTable(Cursor initCursor, PackageDoc packageDoc)
Cursor	writeClass(Cursor initCursor, ClassDoc doc) Writes all related to a class at the end of range in initCursor.
Cursor	writeDataIndex(Cursor initCursor, PackageDoc packageDoc)
Cursor	writeDiagram(Cursor cursor, FigureDoc doc) Writes all related to a diagram at the end of range in cursor.
Cursor	writeFcTable(Cursor initCursor, PackageDoc packageDoc)
Cursor	writeLnMapPackage(Cursor initCursor, PackageDoc packageDoc)
Cursor	writePackage(Cursor initCursor, PackageDoc doc, boolean isRoot) Writes all related to a package.
Cursor	writePresCondTable(Cursor initCursor, PackageDoc packageDoc)
Cursor	writeProperties(Cursor initCursor, PropertiesDoc doc) Writes a set of properties as a table at the end of range in cursor.
Cursor	writeSclEnum(Cursor cursor, PackageDoc packageDoc)
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 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 Cursor writeDataIndex(Cursor initCursor,  
    PackageDoc packageDoc)
```

writeLnMapPackage

```
public Cursor writeLnMapPackage(Cursor initCursor,  
    PackageDoc packageDoc)
```

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)
```

(continued from last page)

writeClass

```
public Cursor writeClass(Cursor initCursor,  
                        ClassDoc doc)
```

Writes all related to a class at the end of range in `initCursor`.

writeProperties

```
public 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 Cursor writeDiagram(Cursor cursor,  
                           FigureDoc doc)
```

Writes all related to a diagram at the end of range in `cursor`.

applyCloseReopen

```
public 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, ModelFinder finder, java.lang.String inTemplatePath, java.lang.String outFilePath, boolean useBinaryDoc, boolean introToFigureBefore, int saveRecloseEvery, boolean isDeep) Constructor, useful to create an instance if you don't have configuration.

Method Summary

ModelFinder	getFinder () Returns object that can find in the UML model items specified in placeholders.
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).
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.
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).

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,
                      ModelFinder finder,
                      java.lang.String inTemplatePath,
                      java.lang.String outFilePath,
                      boolean useBinaryDoc,
                      boolean introToFigureBefore,
                      int saveRecloseEvery,
                      boolean isDeep)
```

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.

`finder` - model facade; if null, most of placeholders will have error.

`inTemplatePath`

`outFilePath`

`useBinaryDoc`

`introToFigureBefore`

`saveRecloseEvery`

`isDeep`

Methods

getPackageDocs

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

(continued from last page)

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.

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.

Package

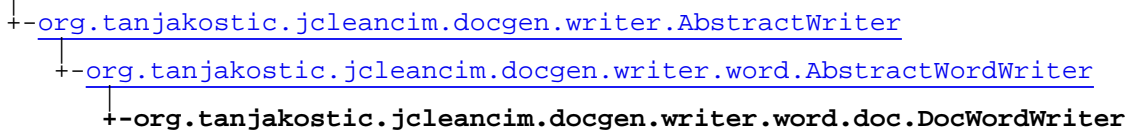
org.tanjakostic.jcleancim.docgen.writer.word.doc

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 the 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



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 Objectype browser in VBA (macro) editor.

We split the processing into two steps:

1. scanning placeholders (and recording ranges), and,
2. overwriting placeholders with the content.

Field Summary

public static final	FILE_EXTENSIONS
public static final	ORIG_CREATE_TABLE Value: true

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 () This default implementation does nothing.
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)
java.util.Map	collectCaptions ()
WordPatternFinder	createPatternFinder ()
Range	createRange (com.jacob.com.Dispatch object)
void	createWordApp ()
Range	duplicateRange (Range range)
void	exitAppAndSaveDocument ()
java.util.Map	getCustomDocProperties ()
Range	getDocumentAsRange ()
java.util.Map	getExistingStyleNames ()
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	insertFigure (Range range, java.io.File pic)
void	insertFigureCaption (Range range, java.lang.String text, java.lang.String logMsg)
void	insertFigureRef (Range insertPointRange, int figNumber)
int	insertTable (Range range, PropertiesDoc doc, Style tabhead)
void	insertTableCaption (Range range, java.lang.String text, java.lang.String logMsg)
void	insertTableRef (Range insertPointRange, int tabNumber)
boolean	isRangeWithTable (com.jacob.com.Dispatch range)
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](#), [initCursorsFromPlaceholders](#), [initDocgenOptimisationOptions](#), [isInTOC](#), [pointForPerc](#), [scanAndPostprocess](#), [setDocgenOptimisationOptions](#), [toString](#), [unsetDocgenOptimisationOptions](#), [write](#), [writeAbbrTable](#), [writeClass](#), [writeDataIndex](#), [writeDiagram](#), [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](#), [writeClass](#), [writeDataIndex](#), [writeDiagram](#), [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](#), [collectCaptions](#), [createCaption](#), [createCursor](#), [createPatternFinder](#), [createRange](#), [createWordApp](#), [duplicateRange](#), [exitAppAndSaveDocument](#), [getCursors](#), [getCustomDocProperties](#), [getDocumentAsRange](#), [getExistingStyleNames](#), [getRangeParagraphCount](#), [getRangeParagraphOutlineLevel](#), [getRangeParagraphStyleName](#), [getWordAppName](#), [getWordAppVersion](#), [initCursorsFromPlaceholders](#), [initDocgenOptimisationOptions](#), [insertFigure](#), [insertFigureCaption](#), [insertFigureRef](#), [insertTable](#), [insertTableCaption](#), [insertTableRef](#), [isInTOC](#), [isRangeWithTable](#), [openDoc](#), [prependNewLine](#), [prependText](#), [scanAndPostprocess](#), [setCustomDocProperties](#), [setDocgenOptimisationOptions](#), [unsetDocgenOptimisationOptions](#), [updateFields](#), [updateTablesOf](#)

Fields

(continued from last page)

ORIG_CREATE_TABLE

```
public static final boolean ORIG_CREATE_TABLE
```

Constant value: **true**

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

[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()
```

getWordAppVersion

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

(continued from last page)

openDoc

```
public void openDoc()
```

closeDoc

```
public void closeDoc()
```

exitAppAndSaveDocument

```
public void exitAppAndSaveDocument()
```

This default implementation does nothing.

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.

getExistingStyleNames

```
public java.util.Map getExistingStyleNames()
```

updateFields

```
public void updateFields()
```

(continued from last page)

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.

collectCaptions

```
public java.util.Map collectCaptions()
```

We have tried with `GetCrossReferenceItems` on document, but that was not reliable.

createRange

```
public Range createRange(com.jacob.com.Dispatch object)
```

createPatternFinder

```
public final WordPatternFinder createPatternFinder()
```

clearUndoCache

```
public void clearUndoCache()
```

This default implementation does nothing.

Scans the whole document to search for placeholders, and records their ranges and captions counts for figures and tables.

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 void insertFigureCaption(Range range,  
    java.lang.String text,  
    java.lang.String logMsg)
```

(continued from last page)

insertTableCaption

```
public void insertTableCaption(Range range,  
    java.lang.String text,  
    java.lang.String logMsg)
```

insertFigureRef

```
public void insertFigureRef(Range insertPointRange,  
    int figNumber)
```

insertTableRef

```
public void insertTableRef(Range insertPointRange,  
    int tabNumber)
```

insertFigure

```
public void insertFigure(Range range,  
    java.io.File pic)
```

insertTable

```
public int insertTable(Range range,  
    PropertiesDoc doc,  
    Style tabhead)
```

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)
```

(continued from last page)

appendNewLine

```
public void appendNewLine(Range range)
```

appendTextInNewParagraph

```
public java.lang.String appendTextInNewParagraph(Range range,  
    java.lang.String newText)
```

appendText

```
public java.lang.String appendText(Range range,  
    java.lang.String newText)
```

isRangeWithTable

```
public boolean isRangeWithTable(com.jacob.com.Dispatch range)
```

collapseRangeToEnd

```
public void collapseRangeToEnd(Range range)
```

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)
java.util.Map	collectCaptions ()
WordPatternFinder	createPatternFinder ()
Range	createRange (java.lang.Object object)
Range	duplicateRange (Range range)
java.util.Map	getCustomDocProperties ()
Range	getDocumentAsRange ()
java.util.Map	getExistingStyleNames ()
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	insertFigure (Range range, java.io.File pic)
void	insertFigureCaption (Range range, java.lang.String text, java.lang.String logMsg)
void	insertFigureRef (Range insertPointRange, int figNumber)
int	insertTable (Range range, PropertiesDoc doc, Style tabhead)
void	insertTableCaption (Range range, java.lang.String text, java.lang.String logMsg)
void	insertTableRef (Range insertPointRange, int tabNumber)

boolean	<u>isRangeWithTable</u> (java.lang.Object range)
void	<u>openDoc</u> ()
void	<u>prependNewLine</u> (<u>Range</u> range)
void	<u>prependText</u> (<u>Range</u> range, java.lang.String newText)
void	<u>setCustomDocProperties</u> (java.util.Map newCustomProps)
void	<u>updateFields</u> ()
void	<u>updateTablesOf</u> (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](#), [initCursorsFromPlaceholders](#), [initDocgenOptimisationOptions](#), [isInTOC](#), [pointForPerc](#), [scanAndPostprocess](#), [setDocgenOptimisationOptions](#), [toString](#), [unsetDocgenOptimisationOptions](#), [write](#), [writeAbbrTable](#), [writeClass](#), [writeDataIndex](#), [writeDiagram](#), [writeFcTable](#), [writeLnMapPackage](#), [writePackage](#), [writePresCondTable](#), [writeProperties](#), [writeScEnum](#), [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](#), [writeClass](#), [writeDataIndex](#), [writeDiagram](#), [writeFcTable](#), [writeLnMapPackage](#), [writePackage](#), [writePresCondTable](#), [writeProperties](#), [writeScEnum](#), [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](#), [collectCaptions](#), [createCaption](#),
[createCursor](#), [createPatternFinder](#), [createRange](#), [createWordApp](#), [duplicateRange](#),
[exitAppAndSaveDocument](#), [getCursors](#), [getCustomDocProperties](#), [getDocumentAsRange](#),
[getExistingStyleNames](#), [getRangeParagraphCount](#), [getRangeParagraphOutlineLevel](#),
[getRangeParagraphStyleName](#), [getWordAppName](#), [getWordAppVersion](#),
[initCursorsFromPlaceholders](#), [initDocgenOptimisationOptions](#), [insertFigure](#),
[insertFigureCaption](#), [insertFigureRef](#), [insertTable](#), [insertTableCaption](#),
[insertTableRef](#), [isInTOC](#), [isRangeWithTable](#), [openDoc](#), [prependNewLine](#), [prependText](#),
[scanAndPostprocess](#), [setCustomDocProperties](#), [setDocgenOptimisationOptions](#),
[unsetDocgenOptimisationOptions](#), [updateFields](#), [updateTablesOf](#)

Fields

FILE_EXTENSIONS

```
public static final java.util.List FILE_EXTENSIONS
```

Constructors

DocxWordWriter

```
public DocxWordWriter(WordWriterInput input)
```

Constructor.

Parameters:

input

Throws:

[UnsupportedInputFormatException](#) - if configured with template in unsupported format

[UnsupportedOutputFormatException](#) - if configured with output in unsupported format

IOException

Methods

getWordAppName

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

getWordAppVersion

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

(continued from last page)

openDoc

```
public void openDoc()  
    throws java.io.IOException
```

closeDoc

```
public void closeDoc()  
    throws java.io.IOException
```

getExistingStyleNames

```
public java.util.Map getExistingStyleNames()
```

updateFields

```
public void updateFields()
```

This is no-op: you need to open the document in MS Word.

updateTablesOf

```
public void updateTablesOf(java.lang.String what)
```

This is no-op: you need to open the document in MS Word.

getCustomDocProperties

```
public java.util.Map getCustomDocProperties()
```

setCustomDocProperties

```
public void setCustomDocProperties(java.util.Map newCustomProps)
```

collectCaptions

```
public java.util.Map collectCaptions()
```

createRange

```
public Range createRange(java.lang.Object object)
```

createPatternFinder

```
public WordPatternFinder createPatternFinder()
```

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 void insertFigureCaption(Range range,  
    java.lang.String text,  
    java.lang.String logMsg)
```

insertTableCaption

```
public void insertTableCaption(Range range,  
    java.lang.String text,  
    java.lang.String logMsg)
```

insertFigureRef

```
public void insertFigureRef(Range insertPointRange,  
    int figNumber)
```

insertTableRef

```
public void insertTableRef(Range insertPointRange,  
    int tabNumber)
```

insertFigure

```
public void insertFigure(Range range,  
    java.io.File pic)
```

insertTable

```
public int insertTable(Range range,  
    PropertiesDoc doc,  
    Style tabhead)
```

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)
```

appendTextInNewParagraph

```
public java.lang.String appendTextInNewParagraph(Range range,  
    java.lang.String newText)
```

appendText

```
public java.lang.String appendText(Range range,  
    java.lang.String newText)
```

isRangeWithTable

```
public boolean isRangeWithTable(java.lang.Object range)
```

collapseRangeToEnd

```
public void collapseRangeToEnd(Range range)
```

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

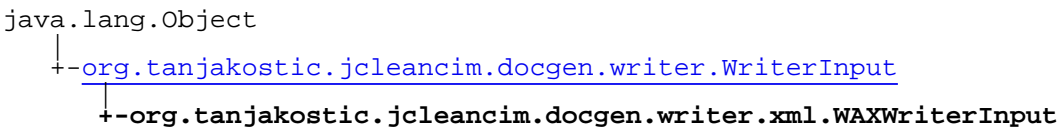
```
public java.util.Set getSupportedFormats()
```

write

```
public void write()
```

org.tanjakostic.jcleancim.docgen.writer.xml

Class WAXWriterInput



```
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!

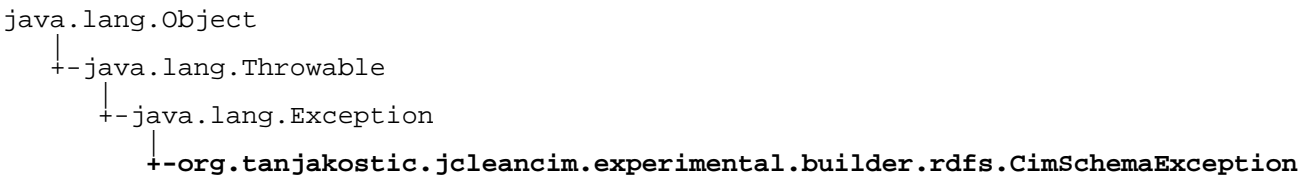
The package contains the [RdfsParser](#), which can parse the CIM profiles in org.tanjakostic.jcleancim.common.Config#XSD_EXT format, as generated with CIMTool.

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



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.rdf

Class RdfsElem

java.lang.Object

↳ org.tanjakostic.jcleancim.experimental.builder.rdf.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 <code>_model</code> .
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
package - 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.rdf.RdfsModel

Class RdfsModel

java.lang.Object

↳ org.tanjakostic.jcleancim.experimental.builder.rdf.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

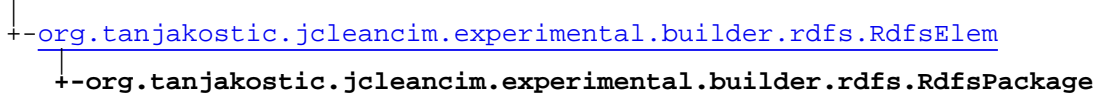
```
public void clear()
```

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



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.rdf

Class RdfsParser

java.lang.Object

↳ org.tanjakostic.jcleancim.experimental.builder.rdf.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.

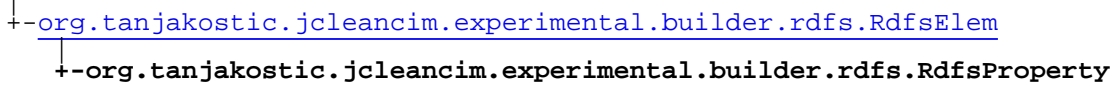
Throws:

IOException

org.tanjakostic.jcleancim.experimental.builder.rdfs

Class RdfsProperty

java.lang.Object



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 <code>_kind</code> .
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

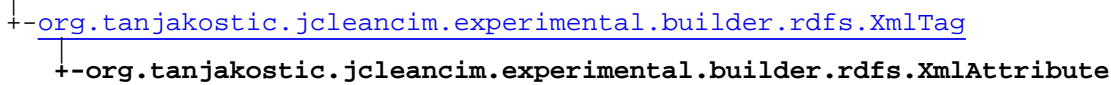
```
public static RdfsProperty.Kind valueOf(java.lang.String name)
```

(continued from last page)

org.tanjakostic.jcleancim.experimental.builder.rdfs

Class XmlAttribute

java.lang.Object



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	<code>getCimNamespaces()</code> Returns namespace information for all model references found in the profile, with prefix as key, and URI as value.
java.util.Collection	<code>getCimPrimitives()</code>
java.util.Map	<code>getClassesAndCompounds()</code>
java.util.Map	<code>getDatatypes()</code>
java.util.Map	<code>getEnums()</code>
java.lang.String	<code>getEnvelopeName()</code> Returns envelope name (in instance file, this will be the root element).
java.util.Map	<code>getLiterals()</code>
java.lang.String	<code>getName()</code> Returns profile name (deduced from the file name, without extension); for envelope name, use <code>getEnvelopeName()</code> .
java.util.List	<code>getNamespaces()</code>
static java.util.Map	<code>getPrimitives()</code>
java.util.List	<code>getSubdirNames()</code> Returns list of names, deduced from the profile file path.
java.util.Map	<code>getTargetNamespace()</code> Returns the target namespace information, with prefix #TARGET_NS_PREFIX as key, and URI as value.
java.util.Set	<code>getUnclassifieds()</code>
java.io.File	<code>getXsdFile()</code> Returns the file that has been used for initialisation, null if the profile has been created from a string.
boolean	<code>hasInconsistentEnvelopeName()</code> Returns whether this profile follows the convention to have the envelope name same as the profile name.
boolean	<code>hasInconsistentNamespace()</code> 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	<code>toString()</code>

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 recursively.

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 initialisation, 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 internally 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 `javax.sql.CommonDataSource` 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: <code>.</code>
public static final	NULL_OBJ_NAME Value: <code>null</code>
public static final	PACKAGE_SEPARATOR Package separator, for qualified names. Value: <code>::</code>

Constructor Summary

protected	AbstractUmlObject (UmlObjectData objData) Constructor.
-----------	--

Method Summary

java.lang.String	addTaggedValue (java.lang.String name, java.lang.String value)
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 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 ()
java.util.Set	getPredefinedTagNames ()
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()
boolean	isDeprecated()
abstract 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 override 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](#), [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.
Constant value: `::`

CLASS_SEPARATOR

```
public static final java.lang.String CLASS_SEPARATOR
```

Class separator, for qualified names.
Constant value: `.`

Constructors

(continued from last page)

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.

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.

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)
```

(continued from last page)

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.

getId

```
public final java.lang.Integer getId()
```

getUuid

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

getSince

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

getOwner

```
public abstract OwningWg getOwner()
```

getNature

```
public abstract Nature getNature()
```

(continued from last page)

isInformative

```
public abstract boolean isInformative()
```

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](#).

getPredefinedTagNames

```
public java.util.Set getPredefinedTagNames()
```

This default implementation returns empty set, and should be overridden by subclasses that have something to return.

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 overwrite 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()
```

(continued from last page)

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 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.

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.

Method Summary

boolean	addDependency (NamespaceInfo namespace) Adds namespace as dependency to this namespace, if the dependency is not circular and returns whether addition happened.
java.lang.String	getDate ()
java.util.Set	getDependencies () Returns all dependencies of this namespace.
java.util.List	getDependencyStrings () Returns namespace strings of all the dependencies.
java.lang.String	getId ()
java.lang.String	getNamespace () 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.

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.

Parameters:

id
version
revision
date
umlVersion
tissuesApplied

Methods

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 circular and returns whether addition happened.

getNamespace

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

Returns formatted string including id, version and revision.

getDependencyStrings

```
public java.util.List 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 one a standard one.
static java.util.Set	getNamesOfImplicits()
java.lang.String	getStem()

java.lang.String	getStemAndArgs() Returns <code>stem(args)</code> if there are arguments, otherwise just <code>stem</code> .
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 one 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 <code>text</code> 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



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 ind	valueOf (java.lang.String name)
static TextDescription.TextKind ind[]	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 only.

Field Summary

public static final	<u>ACT</u> Value: ACT
public static final	<u>ATTR_attr</u> Private attribute name on FCDA meta-model type, to hold actual DA type used. Value: attr
public static final	<u>ATTR_DATE</u> Attribute in a top-package version class (and 61850 namespace class). Value: date
public static final	<u>ATTR_id</u> Value: id
public static final	<u>ATTR_revision</u> Value: revision
public static final	<u>ATTR_tissuesApplied</u> Value: tissuesApplied
public static final	<u>ATTR_val</u> Private attribute name on DA meta-model type, to hold actual primitive type used. Value: val
public static final	<u>ATTR_VERSION</u> Attribute in a top-package version class (and 61850 namespace class). Value: version
public static final	<u>BasePrimitiveCDC</u> Value: BasePrimitiveCDC
public static final	<u>CDCAnalogueInfo</u> 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	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	<u>DA</u> Value: DA
public static final	<u>DetailedDiagrams</u> 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	<u>DOC_FORMAT_STRING</u> CIM-specific 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	<u>ENC</u> Value: ENC
public static final	<u>ENG</u> Value: ENG
public static final	<u>ENS</u> Value: ENS
public static final	<u>ERY</u> Value: ERY
public static final	<u>FCDA_BL</u> Value: FCDA_BL
public static final	<u>FCDA_CF</u> Value: FCDA_CF
public static final	<u>FCDA_DC</u> Value: FCDA_DC
public static final	<u>FCDA_EX</u> Value: FCDA_EX
public static final	<u>FCDA_MX</u> Value: FCDA_MX
public static final	<u>FCDA_OR</u> Value: FCDA_OR
public static final	<u>FCDA_SE</u> Value: FCDA_SE
public static final	<u>FCDA_SG</u> 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_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 a couple of 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_iecRef Value: iecRef
public static final	TVN_ieeeRef Value: ieeeRef
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

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**

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**

(continued from last page)

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
```

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
```

(continued from last page)

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**

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**

(continued from last page)

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
```

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 a couple of 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
```

(continued from last page)

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
```

(continued from last page)

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**

(continued from last page)

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_
```

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.

(continued from last page)

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 actual primitive type used.
Constant value: **val**

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**

(continued from last page)

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
```

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 version class (and 61850 namespace class).
Constant value: **version**

ATTR_DATE

```
public static final java.lang.String ATTR_DATE
```

Attribute in a top-package version class (and 61850 namespace class).
Constant value: **date**

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
```

CIM-specific 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

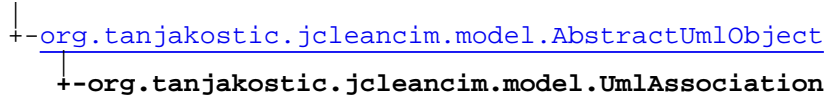
```
public static final java.lang.String DetailedDiagrams
```

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



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

Field Summary

public static final	STEREOTYPES Allowed stereotypes for UML associations.
---------------------	--

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() Returns the direction (navigability).
UmlAssociationEnd.AssociationEndPair	getEndsFor(UmlClass type) Returns the pair of association ends from the perspective of type (this is what we need when printing model documentation for associations of a type, or when reading RDF/OWL properties).
UmlAssociationEnd.AssociationEndPair	getEndsForSelfAssociation(boolean asSource) For recursive association (self-association on a class), returns the pair of association ends from the perspective of source if asSource true, from the perspective of target otherwise.
UmlKind	getKind()
static java.util.List	getKinds(Nature nature) Returns all available classifications (kinds) for associations.
Nature	getNature()
OwningWg	getOwner()
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	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 direction has a navigable end.
boolean	isInformative()
boolean	isMapping() Returns whether this association is between a CIM and an IEC 61850 class.
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](#)

[addTaggedValue](#), [classifyPerScope](#), [classifyPerScopePerTag](#), [classifyPerTag](#), [collectDuplicateDescriptions](#), [collectDuplicateNames](#), [collectForScope](#), [collectNames](#), [collectQNames](#), [findWithSameUuidAndLog](#), [getAlias](#), [getDescription](#), [getHtmlDescription](#), [getId](#), [getKind](#), [getName](#), [getNature](#), [getOwner](#), [getPredefinedTagNames](#), [getQualifiedName](#), [getSince](#), [getStereotype](#), [getTaggedValues](#), [getUnallowedTagNames](#), [getUuid](#), [getVisibility](#), [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](#), [getNature](#), [getOwner](#), [getPredefinedTagNames](#), [getQualifiedName](#), [getSince](#), [getStereotype](#), [getTaggedValues](#), [getUnallowedTagNames](#), [getUuid](#), [getVisibility](#), [isDeprecated](#), [isInformative](#), [toShortString](#)

Fields

STEREOTYPES

```
public static final java.util.List STEREOTYPES
```

Allowed stereotypes for UML associations.

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.

getEndsFor

```
public UmlAssociationEnd.AssociationEndPair getEndsFor(UmlClass type)
```

Returns the pair of association ends from the perspective of `type` (this is what we need when printing model documentation for associations of a `type`, or when reading RDF/OWL properties).

getEndsForSelfAssociation

```
public UmlAssociationEnd.AssociationEndPair getEndsForSelfAssociation(boolean asSource)
```

For recursive association (self-association on a class), returns the pair of association ends from the perspective of source if `asSource` true, from the perspective of target otherwise.

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()
```

Returns the direction (navigability).

isDirectionMismatchForEnds

```
public boolean isDirectionMismatchForEnds()
```

Returns whether an association with unspecified direction 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 direction for its navigable end to unspecified.

isMapping

```
public boolean isMapping()
```

Returns whether this association is between a CIM and an IEC 61850 class.

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()
```

getNature

```
public Nature getNature()
```

isInformative

```
public boolean isInformative()
```

getKind

```
public UmlKind getKind()
```

getQualifiedName

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

(continued from last page)

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	UmlAssociation.Data (UmlAssociation.Direction direction) Constructor.
--------	---

Method Summary

static UmlAssociation.Data	empty () Returns empty instance; sets default multiplicity to import org.tanjakostic.jcleancim.common.OwningWg; #ONE , and direction to empty string.
UmlAssociation.Direction	getDirection ()

Methods inherited from class java.lang.Object

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

Constructors

UmlAssociation.Data

public **UmlAssociation.Data**([UmlAssociation.Direction](#) direction)

Constructor.

Parameters:

direction

Methods

empty

public static [UmlAssociation.Data](#) **empty**()

Returns empty instance; sets default multiplicity to [import org.tanjakostic.jcleancim.common.OwningWg; #ONE](#), and direction to empty string.

(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.AssociationEndPair UmlAssociationEnd.AssociationEndPair
class	UmlAssociationEnd.Data UmlAssociationEnd.Data
class	UmlAssociationEnd.Kind UmlAssociationEnd.Kind
class	UmlAssociationEnd.Navigable UmlAssociationEnd.Navigable

Field Summary

public static final	STEREOTYPES Allowed stereotypes for UML association ends.
---------------------	--

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()
Nature	getNature()
UmlAssociationEnd.Navigable	getNavigable()
OwningWg	getOwner()
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()
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](#)

[addTaggedValue](#), [classifyPerScope](#), [classifyPerScopePerTag](#), [classifyPerTag](#), [collectDuplicateDescriptions](#), [collectDuplicateNames](#), [collectForScope](#), [collectNames](#), [collectQNames](#), [findWithSameUuidAndLog](#), [getAlias](#), [getDescription](#), [getHtmlDescription](#), [getId](#), [getKind](#), [getName](#), [getNature](#), [getOwner](#), [getPredefinedTagNames](#), [getQualifiedName](#), [getSince](#), [getStereotype](#), [getTaggedValues](#), [getUnallowedTagNames](#), [getUuid](#), [getVisibility](#), [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](#), [getNature](#), [getOwner](#), [getPredefinedTagNames](#), [getQualifiedName](#), [getSince](#), [getStereotype](#), [getTaggedValues](#), [getUnallowedTagNames](#), [getUuid](#), [getVisibility](#), [isDeprecated](#), [isInformative](#), [toShortString](#)

Fields

STEREOTYPES

```
public static final java.util.List STEREOTYPES
```

Allowed stereotypes for UML association ends.

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.

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()
```

(continued from last page)

Returns whether this end is the source end of the containing association.

isTarget

```
public boolean isTarget()
```

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.

getNature

```
public Nature getNature()
```

isInformative

```
public boolean isInformative()
```

Before two instances of this are used to create an association, returns false.

Association end is informative if its type, or its other end type is informative, or if the containing association is tagged as 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).

isDeprecated

```
public boolean isDeprecated()
```

(continued from last page)

This default implementation returns whether the stereotype string of this object contains the string [UmlStereotype.DEPRECATED](#).

The implementation of UML association end deduces the value to return as follows:

1. If this instance has stereotype containing org.tanjakostic.jcleancim.model.UmlStereotype#DEPRECATED, returns true.
2. If the containing association is null (i.e., this end has not been added to an association), returns false.
3. Finally, returns [AbstractUmlObject.isDeprecated\(\)](#).

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 d[]	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	UmlAssociationEnd.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

UmlAssociationEnd.Data

```
public UmlAssociationEnd.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 UmlAssociationEnd.AssociationEndPair

java.lang.Object

└─org.tanjakostic.jcleancim.model.UmlAssociationEnd.AssociationEndPair

public static class **UmlAssociationEnd.AssociationEndPair**
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 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()**

(continued from last page)

getOtherEnd

```
public UmlAssociationEnd getOtherEnd()
```

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
public static final	STEREOTYPES Allowed stereotypes for UML attributes.

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	<u>findAbbreviationLiterals</u> (java.util.Collection attributes)
static java.util.Collection	<u>findEnumLiterals</u> (java.util.Collection attributes)
static java.util.Collection	<u>findPresenceConditionLiterals</u> (java.util.Collection attributes)
java.util.Collection	<u>getAllSiblings</u> () Returns all (native and inherited) sibling attributes.
java.lang.String	<u>getArrayBounds</u> () Returns formatted string "minId...maxId" created from attribute constraints if existing, empty string otherwise.
java.lang.String	<u>getArrayBoundsWithBrackets</u> ()
java.util.Map	<u>getConstraintValues</u> () Returns map of {name, constraint} pairs defined as attribute constraints.
<u>UmlClass</u>	<u>getContainingClass</u> () Returns class containing this attribute.
java.util.List	<u>getDsPresConditions</u> (<u>UmlClass</u> context) (IEC 61850) Returns derived statistics presence conditions.
int	<u>getEaTypeId</u> () See <u>UmlAttribute.Data.getEaTypeId()</u> .
java.lang.String	<u>getEaTypeInfo</u> () 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	<u>getEaTypeName</u> () See <u>UmlAttribute.Data.getEaTypeName()</u> .
java.lang.Integer	<u>getInitialValueAsInteger</u> () Returns the initial value if it can be interpreted as integer, null otherwise.
java.lang.String	<u>getInitValue</u> () See <u>UmlAttribute.Data.getInitValue()</u> .
java.lang.String	<u>getInitValueWithPotentialOverrideForSCL</u> () (Special handling for IEC61850) Returns empty string as initial value for the case of an attribute that returns true from <u>displayEmptyValue()</u> .
<u>UmlKind</u>	<u>getKind</u> ()
static java.util.List	<u>getKinds</u> (<u>Nature</u> nature) Returns available classifications (kinds) for attributes.
<u>UmlMultiplicity</u>	<u>getMultiplicity</u> () See <u>UmlAttribute.Data.getMultiplicity()</u> .
<u>NameDecomposition</u>	<u>getNameDecomposition</u> () Equivalent to <u>getNameDecomposition(null)</u> .

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.
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	<u>isMultivalued()</u> Returns whether this attribute is multivalued.
boolean	<u>isOptional()</u> Returns whether the multiplicity is optional.
boolean	<u>isPublic()</u> Returns whether this attribute is public.
boolean	<u>isStatic()</u> See <u>UmlAttribute.Data.isStatic()</u> .
java.lang.String	<u>toString()</u>

Methods inherited from class [org.tanjakostic.jcleancim.model.AbstractUmlObject](#)

[addTaggedValue](#), [classifyPerScope](#), [classifyPerScopePerTag](#), [classifyPerTag](#), [collectDuplicateDescriptions](#), [collectDuplicateNames](#), [collectForScope](#), [collectNames](#), [collectQNames](#), [findWithSameUuidAndLog](#), [getAlias](#), [getDescription](#), [getHtmlDescription](#), [getId](#), [getKind](#), [getName](#), [getNature](#), [getOwner](#), [getPredefinedTagNames](#), [getQualifiedName](#), [getSince](#), [getStereotype](#), [getTaggedValues](#), [getUnallowedTagNames](#), [getUuid](#), [getVisibility](#), [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](#), [getNature](#), [getOwner](#), [getPredefinedTagNames](#), [getQualifiedName](#), [getSince](#), [getStereotype](#), [getTaggedValues](#), [getUnallowedTagNames](#), [getUuid](#), [getVisibility](#), [isDeprecated](#), [isInformative](#), [toShortString](#)

Fields

STEREOTYPES

```
public static final java.util.List STEREOTYPES
```

Allowed stereotypes for UML attributes.

DO_MAX_LENGTH

```
public static final int DO_MAX_LENGTH
```

Constant value: **12**

Methods

(continued from last page)

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\(\)](#).

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.

(continued from last page)

isLiteral

```
public boolean isLiteral()
```

CIM and IEC61850

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.

(continued from last page)

getOwnConstraints

```
public java.util.List getOwnConstraints()
```

Returns constraints defined on this attribute.

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.

(continued from last page)

hasInitialValueAsInteger

```
public boolean hasInitialValueAsInteger()
```

Returns whether the initial value (when present) was interpreted as integer.

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()
```

getNature

```
public Nature getNature()
```

Returns the nature of containing clas.

isInformative

```
public boolean isInformative()
```

Returns true if the containing class is informative.

getKind

```
public UmlKind getKind()
```

getQualifiedName

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

(continued from last page)

getPredefinedTagNames

```
public java.util.Set getPredefinedTagNames()
```

This default implementation returns empty set, and should be overridden by subclasses that have something to return.

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)

<code>static UmlAttribute.Kind[]</code>	<code>values()</code>
---	---------------------------------------

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	UmlAttribute.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)

UmlAttribute.Data

```
public UmlAttribute.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	CIMDATATYPE_MIN_SPEC Minimum requirement for a valid UmlStereotype.CIMDATATYPE : key is attribute name (AbstractUmlObject.getName()) and value is the kind of its type ((UmlAttribute.getKind())).
public static final	SUPER_COMP_CDC Value: ComposedCDC
public static final	SUPER_ENUM_LNS
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	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.
java.util.Map	getConstraints()
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()

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.List	getSuperclassChain() Starting from direct superclasses, returns all the superclasses up to the root.
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	<u>isAnyDA()</u> IEC 61850 (like DA or Vector)
boolean	<u>isAnyFCDA()</u> IEC 61850 (like FCDA_ST_dchg or INT32_ST_dchg)
boolean	<u>isAnyLN()</u> IEC 61850
boolean	<u>isAssociationClass()</u>
boolean	<u>isBasic()</u> IEC 61850 (P_*)
boolean	<u>isClass()</u> CIM non-root class without stereotype.
boolean	<u>isCodedEnum()</u> IEC 61850
boolean	<u>isCodedEnumDA()</u> IEC 61850
boolean	<u>isCodedEnumFCDA()</u> IEC 61850 (like PackedEnumFCDA_ST_dchg or DpStatus_ST_dchg)
boolean	<u>isComposedCDC()</u> IEC 61850 (like ComposedCDC or WYE)
boolean	<u>isComposedDA()</u> IEC 61850
boolean	<u>isComposedFCDA()</u> IEC 61850 (like Analog_MX_dchg)
boolean	<u>isCompound()</u> CIM
boolean	<u>isConditionEnumeration()</u> IEC 61850
boolean	<u>isDatatype()</u> CIM
boolean	<u>isEaInterface()</u>
boolean	<u>isEaLeafPropSet()</u>
boolean	<u>isEaPersistentPropSet()</u>
boolean	<u>isEaRootPropSet()</u>
boolean	<u>isEnumCDC()</u> IEC 61850 (like ENS and its subtypes, derived from SPS)
boolean	<u>isEnumDA()</u> IEC 61850

boolean	<u>isEnumeratedType()</u> Returns whether this is an enumerated type.
boolean	<u>isEnumeration()</u> CIM and IEC61850 - simple enumeration with no other stereotypes
boolean	<u>isEnumFCDA()</u> IEC 61850 (like EnumDA_ST_dchg or CurveChar_SP_dchg)
boolean	<u>isFCDA()</u> IEC 61850 (FCDA from the meta-model)
boolean	<u>isFrom72()</u> IEC 61850 - returns whether the class is from the model supporting IEC 61850-7-2.
boolean	<u>isFromMetaModel()</u> IEC 61850 - returns whether the class is from the meta-model (package and subpackages).
boolean	<u>isFunction()</u> IEC 61850
boolean	<u>isInformative()</u>
boolean	<u>isInterface()</u> IEC 61850
boolean	<u>isNamespaceClass()</u>
boolean	<u>isNullClass()</u>
boolean	<u>isOrHasSuperclass(java.lang.String name)</u> Returns whether name is this class or one of its superclasses in the inheritance chain.
boolean	<u>isOrInheritsFromStatisticsLN()</u> (61850) Returns whether this class is itself or inherits from org.tanjakostic.jcleancim.model.UML#StatisticsLN
boolean	<u>isOther()</u> IEC 61850
boolean	<u>isPackedList()</u> IEC 61850
boolean	<u>isPackedListDA()</u> IEC 61850
boolean	<u>isPackedListFCDA()</u> IEC 61850 (like Quality_ST_dchg)
boolean	<u>isPrimitive()</u> CIM
boolean	<u>isPrimitiveCDC()</u> IEC 61850 (like PrimitiveCDC or SPS)
boolean	<u>isPrimitiveDA()</u> IEC 61850

boolean	<u>isSelfInherited()</u>
boolean	<u>isStructured()</u> IEC 61850 (S_*); at present, S_Originator only
boolean	<u>isTrackingDerivedCDC()</u> IEC 61850 (like CTSINT32, derived from CTS)
boolean	<u>isTransientCDC()</u> IEC 61850 (like SPCTransient, derived from SPC)
boolean	<u>isUnknown()</u> IEC 61850 - with an unknown stereotype
boolean	<u>isUsableForStatistics()</u> (61850) Returns whether this class or any of its superclasses has stereotype org.tanjakostic.jcleancim.model.UmlStereotype#STATISTICS
boolean	<u>isUsedAsTypeForAttributes()</u>
boolean	<u>isVersionClass()</u>
boolean	<u>isWithOldDataStereotype()</u>
boolean	<u>needsAlias()</u> (61850) Returns whether this class needs alias, for doc generation purposes.
boolean	<u>needsTags()</u> (61850) Returns whether this class needs tagged values, for doc generation purposes.
java.lang.String	<u>toString()</u>

Methods inherited from class [org.tanjakostic.jcleancim.model.UmlStructure](#)

[addDependency](#), [addDiagram](#), [addSkippedUmlItem](#), [collectDependencyAfferentStructures](#), [collectDependencyEfferentStructures](#), [getDependenciesAsSource](#), [getDependenciesAsTarget](#), [getDiagrams](#), [getKind](#), [getModel](#), [getNature](#), [getOwner](#), [getQualifiedName](#), [getSkippedUmlItems](#), [isInformative](#), [isSelfDependent](#)

Methods inherited from class [org.tanjakostic.jcleancim.model.AbstractUmlObject](#)

[addTaggedValue](#), [classifyPerScope](#), [classifyPerScopePerTag](#), [classifyPerTag](#), [collectDuplicateDescriptions](#), [collectDuplicateNames](#), [collectForScope](#), [collectNames](#), [collectQNames](#), [findWithSameUuidAndLog](#), [getAlias](#), [getDescription](#), [getHtmlDescription](#), [getId](#), [getKind](#), [getName](#), [getNature](#), [getOwner](#), [getPredefinedTagNames](#), [getQualifiedName](#), [getSince](#), [getStereotype](#), [getTaggedValues](#), [getUnallowedTagNames](#), [getUuid](#), [getVisibility](#), [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, getNature, getOwner, getPredefinedTagNames, getQualifiedName, getSince,  
getStereotype, getTaggedValues, getUnallowedTagNames, getUuid, getVisibility,  
isDeprecated, isInformative, toShortString
```

Fields

CIMDATATYPE_MIN_SPEC

```
public static final java.util.Map CIMDATATYPE_MIN_SPEC
```

Minimum requirement for a valid [UmlStereotype.CIMDATATYPE](#): key is attribute name ([AbstractUmlObject.getName\(\)](#)) and value is the kind of its type (([UmlAttribute.getKind\(\)](#))).

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**

SUPER_ENUM_LNS

```
public static final java.util.List SUPER_ENUM_LNS
```

Constructors

(continued from last page)

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

getKinds

```
public static java.util.List getKinds(Nature nature)
```

Returns available classifications (kinds) for classes with nature.

getSuperclassChain

```
public java.util.List getSuperclassChain()
```

Starting from direct superclasses, returns all the superclasses up to the root.

getContainingPackage

```
public UmlPackage getContainingPackage()
```

isNullClass

```
public boolean isNullClass()
```

isVersionClass

```
public boolean isVersionClass()
```

(continued from last page)

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()
```

isAssociationClass

```
public boolean isAssociationClass()
```

isSelfInherited

```
public boolean isSelfInherited()
```

getRsName

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

(continued from last page)

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.

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

(continued from last page)

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()
```

IEC 61850

isCodedEnum

```
public boolean isCodedEnum()
```

IEC 61850

isAbbreviationEnumeration

```
public boolean isAbbreviationEnumeration()
```

IEC 61850

isConditionEnumeration

```
public boolean isConditionEnumeration()
```

(continued from last page)

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

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)

(continued from last page)

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()
```

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()
```

(continued from last page)

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

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.

(continued from last page)

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)
```

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

(continued from last page)

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.

Parameters:

objData
data

getConstraints

```
public java.util.Map getConstraints()
```

addAttribute

```
public UmlAttribute addAttribute(UmlClass type,  
    UmlObjectData objData,  
    UmlAttribute.Data data)
```

(continued from last page)

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.

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:

`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()
```

(continued from last page)

Returns pairs of association ends, from the perspective of this class (this end vs. other end) - convenient for documentation generation.

See Also:

[UmlAssociationEnd.AssociationEndPair](#)

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:

[UmlAssociationEnd.AssociationEndPair](#)

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.

getOwner

```
public OwningWg getOwner()
```

getNature

```
public Nature getNature()
```

isInformative

```
public boolean isInformative()
```

getKind

```
public UmlKind getKind()
```


(continued from last page)

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()
```

This default implementation returns empty set, and should be overridden by subclasses that have something to return.

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)

<code>static UmlClass.CimKind[]</code>	<code>values()</code>
--	---------------------------------------

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 d	valueOf (java.lang.String name)
static UmlClass.InheritedKind d[]	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	UmlClass.Data (boolean selfDependent, boolean isAbstract, boolean eaPersistentPropSet, boolean eaLeafPropSet, boolean eaRootPropSet, boolean isEaInterface, boolean associationClass, boolean selfInherited) Constructor.
public	UmlClass.Data (UmlStructure.Data data, boolean isAbstract, boolean eaPersistentPropSet, boolean eaLeafPropSet, boolean eaRootPropSet, boolean isEaInterface, boolean associationClass, boolean selfInherited) Constructor.

Method Summary

static UmlClass.Data	empty() Returns an empty instance.
boolean	isAbstract()
boolean	isAssociationClass()
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

UmlClass.Data

```
public UmlClass.Data(boolean selfDependent,  
                     boolean isAbstract,  
                     boolean eaPersistentPropSet,  
                     boolean eaLeafPropSet,  
                     boolean eaRootPropSet,  
                     boolean isEaInterface,  
                     boolean associationClass,  
                     boolean selfInherited)
```

Constructor.

Parameters:

selfDependent
isAbstract
eaPersistentPropSet
eaLeafPropSet
eaRootPropSet
isEaInterface
associationClass
selfInherited

UmlClass.Data

```
public UmlClass.Data(UmlStructure.Data data,  
                     boolean isAbstract,  
                     boolean eaPersistentPropSet,  
                     boolean eaLeafPropSet,  
                     boolean eaRootPropSet,  
                     boolean isEaInterface,  
                     boolean associationClass,  
                     boolean selfInherited)
```

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()
```

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. Allows us to attach constraints to attributes of IEC61850 LNs and CDCs.

Currently, the only supported constraints are those attached to LN and CDC classes. For recognised formats, see [UmlConstraint.Kind](#).

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 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()
Nature	getNature()
OwningWg	getOwner()
PresenceCondition	getPresenceCondition() Returns presence condition deduced from this class constraint; null for attribute constraint.
java.lang.String	getQualifiedName()
boolean	isInformative()
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](#)

[addTaggedValue](#), [classifyPerScope](#), [classifyPerScopePerTag](#), [classifyPerTag](#), [collectDuplicateDescriptions](#), [collectDuplicateNames](#), [collectForScope](#), [collectNames](#), [collectQNames](#), [findWithSameUuidAndLog](#), [getAlias](#), [getDescription](#), [getHtmlDescription](#), [getId](#), [getKind](#), [getName](#), [getNature](#), [getOwner](#), [getPredefinedTagNames](#), [getQualifiedName](#), [getSince](#), [getStereotype](#), [getTaggedValues](#), [getUnallowedTagNames](#), [getUuid](#), [getVisibility](#), [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](#), [getNature](#), [getOwner](#), [getPredefinedTagNames](#), [getQualifiedName](#), [getSince](#), [getStereotype](#), [getTaggedValues](#), [getUnallowedTagNames](#), [getUuid](#), [getVisibility](#), [isDeprecated](#), [isInformative](#), [toShortString](#)

Fields

SEPARATOR

```
public static final java.lang.String SEPARATOR
```

Used to separate attribute names from the condition text for class constraints.

Constant value: :

Methods

(continued from last page)

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 class constraint; null 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()
```

getNature

```
public Nature getNature()
```

isInformative

```
public boolean isInformative()
```

getKind

```
public UmlKind getKind()
```

(continued from last page)

getQualifiedName

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

validateTag

```
protected void validateTag(java.lang.String name,  
    java.lang.String value)
```

Subclasses should override this method in case some validation about the tagged value is needed before adding it. This default implementation is a no-op.

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 of 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 of 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	UmlConstraint.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

UmlConstraint.Data

```
public UmlConstraint.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

Field Summary

public static final	STEREOTYPES Allowed stereotypes for UML dependencies.
---------------------	--

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.
Nature	getNature()
OwningWg	getOwner()
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()
boolean	isWithinSameWg() Returns whether source and target have the same owner.
java.lang.String	toString()

Methods inherited from class [org.tanjakostic.jcleancim.model.AbstractUmlObject](#)

[addTaggedValue](#), [classifyPerScope](#), [classifyPerScopePerTag](#), [classifyPerTag](#), [collectDuplicateDescriptions](#), [collectDuplicateNames](#), [collectForScope](#), [collectNames](#), [collectQNames](#), [findWithSameUuidAndLog](#), [getAlias](#), [getDescription](#), [getHtmlDescription](#), [getId](#), [getKind](#), [getName](#), [getNature](#), [getOwner](#), [getPredefinedTagNames](#), [getQualifiedName](#), [getSince](#), [getStereotype](#), [getTaggedValues](#), [getUnallowedTagNames](#), [getUuid](#), [getVisibility](#), [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](#), [getNature](#), [getOwner](#), [getPredefinedTagNames](#), [getQualifiedName](#), [getSince](#), [getStereotype](#), [getTaggedValues](#), [getUnallowedTagNames](#), [getUuid](#), [getVisibility](#), [isDeprecated](#), [isInformative](#), [toShortString](#)

Fields

STEREOTYPES

```
public static final java.util.List STEREOTYPES
```

Allowed stereotypes for UML dependencies.

Methods

getKinds

```
public static java.util.List getKinds(Nature nature)
```

Returns all available classifications (kinds) for dependencies.

Parameters:

nature - ignored in this method

(continued from last page)

getSource

```
public UmlStructure getSource()
```

Returns source.

getTarget

```
public UmlStructure getTarget()
```

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()
```

getNature

```
public Nature getNature()
```

isInformative

```
public boolean isInformative()
```

Dependency is never informative, and this method always returns false.

getKind

```
public UmlKind getKind()
```

getQualifiedName

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

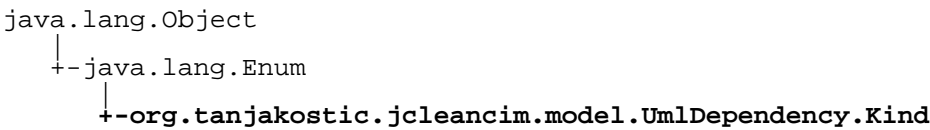
(continued from last page)

toString

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

org.tanjakostic.jcleancim.model

Class 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	
<code>java.lang.String</code>	getDesc()
<code>java.lang.String</code>	getLabel()
<code>java.lang.String</code>	getTag()
<code>java.lang.String</code>	getValue()
static UmlDependency.Kind	valueOf(java.lang.String name)
static UmlDependency.Kind[]	values()

Methods inherited from class <code>java.lang.Enum</code>
<code>clone</code> , <code>compareTo</code> , <code>equals</code> , <code>finalize</code> , <code>getDeclaringClass</code> , <code>hashCode</code> , <code>name</code> , <code>ordinal</code> , <code>toString</code> , <code>valueOf</code>

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

Methods inherited from interface <code>java.lang.Comparable</code>
<code>compareTo</code>

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	UmlDependency.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

UmlDependency.Data

public **UmlDependency.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

Field Summary

public static final	STEREOTYPES Allowed stereotypes for UML diagrams.
---------------------	--

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.
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.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](#)

[addTaggedValue](#), [classifyPerScope](#), [classifyPerScopePerTag](#), [classifyPerTag](#), [collectDuplicateDescriptions](#), [collectDuplicateNames](#), [collectForScope](#), [collectNames](#), [collectQNames](#), [findWithSameUuidAndLog](#), [getAlias](#), [getDescription](#), [getHtmlDescription](#), [getId](#), [getKind](#), [getName](#), [getNature](#), [getOwner](#), [getPredefinedTagNames](#), [getQualifiedName](#), [getSince](#), [getStereotype](#), [getTaggedValues](#), [getUnallowedTagNames](#), [getUuid](#), [getVisibility](#), [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](#), [getNature](#), [getOwner](#), [getPredefinedTagNames](#), [getQualifiedName](#), [getSince](#), [getStereotype](#), [getTaggedValues](#), [getUnallowedTagNames](#), [getUuid](#), [getVisibility](#), [isDeprecated](#), [isInformative](#), [toShortString](#)

Fields

STEREOTYPES

```
public static final java.util.List STEREOTYPES
```

Allowed stereotypes for UML diagrams.

Methods

(continued from last page)

getKinds

```
public static java.util.List getKinds(Nature nature)
```

Returns all available classifications (kinds) for diagrams.

Parameters:

nature - ignored in this method

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()
```

isInformative

```
public boolean isInformative()
```

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 override this method in case some validation about the tagged value is needed before adding it. This default implementation is a no-op.

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	UmlDiagram.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

UmlDiagram.Data

```
public UmlDiagram.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

java.lang.String	getDesc() Returns the description, typically used for statistics printing.
java.lang.String	getLabel() Returns the label to be used for human-readable documentation.
java.lang.String	getTag() Returns the tag to be used for machine-processable documentation, typically used for statistics printing.
java.lang.String	getValue() Returns the value set by the custom constructor.

Methods

getValue

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

Returns the value set by the custom constructor. E.g., for ENUM("enumeration", "enumeration" "Enumeration"), returns first argument "enumeration".

getLabel

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

Returns the label to be used for human-readable documentation. E.g., ENUM("enumeration", "enumeration" "Enumeration"), returns second argument "enumeration".

getTag

```
public 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("enumeration", "enumeration" "Enumeration") returns third argument "Enumeration".

getDesc

```
public 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 two such models:

- TC57CIM, with [Nature.CIM](#) (default), 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 context 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	<code>findAttributes</code> (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	<code>findAttributes</code> (java.lang.String packageName, boolean includeLiterals, boolean includeNonLiterals, boolean namesOrdered) Same as <code>findAttributes(String, boolean, boolean)</code> , but allows for ordering per name to be specified.
java.util.Collection	<code>findAttributesWithConstraints</code> () Returns attributes that have any kind of constraint (own and by class).
java.util.Map	<code>findAttributesWithDuplicates</code> (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	<code>findCimNoncimAssociations</code> () Returns all associations that are mappings between models of different natures.
java.util.Set	<code>findClasses</code> (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	<code>findClasses</code> (java.lang.String name) Returns all classes with name matching name.
java.util.Collection	<code>findClassesWithConstraints</code> () Returns all classes that have constraints.
java.util.Collection	<code>findDiagrams</code> (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	<code>findDOAttributes</code> () Returns attributes on logical nodes (not in meta-model).
java.util.Collection	<code>findMultivaluedAttributes</code> () Returns multi-valued attributes.
java.util.Collection	<code>findPackages</code> (java.util.List names) Returns potentially empty list of all packages whose name matches one of names.
java.util.Map	<code>findPresenceConditionLiterals</code> () Returns all literals that represent presence conditions.
java.util.Map	<code>getAbbreviationLiterals</code> () (IEC61850) Returns all abbreviation literal indexed by name.
java.util.Collection	<code>getAssociations</code> () Returns all associations in this model.
java.util.Collection	<code>getAttributes</code> () Returns all attributes in this model.
<code>Config</code>	<code>getCfg</code> () Returns configuration.

java.util.Collection	<u>getClasses()</u> Returns all classes in this model.
java.util.Collection	<u>getDependencies()</u> Returns all explicit (hand-drawn) dependencies in this model.
java.util.Collection	<u>getDiagrams()</u> Returns all diagrams in this model.
java.lang.String	<u>getModelNamesWithNature()</u> Returns comma-separated pairs {modelPackageName nature}.
java.util.Collection	<u>getModelPackages()</u> Returns all model packages.
java.util.Collection	<u>getModelPackages()</u> (java.util.EnumSet natures) Returns model packages of specified nature.
java.util.Collection	<u>getNamespaceInfos()</u> Returns non-null namespace informations for all packages.
java.util.Collection	<u>getNamespacePackages()</u> (java.util.EnumSet wgs) Returns packages that have namespace info for specified owners.
java.util.Map	<u>getNullClasses()</u> Returns special, "null" classes per nature.
java.util.Map	<u>getNullModelPackages()</u> 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	<u>getOperations()</u> Returns all operations in this model.
java.util.Collection	<u>getPackages()</u> Returns all packages in this model.
java.util.Map	<u>getSortedAbbreviatedTerms()</u> (IEC61850) Returns all abbreviated terms sorted by decreasing length; handles duplicate definitions by appending all of them per term.
java.util.Map	<u>getTags()</u>
java.util.Collection	<u>getTopPackages()</u> (java.util.EnumSet wgs) Returns top packages for specified owners.
java.lang.String	<u>getUuid()</u> Returns model UUID.
java.util.Collection	<u>getVersionInfos()</u> (java.util.EnumSet wgs) Returns version informations for top-level packages of specified owners.
java.lang.String	<u>toString()</u>

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)
```

Parameters:

profilesModel

(continued from last page)

getPackages

```
public java.util.Collection getPackages()
```

Returns all packages in this model.

findPackages

```
public java.util.Collection findPackages(java.util.List names)
```

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.

getSortedAbbreviatedTerms

```
public java.util.Map getSortedAbbreviatedTerms()
```

(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()
```

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:

`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.

(continued from last page)

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.

getDependencies

```
public java.util.Collection getDependencies()
```

Returns all explicit (hand-drawn) dependencies in this model.

(continued from last page)

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

getVersionInfos

```
public java.util.Collection getVersionInfos(java.util.EnumSet wgs)
```

Returns version informations for top-level packages of specified owners.

(continued from last page)

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	isMultivalued() 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

<code>static</code> UmlMultiplicity	parseFromString (java.lang.String mult) Returns multiplicity object from formatted string "lower..upper".
<code>java.lang.String</code>	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	<u>M</u> Mandatory.
public static final	<u>O</u> Optional.

Method Summary

static boolean	<u>isMember</u> (java.lang.String value)
static <u>UmlMultiplicity.Kind</u>	<u>valueOf</u> (java.lang.String name)
static <u>UmlMultiplicity.Kind[]</u>	<u>values</u> ()

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

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.
java.lang.String	getAlias () Returns the alias of this UML object, as defined in the UML model, empty string if not defined.
TextDescription	getDescription () Returns the raw text description for this UML object, as defined in the UML model.
TextDescription	getHtmlDescription () Returns the formatted description for this UML object, as defined in the UML model.
java.lang.Integer	getId () Returns the local ID of this UML object.
UmlKind	getKind () Returns kind of this UML object, as assigned by the application.
java.lang.String	getName () Returns the name of this UML object, as defined in the UML model.
Nature	getNature () Returns the nature of this UML object, which determines the validation rules to apply, and sometimes document generation formats.
OwningWg	getOwner () Returns the IEC working group owning this UML object, as calculated by the application based on the UML model structure.
java.util.Set	getPredefinedTagNames () Returns allowed tag names, as expected to be found in the UML model.
java.lang.String	getQualifiedName () Returns the name of this UML object combined with some container-related information (e.g., packageName.className, or containingPackageName.packageName).
java.lang.String	getSince () Returns the version of the model in which this UML object has been introduced, as defined in the UML model.

UmlStereotype	getStereotype() Returns the stereotype of this UML object, as defined in the UML model.
java.util.Map	getTaggedValues() Returns all the tagged values of this UML object, as defined in the UML model.
java.util.Set	getUnallowedTagNames() Returns actual tag names defined for this object, but not found in getPredefinedTagNames() .
java.lang.String	getUuid() Returns the universally unique ID of this UML object.
UmlVisibility	getVisibility() Returns visibility of this UML object, as defined in the UML model.
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).
boolean	isInformative() Returns whether this UML object is informative (and thus should be ignored when generating official IEC documents).
java.lang.String	toShortString (boolean includeId, boolean isNameQualified) Returns the context as string, for logging purposes:

Methods

getId

```
public 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 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 everything else) and when manipulating XML, 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 java.lang.String getSince()
```

Returns the version of the model in which this UML object has been introduced, as defined in the UML model.

(continued from last page)

getOwner

```
public OwningWg getOwner()
```

Returns the IEC working group owning this UML object, as calculated by the application based on the UML model structure.

getNature

```
public Nature getNature()
```

Returns the nature of this UML object, which determines the validation rules to apply, and sometimes document generation formats.

isInformative

```
public boolean isInformative()
```

Returns whether this UML object is informative (and thus should be ignored when generating official IEC documents).

getVisibility

```
public UmlVisibility getVisibility()
```

Returns visibility of this UML object, as defined in the UML model.

getKind

```
public UmlKind getKind()
```

Returns kind of this UML object, as assigned by the application.

getName

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

Returns the name of this UML object, as defined in the UML model.

getAlias

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

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 java.lang.String toShortString(boolean includeId,  
    boolean isNameQualified)
```

(continued from last page)

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 TextDescription getDescription()
```

Returns the raw text description for this UML object, as defined in the UML model. For formatted description, use [getHtmlDescription\(\)](#).

getHtmlDescription

```
public TextDescription getHtmlDescription()
```

Returns the formatted description for this UML object, as defined in the UML model. For raw text description, use [getDescription\(\)](#).

getStereotype

```
public UmlStereotype getStereotype()
```

Returns the stereotype of this UML object, as defined in the UML model.

isDeprecated

```
public 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 java.util.Set getPredefinedTagNames()
```

Returns allowed tag names, as expected to be found in the UML model.

getUnallowedTagNames

```
public java.util.Set getUnallowedTagNames()
```

Returns actual tag names defined for this object, but not found in [getPredefinedTagNames\(\)](#).

addTaggedValue

```
public 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.

(continued from last page)

Parameters:

name - tag name.

value - tag value.

Returns:

null if the name is a new tag, otherwise old value for name that has been overwritten with value.

Throws:

[InvalidTagException](#) - if either name or value is invalid.

getTaggedValues

```
public 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 from 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()
```

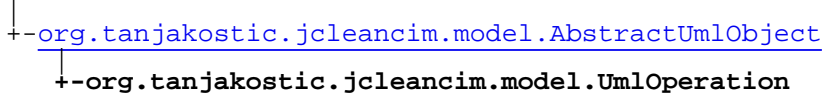
toString

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

(continued from last page)

org.tanjakostic.jcleancim.model Class UmlOperation

java.lang.Object



All Implemented Interfaces:

[UmlObject](#)

public class **UmlOperation**
extends [AbstractUmlObject](#)

UML operation.

Nested Class Summary

class	UmlOperation.Data UmlOperation.Data
class	UmlOperation.ReturnKind UmlOperation.ReturnKind

Field Summary

public static final	STEREOTYPES Allowed stereotypes for UML operations.
public static final	THROWS_TAG Exceptions in EA must be specified as tagged value with the tag name 'throws'. Value: throws

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 <code>exc</code> to this operation, and returns the same object.
UmlParameter	addParameter(UmlParameter par) Adds non-null parameter <code>par</code> to this operation, and returns the same object.
UmlClass	getContainingClass()
java.lang.String	getEaExceptionTypeInfo(int i)
int	getEaReturnTypeId()

java.lang.String	<u>getEaReturnTypeInfo()</u> 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	<u>getEaReturnTypeName()</u>
java.util.Set	<u>getEfferentClasses()</u> Returns all classes that I use as type for return value, parameter or exception.
java.util.List	<u>getExceptions()</u> Returns all exceptions declared for this operation.
java.lang.String	<u>getExceptionsSignature()</u> Returns potentially empty string containing comma-separated list of exceptions that follow the 'throw' statement in operation signature.
<u>UmlKind</u>	<u>getKind()</u>
static java.util.List	<u>getKinds()</u> (<u>Nature</u> nature) Returns all available classifications (kinds) for operations.
<u>Nature</u>	<u>getNature()</u>
<u>OwningWg</u>	<u>getOwner()</u>
java.util.List	<u>getParameters()</u> Returns all parameters of this operation.
java.util.Set	<u>getPredefinedTagNames()</u>
java.lang.String	<u>getQualifiedName()</u>
<u>UmlClass</u>	<u>getReturnType()</u> Returns return type of this operation, null if kind is <u>UmlOperation.ReturnKind.OP_RET_VOID</u> .
java.lang.String	<u>getSignature()</u> E.g.
boolean	<u>isAbstract()</u>
boolean	<u>isFinal()</u>
boolean	<u>isInformative()</u>
boolean	<u>isStatic()</u>
boolean	<u>isVoidReturned()</u>
java.lang.String	<u>toString()</u>

Methods inherited from class [org.tanjakostic.jcleancim.model.AbstractUmlObject](#)

[addTaggedValue](#), [classifyPerScope](#), [classifyPerScopePerTag](#), [classifyPerTag](#), [collectDuplicateDescriptions](#), [collectDuplicateNames](#), [collectForScope](#), [collectNames](#), [collectQNames](#), [findWithSameUuidAndLog](#), [getAlias](#), [getDescription](#), [getHtmlDescription](#), [getId](#), [getKind](#), [getName](#), [getNature](#), [getOwner](#), [getPredefinedTagNames](#), [getQualifiedName](#), [getSince](#), [getStereotype](#), [getTaggedValues](#), [getUnallowedTagNames](#), [getUuid](#), [getVisibility](#), [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](#), [getNature](#), [getOwner](#), [getPredefinedTagNames](#), [getQualifiedName](#), [getSince](#), [getStereotype](#), [getTaggedValues](#), [getUnallowedTagNames](#), [getUuid](#), [getVisibility](#), [isDeprecated](#), [isInformative](#), [toShortString](#)

Fields

STEREOTYPES

public static final `java.util.List` **STEREOTYPES**

Allowed stereotypes for UML operations.

THROWS_TAG

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

Exceptions in EA must be specified as tagged value with the tag name 'throws'.
Constant value: **throws**

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**()

(continued from last page)

isStatic

```
public boolean isStatic()
```

isFinal

```
public boolean isFinal()
```

isVoidReturned

```
public boolean isVoidReturned()
```

getEaReturnTypeId

```
public int getEaReturnTypeId()
```

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.

(continued from last page)

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.

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()
```

getNature

```
public Nature getNature()
```

isInformative

```
public boolean isInformative()
```

getKind

```
public UmlKind getKind()
```

(continued from last page)

getQualifiedName

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

getPredefinedTagNames

```
public java.util.Set getPredefinedTagNames()
```

This default implementation returns empty set, and should be overridden by subclasses that have something to return.

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	UmlOperation.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)

UmlOperation.Data

```
public UmlOperation.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

Field Summary

public static final	STEREOTYPES Allowed stereotypes for UML attributes.
---------------------	--

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.
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.
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).
java.lang.String	getExpectedNamespaceClassName () Returns the expected name for the namespace class, as per IEC TC57 UML models rules.
java.lang.String	getExpectedVersionClassName () Returns the expected name for the version class, as per IEC TC57 UML models rules.
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.
NamespaceInfo	getNamespaceInfo () 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.lang.String	getQualifiedName ()
VersionInfo	getVersionInfo () Returns the version information in case the relevant version class is defined in the package, null otherwise.
boolean	isInformative ()
boolean	isInOrUnderPackage (java.lang.String packageName) Returns whether this package or anywhere below it (recursively) there is a package called packageName.

boolean	<u>isTop()</u> Returns whether this package is a top package (owned by a WG).
boolean	<u>isUnderPackage</u> (java.lang.String packageName) Returns whether anywhere below this package (recursively) there is a package called packageName.
void	<u>orderClasses</u> (java.util.List uuids) Orders classes in the order given in uuids.
boolean	<u>shouldExportDiagrams</u> ()
java.lang.String	<u>toString</u> ()

Methods inherited from class [org.tanjakostic.jcleancim.model.UmlStructure](#)

[addDependency](#), [addDiagram](#), [addSkippedUmlItem](#), [collectDependencyAfferentStructures](#), [collectDependencyEfferentStructures](#), [getDependenciesAsSource](#), [getDependenciesAsTarget](#), [getDiagrams](#), [getKind](#), [getModel](#), [getNature](#), [getOwner](#), [getQualifiedName](#), [getSkippedUmlItems](#), [isInformative](#), [isSelfDependent](#)

Methods inherited from class [org.tanjakostic.jcleancim.model.AbstractUmlObject](#)

[addTaggedValue](#), [classifyPerScope](#), [classifyPerScopePerTag](#), [classifyPerTag](#), [collectDuplicateDescriptions](#), [collectDuplicateNames](#), [collectForScope](#), [collectNames](#), [collectQNames](#), [findWithSameUuidAndLog](#), [getAlias](#), [getDescription](#), [getHtmlDescription](#), [getId](#), [getKind](#), [getName](#), [getNature](#), [getOwner](#), [getPredefinedTagNames](#), [getQualifiedName](#), [getSince](#), [getStereotype](#), [getTaggedValues](#), [getUnallowedTagNames](#), [getUuid](#), [getVisibility](#), [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](#), [getNature](#), [getOwner](#), [getPredefinedTagNames](#), [getQualifiedName](#), [getSince](#), [getStereotype](#), [getTaggedValues](#), [getUnallowedTagNames](#), [getUuid](#), [getVisibility](#), [isDeprecated](#), [isInformative](#), [toShortString](#)

Fields

STEREOTYPES

```
public static final java.util.List STEREOTYPES
```

Allowed stereotypes for UML attributes.

Constructors

(continued from last page)

UmlPackage

```
public UmlPackage(UmlModel model,  
                 UmlObjectData objData,  
                 UmlPackage.Data data)
```

Creates a model package (e.g. TC57CIM 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 [UmlObjects](#).
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.

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.

getVersionInfo

```
public VersionInfo getVersionInfo()
```

Returns the version information in case the relevant version class is defined in the package, null otherwise.

getExpectedVersionClassName

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

Returns the expected name for the version class, as per IEC TC57 UMI models rules.

getNamespaceInfo

```
public NamespaceInfo getNamespaceInfo()
```

Returns the namespace information in case the relevant namespace class (for IEC61850) or version class (for CIM) is defined in the package, null otherwise.

getExpectedNamespaceClassName

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

Returns the expected name for the namespace class, as per IEC TC57 UMI models rules.

getChildPackages

```
public java.util.Set getChildPackages()
```

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.

getOwner

```
public OwningWg getOwner()
```

getNature

```
public Nature getNature()
```

isInformative

```
public boolean isInformative()
```

(continued from last page)

A package is considered as informative if any of the following is true:

- 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()
```

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	UmlPackage.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

UmlPackage.Data

public **UmlPackage.Data**([UmlStructure.Data](#) data)

Constructor.

Methods

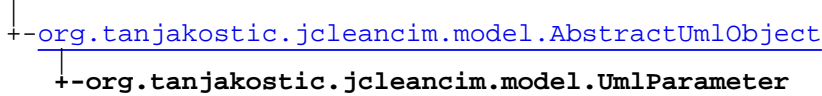
empty

public static [UmlPackage.Data](#) **empty**()

Returns an empty instance.

org.tanjakostic.jcleancim.model Class UmlParameter

java.lang.Object



All Implemented Interfaces:

[UmlObject](#)

public class **UmlParameter**
extends [AbstractUmlObject](#)

Operation parameter.

Nested Class Summary

class	UmlParameter.Data UmlParameter.Data
class	UmlParameter.Kind UmlParameter.Kind

Field Summary

public static final	STEREOTYPES Allowed stereotypes for UML operation parameters.
---------------------	--

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 ()
Nature	getNature ()
OwningWg	getOwner ()

java.lang.String	getQualifiedName()
java.lang.String	getSignature() E.g., "MyClass[] myArg".
UmlClass	getType()
boolean	isArray()
boolean	isInformative()
java.lang.String	toString()

Methods inherited from class [org.tanjakostic.jcleancim.model.AbstractUmlObject](#)

[addTaggedValue](#), [classifyPerScope](#), [classifyPerScopePerTag](#), [classifyPerTag](#), [collectDuplicateDescriptions](#), [collectDuplicateNames](#), [collectForScope](#), [collectNames](#), [collectQNames](#), [findWithSameUuidAndLog](#), [getAlias](#), [getDescription](#), [getHtmlDescription](#), [getId](#), [getKind](#), [getName](#), [getNature](#), [getOwner](#), [getPredefinedTagNames](#), [getQualifiedName](#), [getSince](#), [getStereotype](#), [getTaggedValues](#), [getUnallowedTagNames](#), [getUuid](#), [getVisibility](#), [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](#), [getNature](#), [getOwner](#), [getPredefinedTagNames](#), [getQualifiedName](#), [getSince](#), [getStereotype](#), [getTaggedValues](#), [getUnallowedTagNames](#), [getUuid](#), [getVisibility](#), [isDeprecated](#), [isInformative](#), [toShortString](#)

Fields

STEREOTYPES

```
public static final java.util.List STEREOTYPES
```

Allowed stereotypes for UML operation parameters.

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.

(continued from last page)

Parameters:

type
objData
data

Throws:

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.

getNature

```
public Nature getNature()
```

Returns nature, null if parameter has not yet been added to its operation.

(continued from last page)

isInformative

```
public boolean isInformative()
```

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()
```

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	UmlParameter.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

UmlParameter.Data

```
public UmlParameter.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_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()
Nature	getNature()
java.lang.String	getOtherEndName() Returns name of the other end if this skipped element is a connector, null otherwise.
OwningWg	getOwner()
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](#)

[addTaggedValue](#), [classifyPerScope](#), [classifyPerScopePerTag](#), [classifyPerTag](#), [collectDuplicateDescriptions](#), [collectDuplicateNames](#), [collectForScope](#), [collectNames](#), [collectQNames](#), [findWithSameUuidAndLog](#), [getAlias](#), [getDescription](#), [getHtmlDescription](#), [getId](#), [getKind](#), [getName](#), [getNature](#), [getOwner](#), [getPredefinedTagNames](#), [getQualifiedName](#), [getSince](#), [getStereotype](#), [getTaggedValues](#), [getUnallowedTagNames](#), [getUuid](#), [getVisibility](#), [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](#), [getNature](#), [getOwner](#), [getPredefinedTagNames](#), [getQualifiedName](#), [getSince](#), [getStereotype](#), [getTaggedValues](#), [getUnallowedTagNames](#), [getUuid](#), [getVisibility](#), [isDeprecated](#), [isInformative](#), [toShortString](#)

Fields

(continued from last page)

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_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.

(continued from last page)

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()
```

getNature

```
public Nature getNature()
```

isInformative

```
public boolean isInformative()
```

getKind

```
public UmlKind getKind()
```

getQualifiedName

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

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	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 ()

<code>static UmlSkipped.Kind</code>	<code>valueOf(java.lang.String name)</code>
<code>static UmlSkipped.Kind[]</code>	<code>values()</code>

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
```

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
```

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()
```

(continued from last page)

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	UmlSkipped.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

UmlSkipped.Data

```
public UmlSkipped.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: At present, we don't check for duplicates, and no 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	CIM_STEREOTYPES
public static final	CIMDATATYPE Value: CIMDatatype
public static final	COMPOUND Value: Compound
public static final	COND Used for enumerations that represent presence conditions (constraints). Value: cond
public static final	DEPRECATED UML stereotype for deprecated UML elements or connectors. 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	IEC61850_STEREOTYPES
public static final	INFORMATIVE UML stereotype for non-normative associations. Value: informative
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).
java.util.Set	getMissingTokens (java.util.Set tokens) Returns all the tokens of this instance that are not 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**

UML stereotype for deprecated UML elements or connectors.
Constant value: **deprecated**

INFORMATIVE

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

UML stereotype for non-normative associations.
Constant value: **informative**

PRIMITIVE

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

Constant value: **Primitive**

OLD_DATATYPE

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

Constant value: **Datatype**

(continued from last page)

CIMDATATYPE

```
public static final java.lang.String CIMDATATYPE
```

Constant value: **CIMDatatype**

COMPOUND

```
public static final java.lang.String COMPOUND
```

Constant value: **Compound**

CIM_STEREOTYPES

```
public static final java.util.List CIM_STEREOTYPES
```

EA_INTERFACE

```
public static final java.lang.String EA_INTERFACE
```

Constant value: **interface**

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
```

(continued from last page)

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 (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**

IEC61850_STEREOTYPES

```
public static final java.util.List IEC61850_STEREOTYPES
```

Constructors

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

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).

(continued from last page)

memberOf

```
public boolean memberOf(java.util.Set tokens)
```

Returns whether all the tokens of this instance are contained in tokens.

getMissingTokens

```
public java.util.Set getMissingTokens(java.util.Set tokens)
```

Returns all the tokens of this instance that are not 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 depend on through an explicit UML dependency in the model.

java.util.Set	<u>getDependenciesAsSource()</u> Returns all explicit (hand-drawn) UML dependencies where I am source.
java.util.Set	<u>getDependenciesAsTarget()</u> Returns all explicit (hand-drawn) UML dependencies where I am target.
java.util.Set	<u>getDiagrams()</u> Returns all diagrams in this structure.
abstract <u>UmlKind</u>	<u>getKind()</u>
abstract <u>UmlModel</u>	<u>getModel()</u> Returns the model this structure belongs to.
abstract <u>Nature</u>	<u>getNature()</u>
abstract <u>OwningWg</u>	<u>getOwner()</u>
abstract java.lang.String	<u>getQualifiedName()</u>
java.util.Set	<u>getSkippedUmlItems()</u> Returns all skipped UML items: elements within or connectors with this structure.
abstract boolean	<u>isInformative()</u>
boolean	<u>isSelfDependent()</u>

Methods inherited from class [org.tanjakostic.jcleancim.model.AbstractUmlObject](#)

[addTaggedValue](#), [classifyPerScope](#), [classifyPerScopePerTag](#), [classifyPerTag](#), [collectDuplicateDescriptions](#), [collectDuplicateNames](#), [collectForScope](#), [collectNames](#), [collectQNames](#), [findWithSameUuidAndLog](#), [getAlias](#), [getDescription](#), [getHtmlDescription](#), [getId](#), [getKind](#), [getName](#), [getNature](#), [getOwner](#), [getPredefinedTagNames](#), [getQualifiedName](#), [getSince](#), [getStereotype](#), [getTaggedValues](#), [getUnallowedTagNames](#), [getUuid](#), [getVisibility](#), [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](#), [getNature](#), [getOwner](#), [getPredefinedTagNames](#), [getQualifiedName](#), [getSince](#), [getStereotype](#), [getTaggedValues](#), [getUnallowedTagNames](#), [getUuid](#), [getVisibility](#), [isDeprecated](#), [isInformative](#), [toShortString](#)

Constructors

(continued from last page)

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()
```

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 depend on through an explicit UML dependency in the model.

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.

getOwner

```
public abstract OwningWg getOwner()
```

getNature

```
public abstract Nature getNature()
```

isInformative

```
public abstract boolean isInformative()
```

(continued from last page)

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	UmlStructure.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

UmlStructure.Data

public **UmlStructure.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 initial 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 initial 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 (java.lang.String version, java.lang.String date)
--------	---

Method Summary

java.lang.String	getDate ()
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**(java.lang.String version,
java.lang.String date)

Methods

getVersion

public java.lang.String **getVersion**()

getDate

public java.lang.String **getDate**()

(continued from last page)

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	<u>getScopedTags()</u>
java.util.Map	<u>getStatsPerNature()</u>
int	<u>getTagNamesCount()</u>
void	<u>logAbbreviatedTermUsage</u> (org.apache.log4j.Level level) (IEC61850) For every abbreviated term, logs DOs using it; opposite to <u>logDONameDecomposition(Level)</u> .
void	<u>logAggregationsWithWgClasses</u> (org.apache.log4j.Level level, <u>OwningWg</u> wg)
void	<u>logAttributesWithConstraints</u> (org.apache.log4j.Level level)
void	<u>logCimNoncimAssociations</u> (org.apache.log4j.Level level)
void	<u>logClasses</u> (org.apache.log4j.Level level)
void	<u>logClassesWithAttributeConstraints</u> (org.apache.log4j.Level level)
void	<u>logDONameDecomposition</u> (org.apache.log4j.Level level) (IEC61850) For every DO, logs abbreviations used; opposite to <u>logAbbreviatedTermUsage(Level)</u> .
void	<u>logMultivaluedAttributes</u> (org.apache.log4j.Level level)
void	<u>logNamespaceInfos</u> (org.apache.log4j.Level level)
void	<u>logNormativeAssociationsWithWgClasses</u> (org.apache.log4j.Level level, <u>OwningWg</u> wg)
void	<u>logNormativeClasses</u> (org.apache.log4j.Level level, java.util.EnumSet wgs)
void	<u>logOperations</u> (org.apache.log4j.Level level)
void	<u>logPackages</u> (org.apache.log4j.Level level)
void	<u>logStats()</u>
void	<u>logTaggedValues</u> (org.apache.log4j.Level level)
void	<u>logVersionInfos</u> (org.apache.log4j.Level level)
java.lang.String	<u>toHtml()</u>
java.lang.String	<u>toString()</u>

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)
```

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)
```

(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()
```

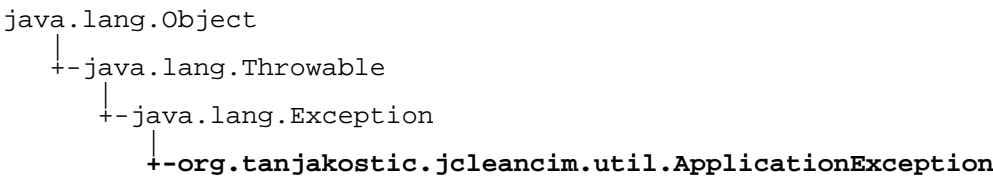
toString

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

Package

org.tanjakostic.jcleancim.util

org.tanjakostic.jcleancim.util Class ApplicationException



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

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

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

```
imageUpdate
```

```
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: </i>
public static final	I_START Value: <i>
public static final	LI_END Value:
public static final	LI_START Value:
public static final	LIST_TAGS
public static final	NBSP Value: &nbsp;

public static final	OL_END Value: <code></code>
public static final	OL_START Value: <code></code>
public static final	P_END Value: <code></p></code>
public static final	P_START Value: <code><p></code>
public static final	SUB_END Value: <code></sub></code>
public static final	SUB_START Value: <code><sub></code>
public static final	SUP_END Value: <code></sup></code>
public static final	SUP_START Value: <code><sup></code>
public static final	U_END Value: <code></u></code>
public static final	U_START Value: <code><u></code>
public static final	UL_END Value: <code></code>
public static final	UL_START Value: <code></code>

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_END

```
public static final java.lang.String UL_END
```

Constant value: ``

OL_START

```
public static final java.lang.String OL_START
```

Constant value: ``

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_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: ``

(continued from last page)

B_END

```
public static final java.lang.String B_END
```

Constant value: ``

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: ``

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>`

P_END

```
public static final java.lang.String P_END
```

Constant value: `</p>`

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 calculated 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 calculated 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 calculated 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 calculated 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 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_DELIMITER 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	<u>capitalise</u> (java.lang.String input) Returns string starting with upper-case letter, all the other letters lower-case.
static void	<u>clearClipboard</u> () Clears system clipboard.
static java.lang.String	<u>concatCharSeparatedTokens</u> (java.lang.String separator, boolean delimitTokens, java.util.List tokens) Concatenates tokens with the separator string between consecutive ones, and returns the resulting string.
static void	<u>copy</u> (java.io.File src, java.io.File dst) Copies src file to dst file.
static void	<u>copyHtmlToClipboard</u> (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	<u>copyImageToClipboard</u> (java.io.File pic) Copies image in pic to clipboard.
static void	<u>copyTextToClipboard</u> (java.lang.String txt) Copies non-empty, non-null txt to clipboard, no-op otherwise.
static java.util.Map	<u>createKeyValuePair</u> (java.lang.Object key, java.lang.Object value) Retruns the map with a single key/value pair.
static java.io.File	<u>createTempImageFile</u> (java.lang.String dirAbsPath, java.lang.String fileName, <u>Util.ImageFormat</u> format, boolean deleteOnExit) Creates file in the given directory (or in default OS tmp directory) and returns the result.
static void	<u>delete</u> (java.io.File f) Wrapper for the File.delete() that accepts null argument and returns nothing.
static void	<u>ensureContainsNotNull</u> (java.util.Collection arg, java.lang.String name)
static void	<u>ensureContainsNotNull</u> (java.lang.Object[] arg, java.lang.String name)
static void	<u>ensureNotEmpty</u> (boolean[] arg, java.lang.String name)
static void	<u>ensureNotEmpty</u> (java.util.Collection arg, java.lang.String name)
static void	<u>ensureNotEmpty</u> (int[] arg, java.lang.String name)
static void	<u>ensureNotEmpty</u> (java.util.Map arg, java.lang.String name)
static void	<u>ensureNotEmpty</u> (java.lang.Object[] arg, java.lang.String name)
static void	<u>ensureNotEmpty</u> (java.lang.String arg, java.lang.String name)
static void	<u>ensureNotNull</u> (java.lang.Object arg, java.lang.String name)

static java.lang.String	<code>fetchTextFromClipboard()</code> Returns text contained in the clipboard (text could be plain or markup), null if clipboard is empty.
static java.lang.String	<code>fillString(int count, char ch)</code> Returns the string filled with number count of characters c.
static java.io.InputStream	<code>findResourceOnClasspath(java.lang.String resourceName)</code> Returns resource as input stream for its name, given that it is found on the classpath.
static java.lang.String	<code>formatDuration(long millis)</code>
static java.io.File	<code>getDirectory(java.lang.String dirRelPath, boolean createIfMissing)</code> Returns file representing directory dirName under #USER_DIR_KEY.
static java.lang.String	<code>getFileExtension(java.lang.String filePath)</code> Returns extension (after the last ".") if being part of filePath, null otherwise.
static java.lang.String	<code>getFileExtensionWithDot(java.lang.String filePath)</code> Returns extension with the "." if being part of filePath, null otherwise.
static java.lang.String	<code>getIndentSpaces(int count)</code> Returns string of spaces of the size equal to count * <code>INDENT_COUNT</code> .
static java.lang.Object	<code>getKeyByValue(java.util.Map map, java.lang.Object value)</code> FIXME: tests
static java.util.Set	<code>getKeysByValue(java.util.Map map, java.lang.Object value)</code> FIXME: tests
static java.lang.String	<code>getNonBreakingSpaces(int count)</code> Returns string of count non-breaking spaces.
static java.io.File	<code>getOutputFileRenameIfExists(java.lang.String outDirName, java.lang.String outFileName)</code> Returns file #USER_DIR_KEY/outDirName/outFileName.
static java.lang.String	<code>getResourceAbsPath(java.lang.String resourceName, java.lang.String detail)</code> Returns absolute path of the resource found on the classpath.
static boolean	<code>hasContent(java.lang.String value)</code> Returns true is value is not null, and the trimmed content is not empty.
static java.util.Properties	<code>initPropsFromFile(java.lang.String propsFileName)</code> 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	<code>listFiles(java.io.File directory, java.io FilenameFilter filter, boolean recurse)</code> Returns potentially empty list of files under directory (and its sub-directories if recurse is true), filtered with filter.
static void	<code>logCollection(org.apache.log4j.Level level, java.util.Collection objects, java.lang.String what)</code> Logs each element in objects.
static void	<code>logCompletion(org.apache.log4j.Level level, java.lang.String text, long startMillis, boolean skipTime)</code> Logs text with level (and if skipTime=false, duration since startMillis).

static void	<code>logMap</code> (org.apache.log4j.Level level, java.util.Map objects, java.lang.String what) Logs each element in objects.
static void	<code>logSubtitle</code> (org.apache.log4j.Level level, java.lang.String subtitle) Logs subtitle with level (for sub-steps in the application).
static void	<code>logTitle</code> (org.apache.log4j.Level level, java.lang.String[] title) Logs title with level (for major steps in the application).
static boolean	<code>looksLikePlural</code> (java.lang.String token) Returns whether token looks like plural; returns false for null or empty arg.
static java.lang.String	<code>null2empty</code> (java.lang.String s) Returns empty string if s is null, s otherwise.
static java.lang.Integer	<code>parseInt</code> (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	<code>parseIntZero</code> (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	<code>saveImageFromClipboard</code> (java.io.File pic)
static java.io.File	<code>saveToFile</code> (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	<code>sortByDecreasingLength</code> (java.util.Map items) Returns copy of items sorted by decreasing length of keys (longest first).
static java.lang.String[]	<code>sortByDecreasingLength</code> (java.lang.String[] items) Returns copy of items sorted by decreasing length (longest first).
static java.util.List	<code>splitCharSeparatedTokens</code> (java.lang.String input, char c) Splits c-separated string into a list of non-empty tokens.
static java.util.List	<code>splitCommaSeparatedTokens</code> (java.lang.String input) Splits comma-separated string into a list of non-empty tokens.
static java.util.List	<code>splitDirAndFileNames</code> (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	<code>splitLines</code> (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	<code>splitStringSeparatedTokens</code> (java.lang.String input, java.lang.String separator) Splits c-separated string into a list of non-empty tokens.
static java.lang.String	<code>truncateEnd</code> (java.lang.String input) Identical to <code>truncateEnd(String, int)</code> with default value for charCount = #TRUNCATE_GREATER_THAN.
static java.lang.String	<code>truncateEnd</code> (java.lang.String input, int charCount) Truncates input to first charCount characters and appends "...".

<code>static java.lang.String</code>	<code>truncateStart(java.lang.String input)</code> Identical to <code>truncateEnd(String, int)</code> with default value for charCount = <code>#TRUNCATE_GREATER_THAN</code> .
<code>static java.lang.String</code>	<code>truncateStart(java.lang.String input, int charCount)</code> 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
```

Constant value: **user.dir**

USER_DIR

```
public static final java.lang.String USER_DIR
```

TRUNCATE_GREATER_THAN

```
public static final int TRUNCATE_GREATER_THAN
```

(continued from last page)

Constant value: **30**

TOKEN_DELIMITER

```
public static final char TOKEN_DELIMITER
```

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

(continued from last page)

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,
    boolean delimitTokens,
    java.util.List tokens)
```

Concatenates tokens with the separator string between consecutive ones, 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. comma separated file format, or for logging and debugging with any desired separation (e.g. " | ", " / ").

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_DELIMITER 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)
```

(continued from last page)

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](#).

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 if 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 input 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.

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.

(continued from last page)

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

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:

(continued from last page)

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
```

(continued from last page)

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.

Parameters:

`dirAbsPath` - absolute path of the file; if null, temporary directory is used.

`fileName` - name of the file (without 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:

(continued from last page)

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)
```

(continued from last page)

ensureNotEmpty

```
public static void ensureNotEmpty(java.util.Collection arg,  
    java.lang.String name)
```

ensureNotEmpty

```
public static void ensureNotEmpty(boolean[] arg,  
    java.lang.String name)
```

ensureNotEmpty

```
public static void ensureNotEmpty(java.lang.Object[] arg,  
    java.lang.String name)
```

ensureContainsNotNull

```
public static void ensureContainsNotNull(java.lang.Object[] arg,  
    java.lang.String name)
```

ensureContainsNotNull

```
public static void ensureContainsNotNull(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).

(continued from last page)

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.

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)
```

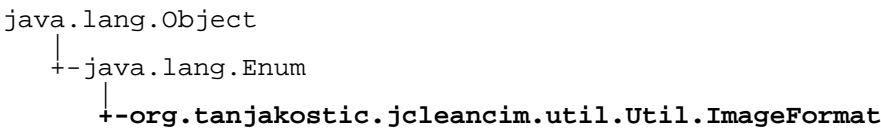
Logs each element in objects.

Parameters:

level - logging level.
objects - objects to log.
what - title to print when objects is not empty.

org.tanjakostic.jcleancim.util

Class Util.ImageFormat



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](#), [AttributesWhoseTypesInformative](#), [CimAttributesThatShouldBeReplacedWithAssociation](#), [AttributesThatAreEnumsInNonEnumeratedClass](#), [AttributesThatAreConstNonStatic](#), [CimAttributesThatAreNotStaticNonConstWithInitVal](#), [AttributesThatAreStaticButNotConst](#), [CimAttributesThatShouldBePublic](#), [AttributesWithTypeIdMismatch](#), [AttributesWithInvalidTypeString](#), [AttributesWithInvalidTypeNull](#), [CimAttributesThatShouldBeOptional](#), [AttributesWithInvalidMultiplicity](#), [EnumLiteralsWithoutEnumStereotype](#), [EnumLiteralsWithSuperfluousType](#), [CimClassesNeverUsedAsTypeForAttribute](#), [ClassesWithSameName](#), [EnumClassesWithDuplicateCodes](#), [EnumClassesWithSomeCodesMissing](#), [Iec61850LNClassesMalformedName](#), [Iec61850LNClassesInWrongGroup](#), [CimClassesNameShouldBeSingular](#), [CimClassesNameStartingWithLowerCase](#), [CimDatatypeClassesWithInvalidAttributes](#), [Iec61850ClassesWithMissingCondIDTextInConstraints](#), [Iec61850ClassesWithInvalidConstraints](#), [CimClassesNeverUsedInRelationships](#), [Iec61850ClassesThatShouldHaveTaggedValuesForDocgen](#), [Iec61850ClassesThatShouldHaveAliasAsTitle](#), [ClassesThatShouldNotHaveNestingThroughAttribute](#), [CimClassesThatShouldNotHaveExplicitDependencies](#), [CimClassesThatShouldNotHaveOperations](#), [CimClassesThatShouldNotBeAbstract](#), [CimClassesUsedForAttributesButHaveSuperclasses](#), [CimClassesUsedForAttributesButHaveSubclasses](#), [CimClassesUsedForAttributesButHaveAssociations](#), [CimClassesWithOldDatatypeStereotype](#), [ClassesWithUnallowedStereotype](#), [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	AbstractRule.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

AbstractRule.AbstractRuleWithSubobjectsAndSkips

```
protected AbstractRule.AbstractRuleWithSubobjectsAndSkips(org.apache.log4j.Logger  
extLogger,  
level,  
org.apache.log4j.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)
```

(continued from last page)

getSubObjects

```
protected 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.

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](#), [DependenciesWithUnallowedStereotype](#), [DiagramsWithUnallowedStereotype](#),
[OperationParametersWithUnallowedStereotype](#), [OperationsWithUnallowedStereotype](#),
[PackagesWithUnallowedStereotype](#)

public static abstract class **AbstractRule.UmlObjectsWithUnallowedStereotype**
extends [AbstractRule.AbstractRuleWithSubobjectsAndSkips](#)

Constructor Summary

protected

[AbstractRule.UmlObjectsWithUnallowedStereotype](#)(org.apache.log4j.Logger logger, java.lang.String what, java.util.Set 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

AbstractRule.UmlObjectsWithUnallowedStereotype

```
protected AbstractRule.UmlObjectsWithUnallowedStereotype(org.apache.log4j.Logger
logger,
                                                    java.lang.String what,
                                                    java.util.Set 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.

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	AbstractRule.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

AbstractRule.UmlObjectsWithUnallowedTagNames

```
public AbstractRule.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	AbstractRule.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

AbstractRule.UmlObjectsMissingDoc

```
protected AbstractRule.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](#)

Field Summary

public static final	DOCSTART_CHARS
---------------------	--------------------------------

Constructor Summary

protected	AbstractRule.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
```

Fields

DOCSTART_CHARS

```
public static final java.util.List DOCSTART_CHARS
```

Constructors

AbstractRule.UmlObjectsWithBadDocStart

```
protected AbstractRule.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	AbstractRule.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](#)

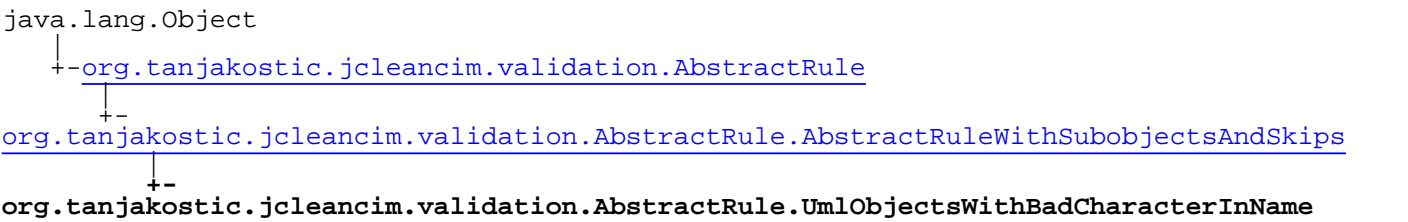
[validate](#)

AbstractRule.UmlObjectsWithBadDocEnd

Methods

Matches objects with non-empty description ending with a character different than ' '.

org.tanjakostic.jcleancim.validation
Class 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	AbstractRule.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

AbstractRule.UmlObjectsWithBadCharacterInName

```
protected AbstractRule.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.isValidationsAssociationsOn\(\)](#)), 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 visit 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	AssociationValidator.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

AssociationValidator.AssociationsWithExplicitDirection

```
public AssociationValidator.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	AssociationValidator.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

AssociationValidator.AssociationsWithRoleBadDirection

```
public AssociationValidator.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	AssociationValidator.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

AssociationValidator.AssociationsWithDoc

```
public AssociationValidator.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	AssociationValidator.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

AssociationValidator.AssociationsWithSameDocOnBothEnds

```
public AssociationValidator.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	AssociationValidator.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

AssociationValidator.AssociationsWithName

```
public AssociationValidator.AssociationsWithName()
```

Methods

getApplicability

```
public java.util.EnumSet getApplicability()
```

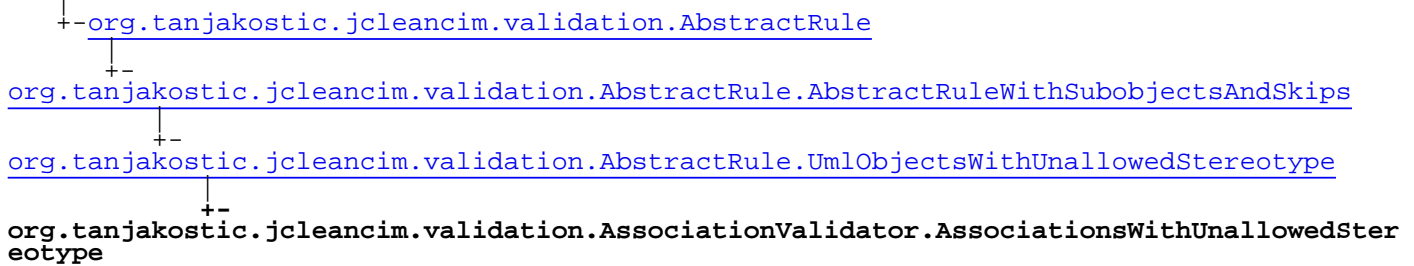
validate

```
public void validate(UmlAssociation o,  
    ModelIssues issues)
```

org.tanjakostic.jcleancim.validation

Class AssociationValidator.AssociationsWithUnallowedStereotype

java.lang.Object



All Implemented Interfaces:

Rule, [SimpleRule](#)

public static class **AssociationValidator.AssociationsWithUnallowedStereotype**
 extends [AbstractRule.UmlObjectsWithUnallowedStereotype](#)

Constructor Summary

public	AssociationValidator.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

AssociationValidator.AssociationsWithUnallowedStereotype

```
public AssociationValidator.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	AssociationValidator.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

AssociationValidator.AssociationEndsWithUnallowedStereotype

```
public AssociationValidator.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	AssociationValidator.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

AssociationValidator.AssociationsMissingInformativeStereotype

```
public AssociationValidator.AssociationsMissingInformativeStereotype()
```

Methods

getApplicability

```
public java.util.EnumSet getApplicability()
```

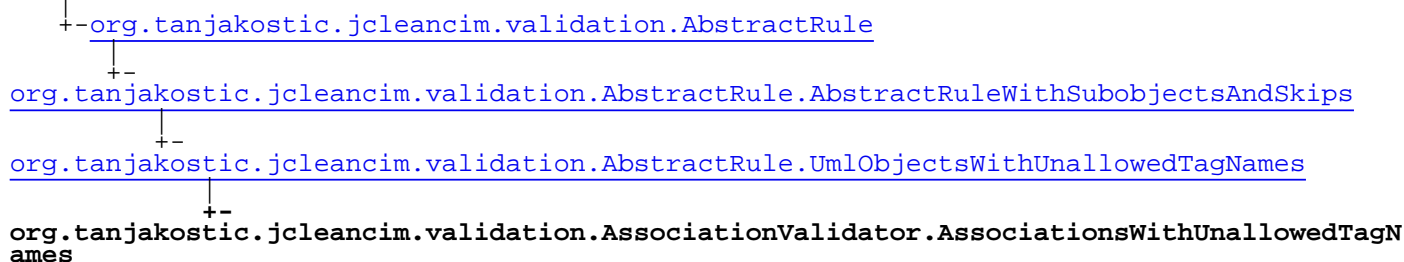
validate

```
public void validate(UmlAssociation o,  
    ModelIssues issues)
```

org.tanjakostic.jcleancim.validation

Class AssociationValidator.AssociationsWithUnallowedTagNames

java.lang.Object



All Implemented Interfaces:

Rule, [SimpleRule](#)

public static class **AssociationValidator.AssociationsWithUnallowedTagNames**
 extends [AbstractRule.UmlObjectsWithUnallowedTagNames](#)

Constructor Summary

public	AssociationValidator.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

AssociationValidator.AssociationsWithUnallowedTagNames

```
public AssociationValidator.AssociationsWithUnallowedTagNames()
```

Methods

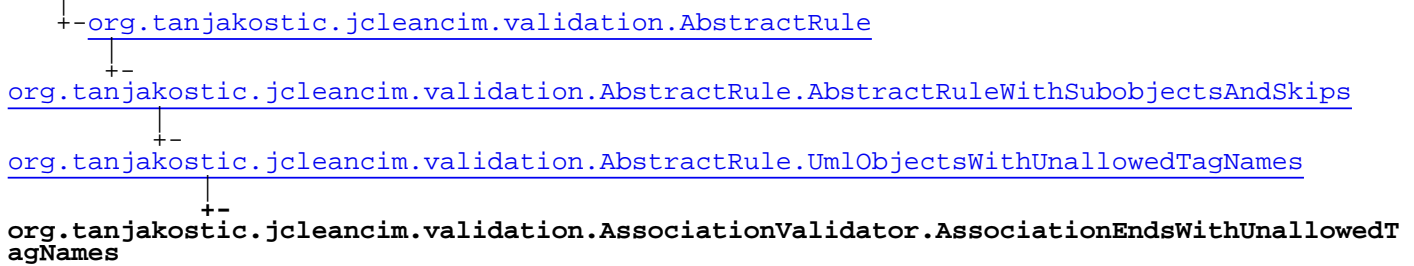
getApplicability

```
public java.util.EnumSet getApplicability()
```

org.tanjakostic.jcleancim.validation

Class AssociationValidator.AssociationEndsWithUnallowedTagNames

java.lang.Object



All Implemented Interfaces:

Rule, [SimpleRule](#)

public static class **AssociationValidator.AssociationEndsWithUnallowedTagNames**
 extends [AbstractRule.UmlObjectsWithUnallowedTagNames](#)

Constructor Summary

public	AssociationValidator.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

AssociationValidator.AssociationEndsWithUnallowedTagNames

```
public AssociationValidator.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	AssociationValidator.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

AssociationValidator.AssociationsWithNoMultiplicity

```
public AssociationValidator.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	AssociationValidator.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

AssociationValidator.AssociationsWithWrongSource

```
public AssociationValidator.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

[AssociationValidator.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

AssociationValidator.Iec61850AssociationsThatShouldBePrivate

```
public AssociationValidator.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	AssociationValidator.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

AssociationValidator.Iec61850AssociationsWithDifferentEndVisibility

```
public AssociationValidator.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	AssociationValidator.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

AssociationValidator.AssociationEndsMissingDoc

```
public AssociationValidator.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.

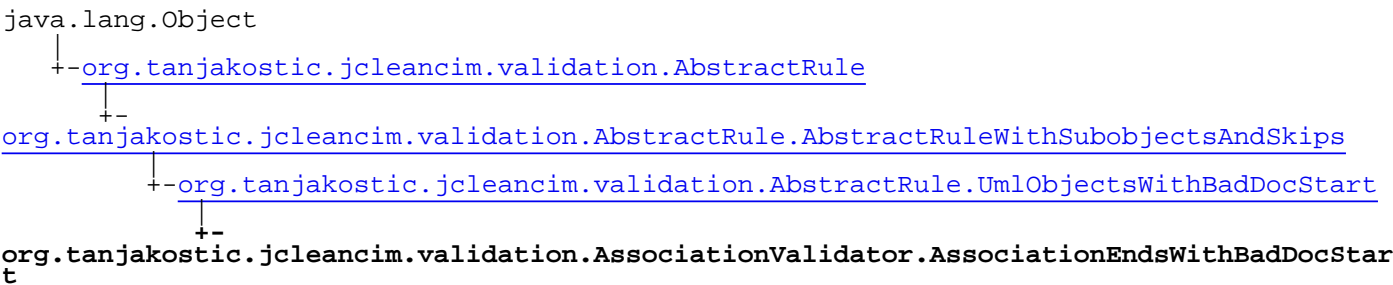
getSubObjects

```
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



All Implemented Interfaces:
Rule, [SimpleRule](#)

public static class **AssociationValidator.AssociationEndsWithBadDocStart**
extends [AbstractRule.UmlObjectsWithBadDocStart](#)

Fields inherited from class org.tanjakostic.jcleancim.validation.AbstractRule.UmlObjectsWithBadDocStart	
	DOCSTART_CHARS

Constructor Summary	
public	AssociationValidator.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

AssociationValidator.AssociationEndsWithBadDocStart

```
public AssociationValidator.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	AssociationValidator.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

AssociationValidator.AssociationEndsWithBadDocEnd

```
public AssociationValidator.AssociationEndsWithBadDocEnd()
```

Methods

getApplicability

```
public java.util.EnumSet getApplicability()
```

getSubObjects

```
protected final java.util.List getSubObjects(UmlAssociation assoc)
```

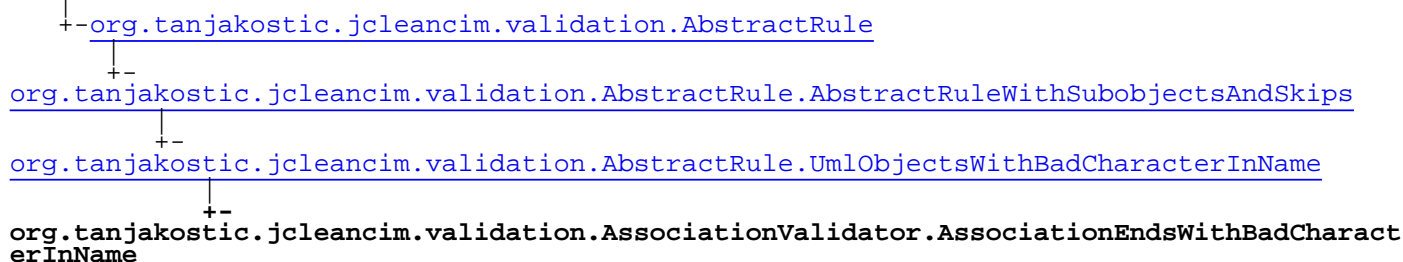
Used for validation of "sub-objects" (such as association ends and operation parameters) that do not have their own validators, but get validated through their container.

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



All Implemented Interfaces:

Rule, [SimpleRule](#)

public static class **AssociationValidator.AssociationEndsWithBadCharacterInName**
 extends [AbstractRule.UmlObjectsWithBadCharacterInName](#)

Constructor Summary

public	AssociationValidator.AssociationEndsWithBadCharacterInName()
--------	--

Method Summary

java.util.EnumSet	getApplicability()
InvalidCharacterFinder	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

AssociationValidator.AssociationEndsWithBadCharacterInName

```
public AssociationValidator.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	AssociationValidator.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

AssociationValidator.CimAssociationEndsNameStartingWithLowerCase

```
public AssociationValidator.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	AssociationValidator.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

AssociationValidator.CimAssociationEndsNameShouldBePlural

```
public AssociationValidator.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	AssociationValidator.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

AssociationValidator.CimAssociationEndsNameShouldBeSingular

```
public AssociationValidator.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	AttributeValidator.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

AttributeValidator.EnumLiteralsWithSuperfluousType

```
public AttributeValidator.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	AttributeValidator.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

AttributeValidator.EnumLiteralsWithoutEnumStereotype

```
public AttributeValidator.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	AttributeValidator.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

AttributeValidator.AttributesWithInvalidMultiplicity

```
public AttributeValidator.AttributesWithInvalidMultiplicity()
```

Methods

getApplicability

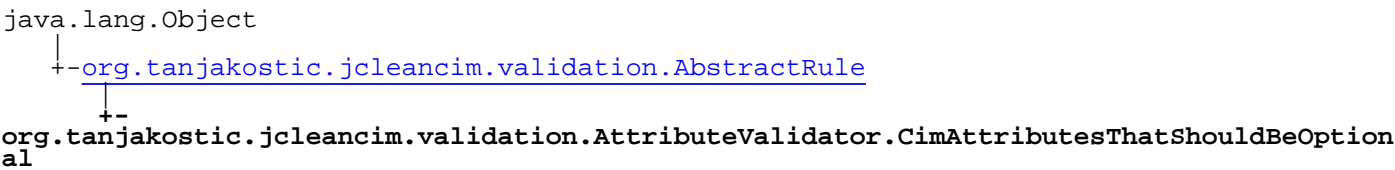
```
public java.util.EnumSet getApplicability()
```

validate

```
public void validate(UmlAttribute o,  
                    ModelIssues issues)
```

org.tanjakostic.jcleancim.validation

Class AttributeValidator.CimAttributesThatShouldBeOptional



All Implemented Interfaces:
[SimpleRule](#), [Rule](#)

public static class **AttributeValidator.CimAttributesThatShouldBeOptional**
extends [AbstractRule](#)
implements [Rule](#), [SimpleRule](#)

Constructor Summary	
public	AttributeValidator.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

AttributeValidator.CimAttributesThatShouldBeOptional

```
public AttributeValidator.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	AttributeValidator.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

AttributeValidator.AttributesWithInvalidTypeNull

```
public AttributeValidator.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	AttributeValidator.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

AttributeValidator.AttributesWithInvalidTypeString

```
public AttributeValidator.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	AttributeValidator.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

AttributeValidator.AttributesWithTypeIdMismatch

```
public AttributeValidator.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	AttributeValidator.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

AttributeValidator.CimAttributesThatShouldBePublic

```
public AttributeValidator.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

[AttributeValidator.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

AttributeValidator.AttributesThatAreStaticButNotConst

```
public AttributeValidator.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	AttributeValidator.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

AttributeValidator.CimAttributesThatAreNotStaticNonConstWithInitVal

```
public AttributeValidator.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	AttributeValidator.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

AttributeValidator.AttributesThatAreConstNonStatic

```
public AttributeValidator.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	AttributeValidator.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

AttributeValidator.AttributesWithUnallowedStereotype

```
public AttributeValidator.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	AttributeValidator.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

AttributeValidator.AttributesThatAreEnumsInNonEnumeratedClass

```
public AttributeValidator.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	AttributeValidator.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

AttributeValidator.CimAttributesThatShouldBeReplacedWithAssociation

```
public AttributeValidator.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	AttributeValidator.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

AttributeValidator.AttributesWhoseTypeIsInformative

```
public AttributeValidator.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	AttributeValidator.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

AttributeValidator.AttributesWithUnallowedTagNames

```
public AttributeValidator.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	AttributeValidator.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

AttributeValidator.Iec61850AttributesWithInexistingSibling

```
public AttributeValidator.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

[AttributeValidator.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

AttributeValidator.CimAttributesWithFlagInName

```
public AttributeValidator.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	AttributeValidator.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

AttributeValidator.AttributesMissingDoc

```
public AttributeValidator.AttributesMissingDoc()
```

Methods

getApplicability

```
public java.util.EnumSet getApplicability()
```

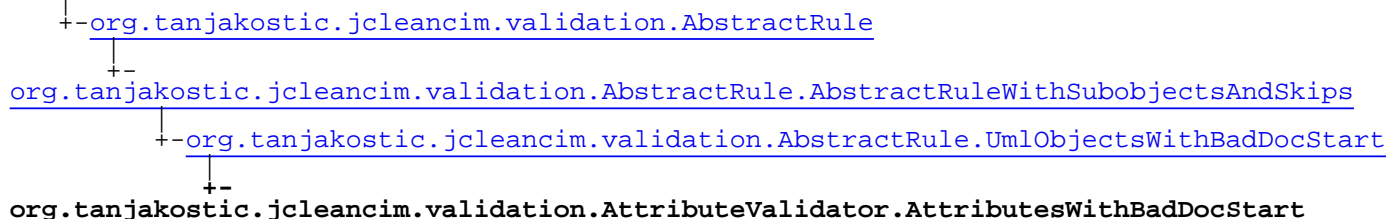
skipValidation

```
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



All Implemented Interfaces:

Rule, [SimpleRule](#)

public static class **AttributeValidator.AttributesWithBadDocStart**
extends [AbstractRule.UmlObjectsWithBadDocStart](#)

Fields inherited from class

[org.tanjakostic.jcleancim.validation.AbstractRule.UmlObjectsWithBadDocStart](#)

[DOCSTART_CHARS](#)

Constructor Summary

public	AttributeValidator.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

AttributeValidator.AttributesWithBadDocStart

```
public AttributeValidator.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	AttributeValidator.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

AttributeValidator.AttributesWithBadDocEnd

```
public AttributeValidator.AttributesWithBadDocEnd()
```

Methods

getApplicability

```
public java.util.EnumSet getApplicability()
```

skipValidation

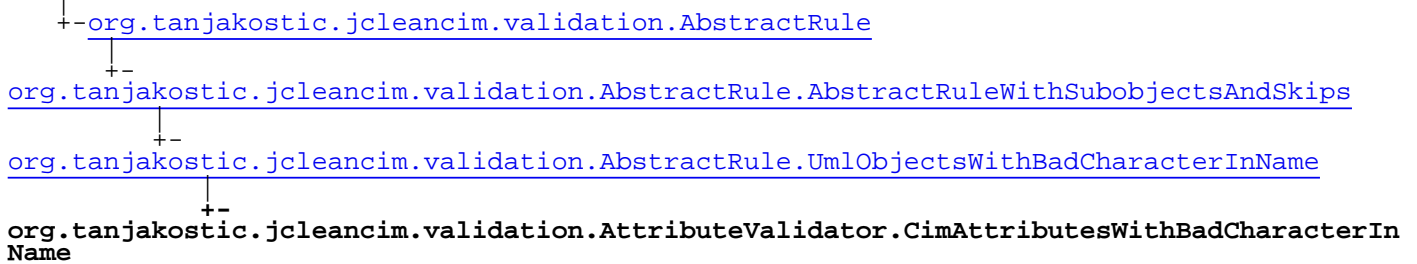
```
protected boolean skipValidation(UmlAttribute o)
```

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



All Implemented Interfaces:

Rule, [SimpleRule](#)

public static class **AttributeValidator.CimAttributesWithBadCharacterInName**
 extends [AbstractRule.UmlObjectsWithBadCharacterInName](#)

Constructor Summary

public	AttributeValidator.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

AttributeValidator.CimAttributesWithBadCharacterInName

```
public AttributeValidator.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.

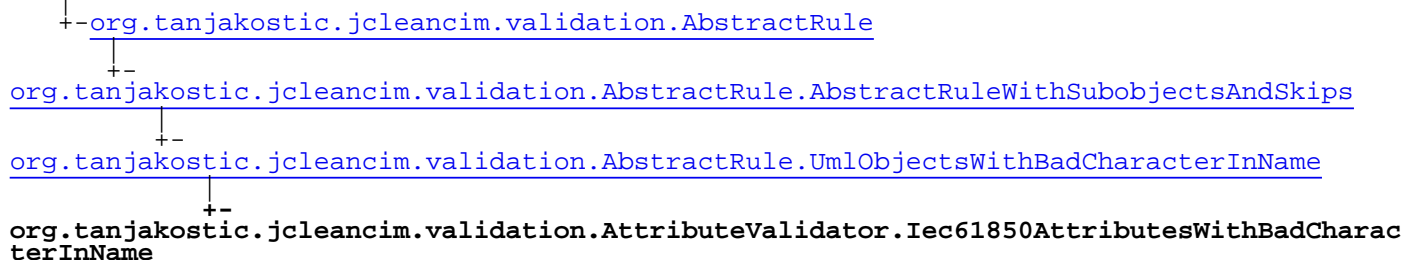
getInvalidCharacterFinder

```
protected InvalidCharactersFinder getInvalidCharacterFinder(UmlObject o)
```

org.tanjakostic.jcleancim.validation

Class AttributeValidator.Iec61850AttributesWithBadCharacterInName

java.lang.Object



All Implemented Interfaces:

Rule, [SimpleRule](#)

public static class **AttributeValidator.Iec61850AttributesWithBadCharacterInName**
 extends [AbstractRule.UmlObjectsWithBadCharacterInName](#)

Constructor Summary

public	AttributeValidator.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

AttributeValidator.Iec61850AttributesWithBadCharacterInName

```
public AttributeValidator.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	AttributeValidator.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

AttributeValidator.Iec61850DOAttributesWithTooLongName

```
public AttributeValidator.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	AttributeValidator.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

AttributeValidator.Iec61850FCDAAttributesWithMissingConstraint

```
public AttributeValidator.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	AttributeValidator.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

AttributeValidator.AttributesWithInexistingEnumLiteralAsInitValue

```
public AttributeValidator.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	AttributeValidator.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

AttributeValidator.Iec61850DOAttributesWithNameMissingAbbreviation

```
public AttributeValidator.Iec61850DOAttributesWithNameMissingAbbreviation()
```

Methods

getApplicability

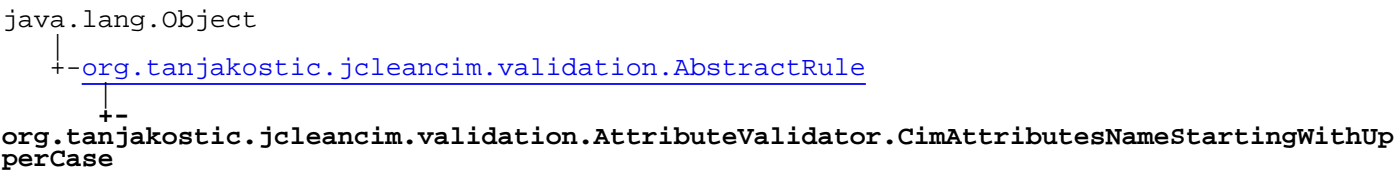
```
public java.util.EnumSet getApplicability()
```

validate

```
public void validate(UmlAttribute o,  
                    ModelIssues issues)
```

org.tanjakostic.jcleancim.validation

Class AttributeValidator.CimAttributesNameStartingWithUpperCase



All Implemented Interfaces:
[SimpleRule](#), [Rule](#)

public static class **AttributeValidator.CimAttributesNameStartingWithUpperCase**
extends [AbstractRule](#)
implements [Rule](#), [SimpleRule](#)

Constructor Summary	
public	AttributeValidator.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

AttributeValidator.CimAttributesNameStartingWithUpperCase

```
public AttributeValidator.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	AttributeValidator.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

AttributeValidator.CimAttributesNameShouldBeSingular

```
public AttributeValidator.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	AttributeValidator.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

AttributeValidator.CimAttributesNameShouldNotStartWithClassName

```
public AttributeValidator.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	AttributeValidator.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

AttributeValidator.Iec61850AbbreviationLiteralsNameStartingWithLowerCase

```
public AttributeValidator.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	AttributeValidator.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

AttributeValidator.Iec61850DOAttributesNameStartingWithLowerCase

```
public AttributeValidator.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	AttributeValidator.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

AttributeValidator.AttributesWithTypeFromUnallowedOwner

```
public AttributeValidator.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	AttributeValidator.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

AttributeValidator.Iec61850DOAbbreviationLiteralsDuplicateName

```
public  
AttributeValidator.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	AttributeValidator.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

AttributeValidator.Iec61850DOAbbreviationLiteralsDuplicateDescription

```
public  
AttributeValidator.Iec61850DOAbbreviationLiteralsDuplicateDescription(java.util Collec  
tion 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	AttributeValidator.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

AttributeValidator.Iec61850DOAbbreviationLiteralsNeverUsedInDOName

```
public  
AttributeValidator.Iec61850DOAbbreviationLiteralsNeverUsedInDOName(java.util.Collectio  
n 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	AttributeValidator.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

AttributeValidator.Iec61850DOAttributesWithSameNameDifferentType

```
public  
AttributeValidator.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	AttributeValidator.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

AttributeValidator.Iec61850ConditionLiteralsNeverUsedAsConstraints

```
public  
AttributeValidator.Iec61850ConditionLiteralsNeverUsedAsConstraints(java.util.Collectio  
n 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

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	ClassValidator.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

ClassValidator.CimClassesWithUnexpectedElements

```
public ClassValidator.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	ClassValidator.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

ClassValidator.ClassesWithUnexpectedConnectors

```
public ClassValidator.ClassesWithUnexpectedConnectors()
```

Methods

getApplicability

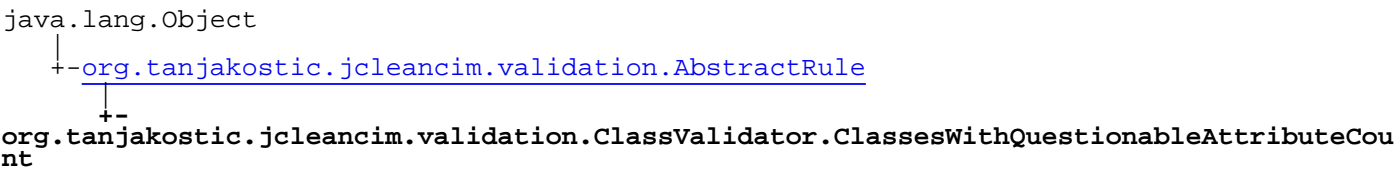
```
public java.util.EnumSet getApplicability()
```

validate

```
public void validate(UmlClass o,  
                    ModelIssues issues)
```

org.tanjakostic.jcleancim.validation

Class 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	ClassValidator.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

ClassValidator.ClassesWithQuestionableAttributeCount

```
public ClassValidator.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.ClassesWithQuestionableAttributeCount

+--org.tanjakostic.jcleancim.validation.ClassValidator.EnumClassesWithNoLiterals

All Implemented Interfaces:
Rule, SimpleRule

public static class **ClassValidator.EnumClassesWithNoLiterals**
extends [ClassValidator.ClassesWithQuestionableAttributeCount](#)

Constructor Summary

public	ClassValidator.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

ClassValidator.EnumClassesWithNoLiterals

```
public ClassValidator.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.ClassesWithQuestionableAttributeCount

+

-

org.tanjakostic.jcleancim.validation.ClassValidator.CimCompoundClassesWithNoAttributes

All Implemented Interfaces:
Rule, SimpleRule

public static class **ClassValidator.CimCompoundClassesWithNoAttributes**
extends [ClassValidator.ClassesWithQuestionableAttributeCount](#)

Constructor Summary

public	ClassValidator.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

ClassValidator.CimCompoundClassesWithNoAttributes

```
public ClassValidator.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.ClassesWithQuestionableAttributeCount](#)

└- [org.tanjakostic.jcleancim.validation.ClassValidator.EnumClassesWithSingleLiteral](#)

All Implemented Interfaces:

Rule, [SimpleRule](#)

public static class **ClassValidator.EnumClassesWithSingleLiteral**

extends [ClassValidator.ClassesWithQuestionableAttributeCount](#)

Constructor Summary

public	ClassValidator.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

ClassValidator.EnumClassesWithSingleLiteral

```
public ClassValidator.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.ClassesWithQuestionableAttributeCount](#)

└- **org.tanjakostic.jcleancim.validation.ClassValidator.EnumClassesWithTwoLiterals**

All Implemented Interfaces:

Rule, [SimpleRule](#)

public static class **ClassValidator.EnumClassesWithTwoLiterals**
extends [ClassValidator.ClassesWithQuestionableAttributeCount](#)

Constructor Summary

public	ClassValidator.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

ClassValidator.EnumClassesWithTwoLiterals

```
public ClassValidator.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	ClassValidator.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

ClassValidator.EnumClassesWithBadName

```
public ClassValidator.EnumClassesWithBadName()
```

Methods

getApplicability

```
public java.util.EnumSet getApplicability()
```

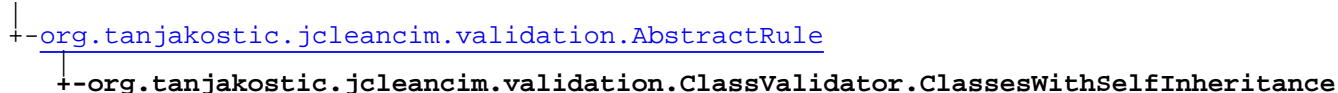
validate

```
public void validate(UmlClass o,
ModelIssues issues)
```

(continued from last page)

org.tanjakostic.jcleancim.validation Class ClassValidator.ClassesWithSelfInheritance

java.lang.Object



All Implemented Interfaces:

[SimpleRule](#), [Rule](#)

public static class **ClassValidator.ClassesWithSelfInheritance**

extends [AbstractRule](#)

implements [Rule](#), [SimpleRule](#)

Constructor Summary

public	ClassValidator.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

ClassValidator.ClassesWithSelfInheritance

```
public ClassValidator.ClassesWithSelfInheritance()
```

Methods

getApplicability

```
public java.util.EnumSet getApplicability()
```

validate

```
public void validate(UmlClass o,  
                    ModelIssues issues)
```

SimpleRule, Rule

```
implements Rule, SimpleRule
```

(continued from last page)

Constructors

ClassValidator.CimPrimitiveClassesWithAttributes

```
public ClassValidator.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	ClassValidator.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

ClassValidator.CimPrimitiveClassesWithIllegalOwner

```
public ClassValidator.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	ClassValidator.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

ClassValidator.ClassesWithDuplicateInheritedAttributeNames

```
public ClassValidator.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	ClassValidator.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

ClassValidator.ClassesWithDuplicateOwnOrInheritedAssociationEndNames

```
public ClassValidator.ClassesWithDuplicateOwnOrInheritedAssociationEndNames()
```

Methods

getApplicability

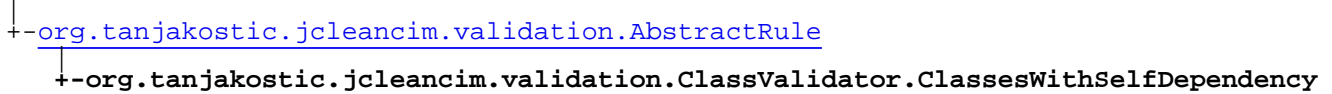
```
public java.util.EnumSet getApplicability()
```

validate

```
public void validate(UmlClass o,  
                    ModelIssues issues)
```

org.tanjakostic.jcleancim.validation Class ClassValidator.ClassesWithSelfDependency

java.lang.Object



All Implemented Interfaces:

[SimpleRule](#), [Rule](#)

public static class **ClassValidator.ClassesWithSelfDependency**

extends [AbstractRule](#)

implements [Rule](#), [SimpleRule](#)

Constructor Summary

public	ClassValidator.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

ClassValidator.ClassesWithSelfDependency

```
public ClassValidator.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	ClassValidator.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

ClassValidator.ClassesWithLeafPropSet

```
public ClassValidator.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	ClassValidator.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

ClassValidator.ClassesWithRootPropSet

```
public ClassValidator.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	ClassValidator.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

ClassValidator.ClassesWithPersistentPropSet

```
public ClassValidator.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	ClassValidator.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

ClassValidator.ClassesWithMultipleSuperclasses

```
public ClassValidator.ClassesWithMultipleSuperclasses()
```

Methods

getApplicability

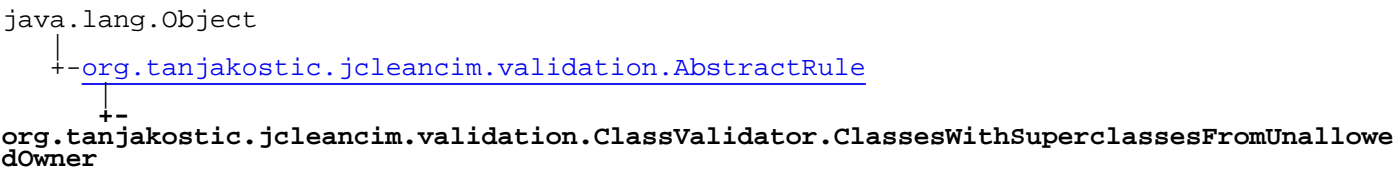
```
public java.util.EnumSet getApplicability()
```

validate

```
public void validate(UmlClass o,  
    ModelIssues issues)
```

org.tanjakostic.jcleancim.validation

Class ClassValidator.ClassesWithSuperclassesFromUnallowedOwner



All Implemented Interfaces:
[SimpleRule](#), Rule

public static class **ClassValidator.ClassesWithSuperclassesFromUnallowedOwner**
extends [AbstractRule](#)
implements Rule, [SimpleRule](#)

Constructor Summary

public	ClassValidator.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

ClassValidator.ClassesWithSuperclassesFromUnallowedOwner

```
public ClassValidator.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	ClassValidator.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

ClassValidator.ClassesThatShouldNotBeAssociationClass

```
public ClassValidator.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.ClassValidator.ClassesWithUnallowedStereotype

All Implemented Interfaces:

[SimpleRule](#), Rule

public static class **ClassValidator.ClassesWithUnallowedStereotype**

extends [AbstractRule](#)

implements Rule, [SimpleRule](#)

Constructor Summary

public	ClassValidator.ClassesWithUnallowedStereotype()
--------	---

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

ClassValidator.ClassesWithUnallowedStereotype

```
public ClassValidator.ClassesWithUnallowedStereotype()
```

Methods

getApplicability

```
public java.util.EnumSet getApplicability()
```

validate

```
public void validate(UmlClass o,  
    ModelIssues issues)
```

org.tanjakostic.jcleancim.validation Class ClassValidator.CimClassesWithOldDataaStereotype

java.lang.Object

└─ [org.tanjakostic.jcleancim.validation.AbstractRule](#)

└─

org.tanjakostic.jcleancim.validation.ClassValidator.CimClassesWithOldDataaStereotype

All Implemented Interfaces:

[SimpleRule](#), Rule

public static class **ClassValidator.CimClassesWithOldDataaStereotype**

extends [AbstractRule](#)

implements Rule, [SimpleRule](#)

Constructor Summary

public	ClassValidator.CimClassesWithOldDataaStereotype()
--------	---

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

ClassValidator.CimClassesWithOldDatatypeStereotype

```
public ClassValidator.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	ClassValidator.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

ClassValidator.CimClassesUsedForAttributesButHaveAssociations

```
public ClassValidator.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	ClassValidator.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

ClassValidator.CimClassesUsedForAttributesButHaveSubclasses

```
public ClassValidator.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	ClassValidator.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

ClassValidator.CimClassesUsedForAttributesButHaveSuperclasses

```
public ClassValidator.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	ClassValidator.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

ClassValidator.CimClassesThatShouldNotBeAbstract

```
public ClassValidator.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	ClassValidator.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

ClassValidator.CimClassesThatShouldNotHaveOperations

```
public ClassValidator.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	ClassValidator.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

ClassValidator.CimClassesThatShouldNotHaveExplicitDependencies

```
public ClassValidator.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	ClassValidator.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

ClassValidator.ClassesThatShouldNotHaveNestingThroughAttribute

```
public ClassValidator.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	ClassValidator.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

ClassValidator.Iec61850ClassesThatShouldHaveAliasAsTitle

```
public ClassValidator.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	ClassValidator.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

ClassValidator.Iec61850ClassesThatShouldHaveTaggedValuesForDocgen

```
public ClassValidator.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	ClassValidator.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

ClassValidator.CimClassesNeverUsedInRelationships

```
public ClassValidator.CimClassesNeverUsedInRelationships()
```

Methods

getApplicability

```
public java.util.EnumSet getApplicability()
```

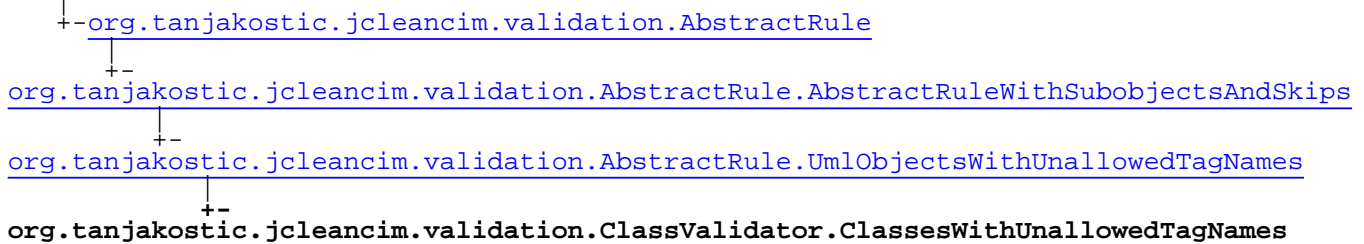
validate

```
public void validate(UmlClass o,  
    ModelIssues issues)
```

org.tanjakostic.jcleancim.validation

Class ClassValidator.ClassesWithUnallowedTagNames

java.lang.Object



All Implemented Interfaces:

Rule, [SimpleRule](#)

public static class **ClassValidator.ClassesWithUnallowedTagNames**

extends [AbstractRule.UmlObjectsWithUnallowedTagNames](#)

Constructor Summary

public [ClassValidator.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

ClassValidator.ClassesWithUnallowedTagNames

```
public ClassValidator.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	ClassValidator.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

ClassValidator.Iec61850ClassesWithInvalidConstraints

```
public ClassValidator.Iec61850ClassesWithInvalidConstraints()
```

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	ClassValidator.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

ClassValidator.Iec61850ClassesWithMissingCondIDTextInConstraints

```
public ClassValidator.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	ClassValidator.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

ClassValidator.CimDatatypeClassesWithInvalidAttributes

```
public ClassValidator.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	ClassValidator.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

ClassValidator.ClassesMissingDoc

```
public ClassValidator.ClassesMissingDoc()
```

Methods

getApplicability

```
public java.util.EnumSet getApplicability()
```

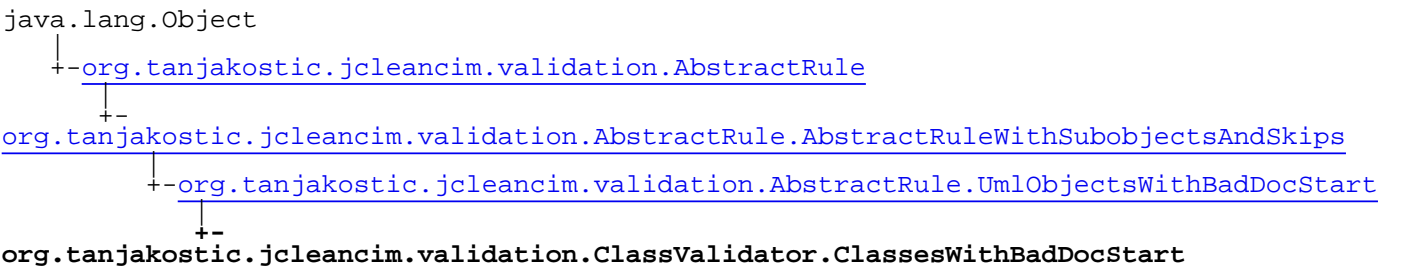
skipValidation

```
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



All Implemented Interfaces:
Rule, [SimpleRule](#)

public static class **ClassValidator.ClassesWithBadDocStart**
extends [AbstractRule.UmlObjectsWithBadDocStart](#)

Fields inherited from class org.tanjakostic.jcleancim.validation.AbstractRule.UmlObjectsWithBadDocStart	
	DOCSTART_CHARS

Constructor Summary	
public	ClassValidator.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

ClassValidator.ClassesWithBadDocStart

```
public ClassValidator.ClassesWithBadDocStart()
```

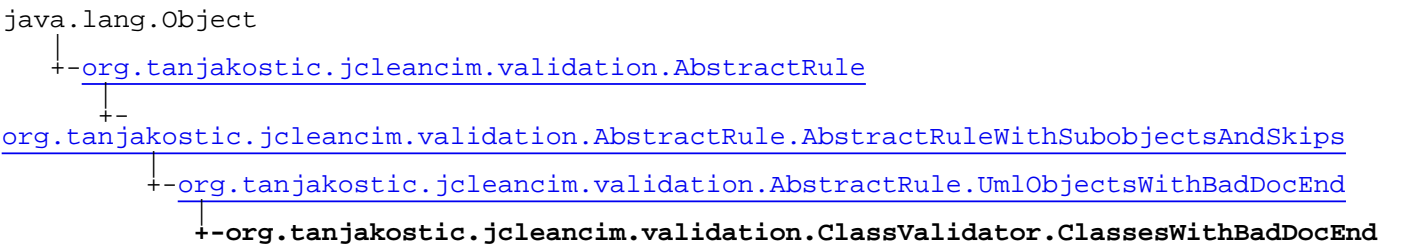
Methods

getApplicability

```
public java.util.EnumSet getApplicability()
```

org.tanjakostic.jcleancim.validation

Class ClassValidator.ClassesWithBadDocEnd



All Implemented Interfaces:
Rule, [SimpleRule](#)

public static class **ClassValidator.ClassesWithBadDocEnd**
extends [AbstractRule.UmlObjectsWithBadDocEnd](#)

Constructor Summary	
public	ClassValidator.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

ClassValidator.ClassesWithBadDocEnd

```
public ClassValidator.ClassesWithBadDocEnd()
```

Methods

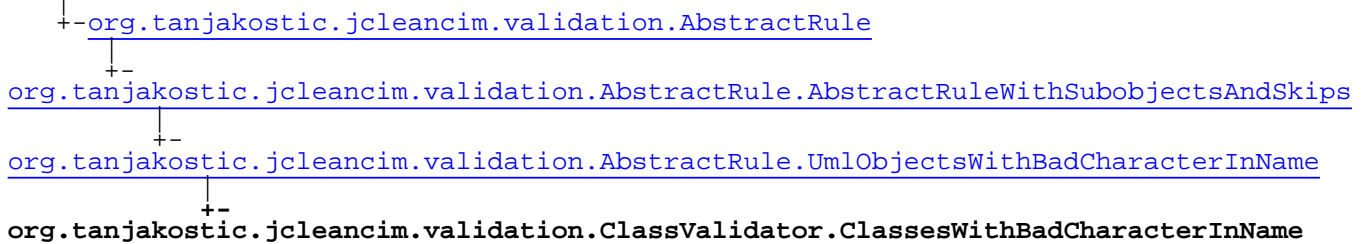
getApplicability

```
public java.util.EnumSet getApplicability()
```

org.tanjakostic.jcleancim.validation

Class ClassValidator.ClassesWithBadCharacterInName

java.lang.Object



All Implemented Interfaces:

Rule, [SimpleRule](#)

public static class **ClassValidator.ClassesWithBadCharacterInName**
 extends [AbstractRule.UmlObjectsWithBadCharacterInName](#)

Constructor Summary

public	ClassValidator.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

ClassValidator.ClassesWithBadCharacterInName

```
public ClassValidator.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	ClassValidator.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

ClassValidator.CimClassesNameStartingWithLowerCase

```
public ClassValidator.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	ClassValidator.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

ClassValidator.CimClassesNameShouldBeSingular

```
public ClassValidator.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	ClassValidator.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

ClassValidator.Iec61850LNClassesInWrongGroup

```
public ClassValidator.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	ClassValidator.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

ClassValidator.Iec61850LNClassesMalformedName

```
public ClassValidator.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	ClassValidator.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

ClassValidator.EnumClassesWithSomeCodesMissing

```
public ClassValidator.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	ClassValidator.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

ClassValidator.EnumClassesWithDuplicateCodes

```
public ClassValidator.EnumClassesWithDuplicateCodes()
```

Methods

getApplicability

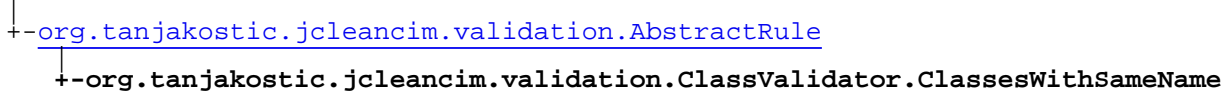
```
public java.util.EnumSet getApplicability()
```

validate

```
public void validate(UmlClass o,  
    ModelIssues issues)
```

org.tanjakostic.jcleancim.validation Class ClassValidator.ClassesWithSameName

java.lang.Object



All Implemented Interfaces:

[CrossRule](#), [Rule](#)

public static class **ClassValidator.ClassesWithSameName**

extends [AbstractRule](#)

implements [Rule](#), [CrossRule](#)

Constructor Summary

public	ClassValidator.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

ClassValidator.ClassesWithSameName

```
public ClassValidator.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	ClassValidator.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

ClassValidator.CimClassesNeverUsedAsTypeForAttribute

```
public ClassValidator.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

java.util.Collection	getObjsToTestAgainst() Returns list of objects against which validate(List, ModelIssues) works.
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 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 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)

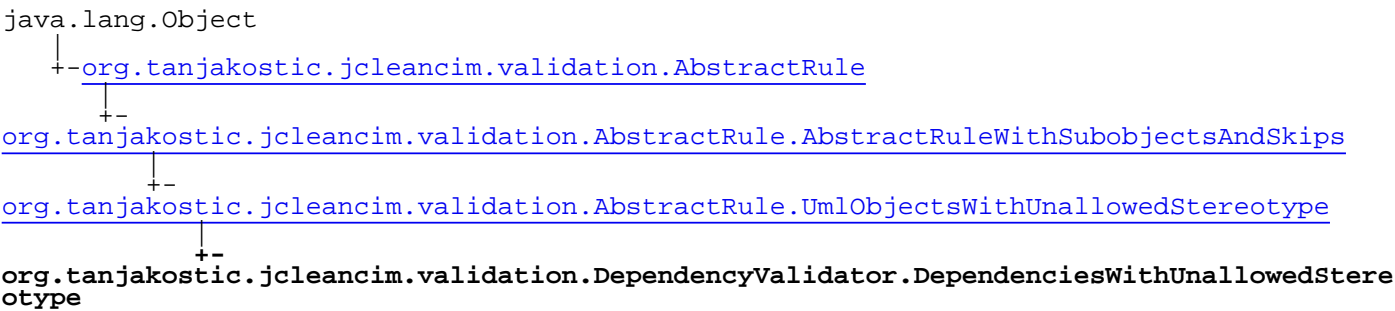
getScopedUmlObjects

```
public java.util.List getScopedUmlObjects()
```

Returns the elements retained for validation, for the configured scope.

org.tanjakostic.jcleancim.validation

Class DependencyValidator.DependenciesWithUnallowedStereotype



All Implemented Interfaces:
Rule, [SimpleRule](#)

public static class **DependencyValidator.DependenciesWithUnallowedStereotype**
extends [AbstractRule.UmlObjectsWithUnallowedStereotype](#)

Constructor Summary	
public	DependencyValidator.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

DependencyValidator.DependenciesWithUnallowedStereotype

```
public DependencyValidator.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	DependencyValidator.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

DependencyValidator.DependenciesWithUnallowedDirection

```
public DependencyValidator.DependenciesWithUnallowedDirection()
```

Methods

getApplicability

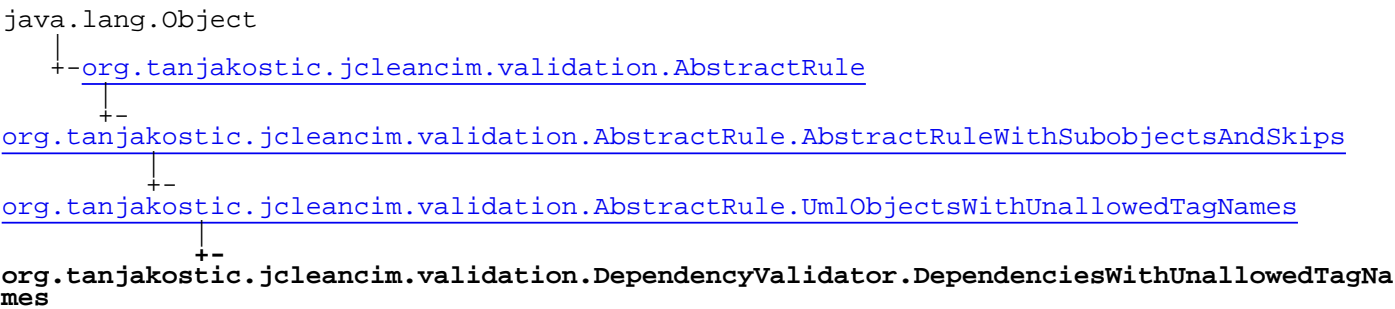
```
public java.util.EnumSet getApplicability()
```

validate

```
public void validate(UmlDependency d,  
                    ModelIssues issues)
```

org.tanjakostic.jcleancim.validation

Class DependencyValidator.DependenciesWithUnallowedTagNames



All Implemented Interfaces:
Rule, [SimpleRule](#)

public static class **DependencyValidator.DependenciesWithUnallowedTagNames**
extends [AbstractRule.UmlObjectsWithUnallowedTagNames](#)

Constructor Summary	
public	DependencyValidator.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

DependencyValidator.DependenciesWithUnallowedTagNames

```
public DependencyValidator.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	DiagramValidator.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

DiagramValidator.DiagramsWithBadOrientation

```
public DiagramValidator.DiagramsWithBadOrientation()
```

Methods

getApplicability

```
public java.util.EnumSet getApplicability()
```

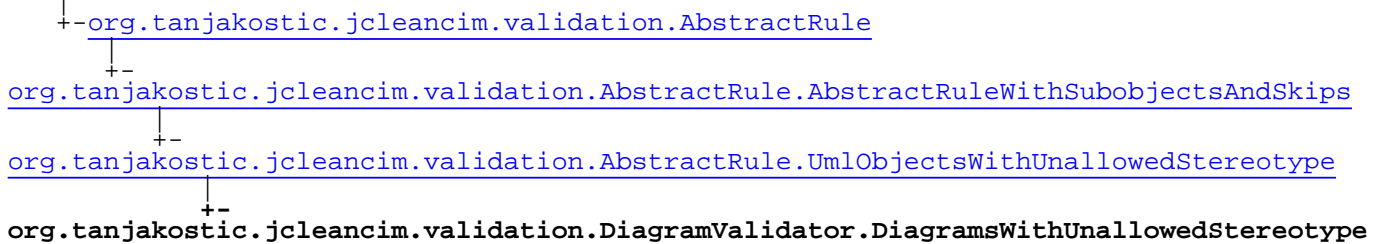
validate

```
public void validate(UmlDiagram d,  
    ModelIssues issues)
```

Matches if diagram orientation is not portrait.

org.tanjakostic.jcleancim.validation Class DiagramValidator.DiagramsWithUnallowedStereotype

java.lang.Object



All Implemented Interfaces:

Rule, [SimpleRule](#)

public static class **DiagramValidator.DiagramsWithUnallowedStereotype**
extends [AbstractRule.UmlObjectsWithUnallowedStereotype](#)

Constructor Summary

public	DiagramValidator.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

DiagramValidator.DiagramsWithUnallowedStereotype

```
public DiagramValidator.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	DiagramValidator.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

DiagramValidator.DiagramsMissingDoc

```
public DiagramValidator.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](#)

Fields inherited from class

[org.tanjakostic.jcleancim.validation.AbstractRule.UmlObjectsWithBadDocStart](#)

[DOCSTART_CHARS](#)

Constructor Summary

public [DiagramValidator.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

DiagramValidator.DiagramsWithBadDocStart

```
public DiagramValidator.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	DiagramValidator.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

DiagramValidator.DiagramsWithBadDocEnd

```
public DiagramValidator.DiagramsWithBadDocEnd()
```

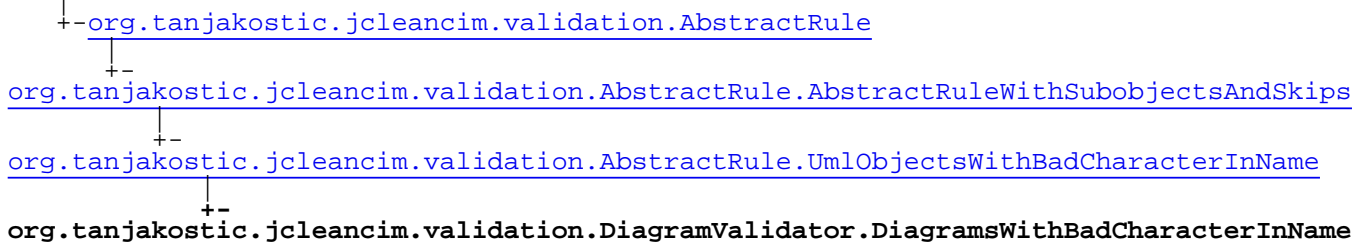
Methods

getApplicability

```
public java.util.EnumSet getApplicability()
```

org.tanjakostic.jcleancim.validation Class DiagramValidator.DiagramsWithBadCharacterInName

java.lang.Object



All Implemented Interfaces:

Rule, [SimpleRule](#)

public static class **DiagramValidator.DiagramsWithBadCharacterInName**
 extends [AbstractRule.UmlObjectsWithBadCharacterInName](#)

Constructor Summary

public	DiagramValidator.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

DiagramValidator.DiagramsWithBadCharacterInName

```
public DiagramValidator.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	OperationValidator.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

OperationValidator.OperationsWithUpperCaseName

```
public OperationValidator.OperationsWithUpperCaseName()
```

Methods

getApplicability

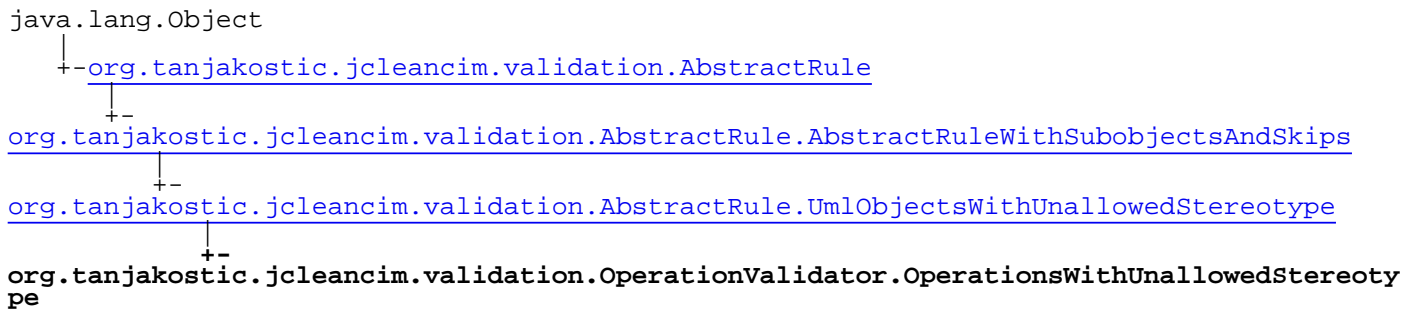
```
public java.util.EnumSet getApplicability()
```

validate

```
public void validate(UmlOperation o,  
    ModelIssues issues)
```

org.tanjakostic.jcleancim.validation

Class OperationValidator.OperationsWithUnallowedStereotype



All Implemented Interfaces:

Rule, [SimpleRule](#)

public static class **OperationValidator.OperationsWithUnallowedStereotype**
 extends [AbstractRule.UmlObjectsWithUnallowedStereotype](#)

Constructor Summary

public	OperationValidator.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

OperationValidator.OperationsWithUnallowedStereotype

```
public OperationValidator.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	OperationValidator.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

OperationValidator.OperationParametersWithUnallowedStereotype

```
public OperationValidator.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.OperationsWithInvalidReturnTypeInfo

java.lang.Object

+

-

org.tanjakostic.jcleancim.validation.AbstractRule

+

-

org.tanjakostic.jcleancim.validation.OperationValidator.OperationsWithInvalidReturnTypeInfo

Null

All Implemented Interfaces:

[SimpleRule](#), Rule

public static class **OperationValidator.OperationsWithInvalidReturnTypeInfo**
extends [AbstractRule](#)
implements Rule, [SimpleRule](#)

Constructor Summary

public	OperationValidator.OperationsWithInvalidReturnTypeInfo()
--------	--

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

OperationValidator.OperationsWithInvalidReturnTypeInfo

```
public OperationValidator.OperationsWithInvalidReturnTypeInfo()
```

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	OperationValidator.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

OperationValidator.OperationsWithInvalidArgTypeNull

```
public OperationValidator.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	OperationValidator.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

OperationValidator.OperationsWithInvalidExcTypeNull

```
public OperationValidator.OperationsWithInvalidExcTypeNull()
```

Methods

getApplicability

```
public java.util.EnumSet getApplicability()
```

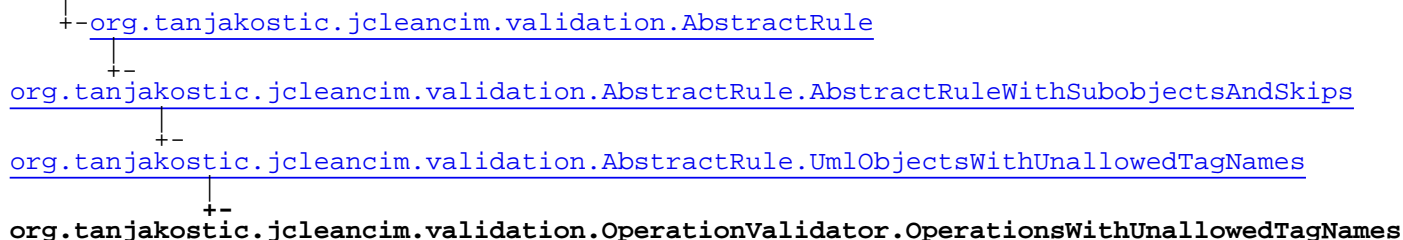
validate

```
public void validate(UmlOperation o,  
                    ModelIssues issues)
```

org.tanjakostic.jcleancim.validation

Class OperationValidator.OperationsWithUnallowedTagNames

java.lang.Object



All Implemented Interfaces:

Rule, [SimpleRule](#)

public static class **OperationValidator.OperationsWithUnallowedTagNames**
 extends [AbstractRule.UmlObjectsWithUnallowedTagNames](#)

Constructor Summary

public	OperationValidator.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

OperationValidator.OperationsWithUnallowedTagNames

```
public OperationValidator.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	OperationValidator.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

OperationValidator.OperationParametersWithUnallowedTagNames

```
public OperationValidator.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	OperationValidator.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

OperationValidator.OperationsMissingDoc

```
public OperationValidator.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	OperationValidator.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

OperationValidator.OperationParametersMissingDoc

```
public OperationValidator.OperationParametersMissingDoc()
```

Methods

getApplicability

```
public java.util.EnumSet getApplicability()
```

getSubObjects

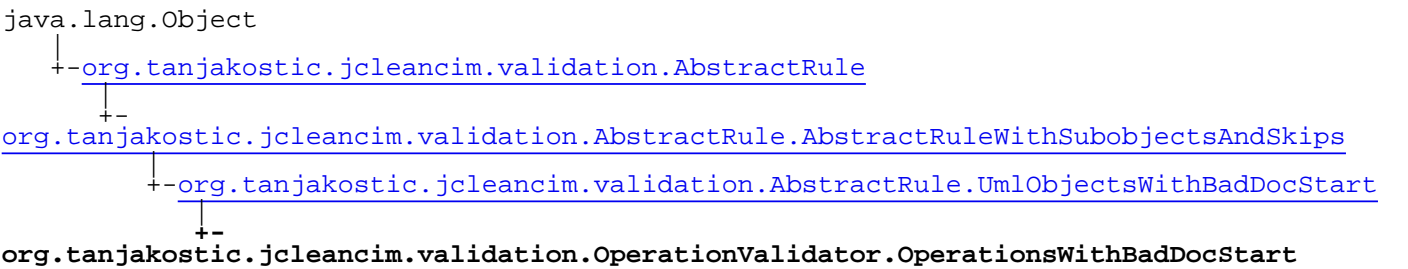
```
protected java.util.List getSubObjects(UmlOperation op)
```

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



All Implemented Interfaces:
Rule, [SimpleRule](#)

public static class **OperationValidator.OperationsWithBadDocStart**
extends [AbstractRule.UmlObjectsWithBadDocStart](#)

Fields inherited from class org.tanjakostic.jcleancim.validation.AbstractRule.UmlObjectsWithBadDocStart	
	DOCSTART_CHARS

Constructor Summary	
public	OperationValidator.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

OperationValidator.OperationsWithBadDocStart

```
public OperationValidator.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.OperationParametersWithBadDocStart

```

All Implemented Interfaces:

Rule, [SimpleRule](#)

public static class **OperationValidator.OperationParametersWithBadDocStart**
 extends [AbstractRule.UmlObjectsWithBadDocStart](#)

Fields inherited from class

[org.tanjakostic.jcleancim.validation.AbstractRule.UmlObjectsWithBadDocStart](#)

[DOCSTART_CHARS](#)

Constructor Summary

public	OperationValidator.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

OperationValidator.OperationParametersWithBadDocStart

```
public OperationValidator.OperationParametersWithBadDocStart()
```

Methods

getApplicability

```
public java.util.EnumSet getApplicability()
```

getSubObjects

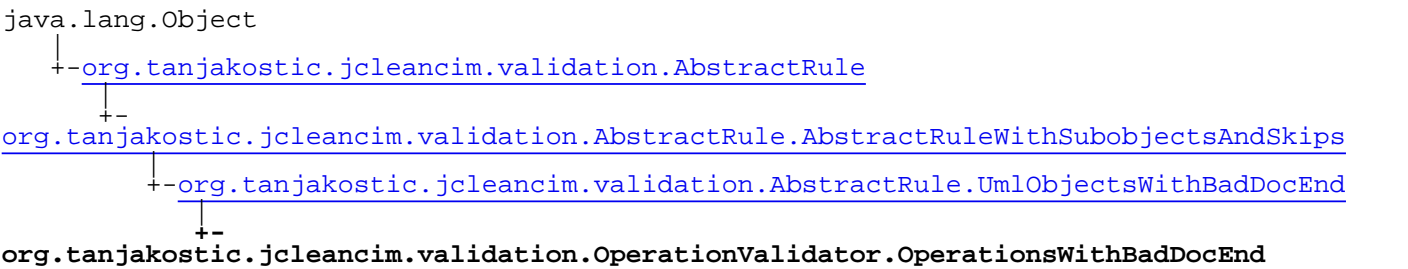
```
protected java.util.List getSubObjects(UmlOperation op)
```

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



All Implemented Interfaces:
Rule, [SimpleRule](#)

public static class **OperationValidator.OperationsWithBadDocEnd**
extends [AbstractRule.UmlObjectsWithBadDocEnd](#)

Constructor Summary	
public	OperationValidator.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

OperationValidator.OperationsWithBadDocEnd

```
public OperationValidator.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	OperationValidator.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

OperationValidator.OperationParametersWithBadDocEnd

```
public OperationValidator.OperationParametersWithBadDocEnd()
```

Methods

getApplicability

```
public java.util.EnumSet getApplicability()
```

getSubObjects

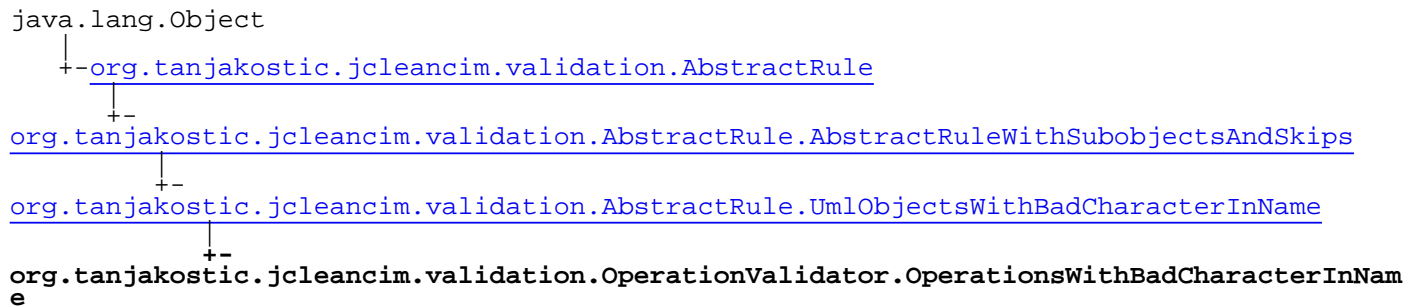
```
protected java.util.List getSubObjects(UmlOperation op)
```

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



All Implemented Interfaces:

Rule, [SimpleRule](#)

public static class **OperationValidator.OperationsWithBadCharacterInName**
 extends [AbstractRule.UmlObjectsWithBadCharacterInName](#)

Constructor Summary

public	OperationValidator.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

OperationValidator.OperationsWithBadCharacterInName

```
public OperationValidator.OperationsWithBadCharacterInName()
```

Methods

getApplicability

```
public java.util.EnumSet getApplicability()
```

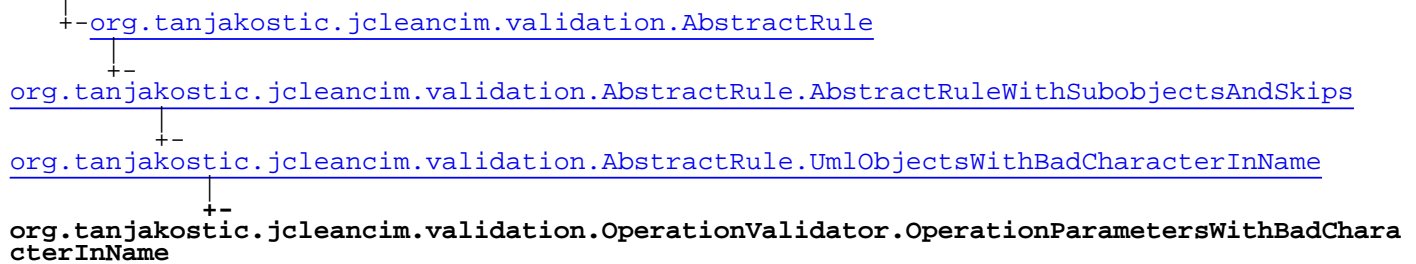
getInvalidCharacterFinder

```
protected InvalidCharactersFinder getInvalidCharacterFinder(UmlObject o)
```


org.tanjakostic.jcleancim.validation

Class OperationValidator.OperationParametersWithBadCharacterInName

java.lang.Object



All Implemented Interfaces:

Rule, [SimpleRule](#)

public static class **OperationValidator.OperationParametersWithBadCharacterInName**
 extends [AbstractRule.UmlObjectsWithBadCharacterInName](#)

Constructor Summary

public	OperationValidator.OperationParametersWithBadCharacterInName()
--------	--

Method Summary

java.util.EnumSet	getApplicability()
InvalidCharacterFinder	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

OperationValidator.OperationParametersWithBadCharacterInName

```
public OperationValidator.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	PackageValidator.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

PackageValidator.PackageUnexpectedElements

```
public PackageValidator.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	PackageValidator.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

PackageValidator.PackageUnexpectedConnectors

```
public PackageValidator.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	PackageValidator.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

PackageValidator.PackagesWithSelfDependency

```
public PackageValidator.PackagesWithSelfDependency()
```

Methods

getApplicability

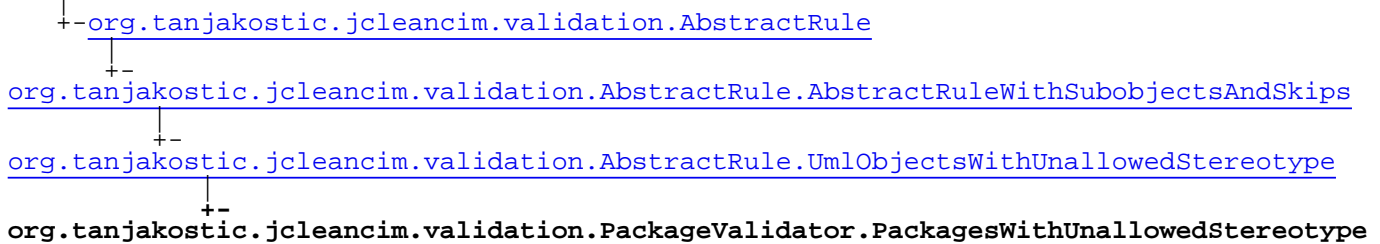
```
public java.util.EnumSet getApplicability()
```

validate

```
public void validate(UmlPackage o,  
    ModelIssues issues)
```

org.tanjakostic.jcleancim.validation Class PackageValidator.PackagesWithUnallowedStereotype

java.lang.Object



All Implemented Interfaces:

Rule, [SimpleRule](#)

public static class **PackageValidator.PackagesWithUnallowedStereotype**
 extends [AbstractRule.UmlObjectsWithUnallowedStereotype](#)

Constructor Summary

public	PackageValidator.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

PackageValidator.PackagesWithUnallowedStereotype

```
public PackageValidator.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	PackageValidator.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

PackageValidator.PackagesTopLevelWithoutVersionClass

```
public PackageValidator.PackagesTopLevelWithoutVersionClass()
```

Methods

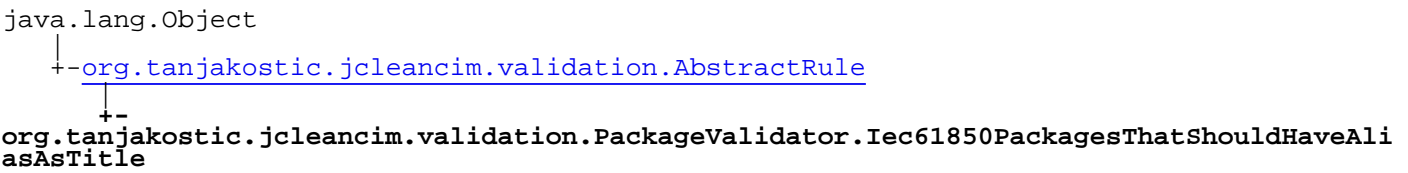
getApplicability

```
public java.util.EnumSet getApplicability()
```

validate

```
public void validate(UmlPackage o,  
                    ModelIssues issues)
```

org.tanjakostic.jcleancim.validation
Class PackageValidator.Iec61850PackagesThatShouldHaveAliasAsTitle



All Implemented Interfaces:
[SimpleRule](#), Rule

public static class **PackageValidator.Iec61850PackagesThatShouldHaveAliasAsTitle**
extends [AbstractRule](#)
implements Rule, [SimpleRule](#)

Constructor Summary

public	PackageValidator.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

PackageValidator.Iec61850PackagesThatShouldHaveAliasAsTitle

```
public PackageValidator.Iec61850PackagesThatShouldHaveAliasAsTitle()
```

Methods

getApplicability

```
public java.util.EnumSet getApplicability()
```

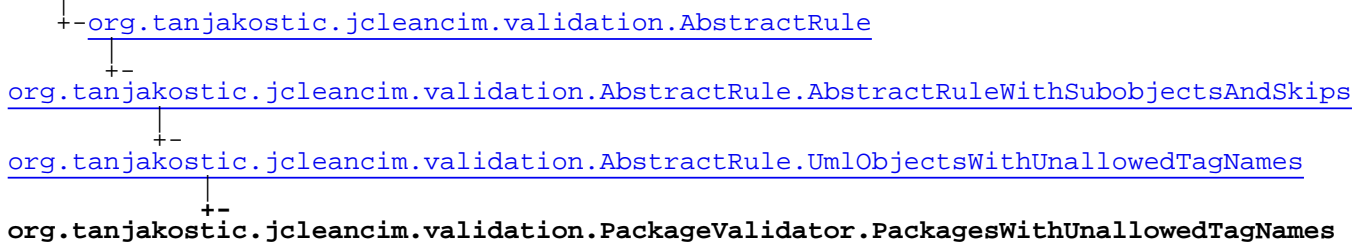
validate

```
public void validate(UmlPackage o,  
                    ModelIssues issues)
```


org.tanjakostic.jcleancim.validation

Class PackageValidator.PackagesWithUnallowedTagNames

java.lang.Object



All Implemented Interfaces:

Rule, [SimpleRule](#)

public static class **PackageValidator.PackagesWithUnallowedTagNames**
 extends [AbstractRule.UmlObjectsWithUnallowedTagNames](#)

Constructor Summary

public	PackageValidator.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

PackageValidator.PackagesWithUnallowedTagNames

```
public PackageValidator.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	PackageValidator.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

PackageValidator.PackagesMissingDoc

```
public PackageValidator.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](#)

Fields inherited from class

[org.tanjakostic.jcleancim.validation.AbstractRule.UmlObjectsWithBadDocStart](#)

[DOCSTART_CHARS](#)

Constructor Summary

public [PackageValidator.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

PackageValidator.PackagesWithBadDocStart

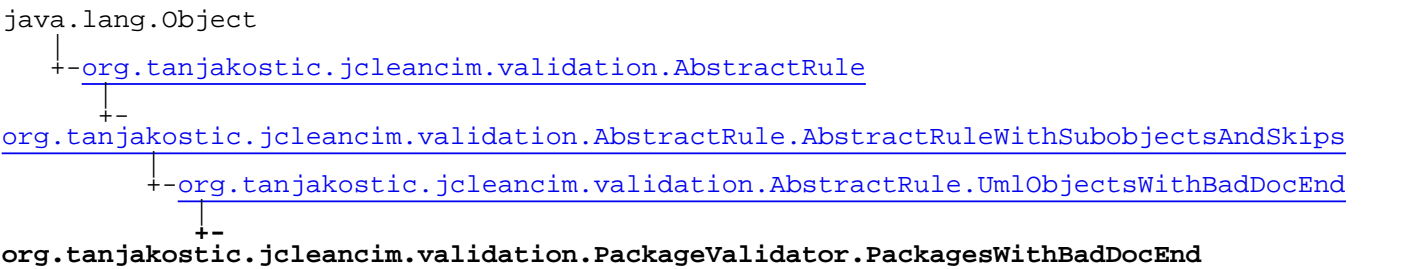
```
public PackageValidator.PackagesWithBadDocStart()
```

Methods

getApplicability

```
public java.util.EnumSet getApplicability()
```

org.tanjakostic.jcleancim.validation
Class PackageValidator.PackagesWithBadDocEnd



All Implemented Interfaces:
Rule, [SimpleRule](#)

public static class **PackageValidator.PackagesWithBadDocEnd**
extends [AbstractRule.UmlObjectsWithBadDocEnd](#)

Constructor Summary	
public	PackageValidator.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

PackageValidator.PackagesWithBadDocEnd

```
public PackageValidator.PackagesWithBadDocEnd()
```

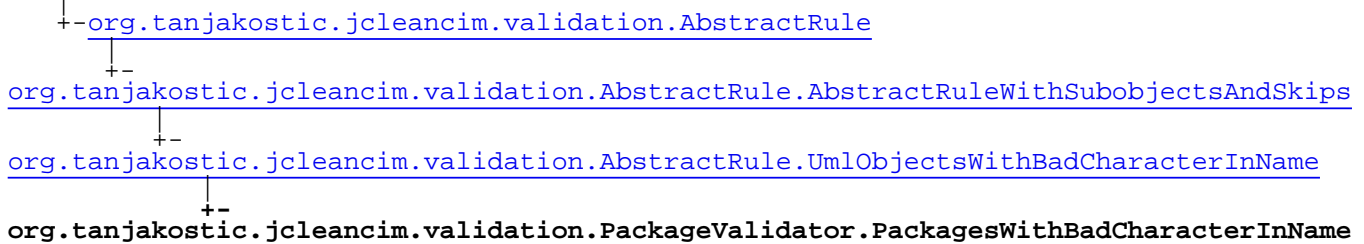
Methods

getApplicability

```
public java.util.EnumSet getApplicability()
```


org.tanjakostic.jcleancim.validation Class PackageValidator.PackagesWithBadCharacterInName

java.lang.Object



All Implemented Interfaces:

Rule, [SimpleRule](#)

public static class **PackageValidator.PackagesWithBadCharacterInName**
extends [AbstractRule.UmlObjectsWithBadCharacterInName](#)

Constructor Summary

public	PackageValidator.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

PackageValidator.PackagesWithBadCharacterInName

```
public PackageValidator.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	PackageValidator.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

PackageValidator.PackagesWithSameName

```
public PackageValidator.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](#), [AttributesWhoseTypesInformative](#), [CimAttributesThatShouldBeReplacedWithAssociation](#), [AttributesThatAreEnumsInNonEnumeratedClass](#), [AttributesThatAreConstNonStatic](#), [CimAttributesThatAreNotStaticNonConstWithInitVal](#), [AttributesThatAreStaticButNotConst](#), [CimAttributesThatShouldBePublic](#), [AttributesWithTypeIdMismatch](#), [AttributesWithInvalidTypeString](#), [AttributesWithInvalidTypeNull](#), [CimAttributesThatShouldBeOptional](#), [AttributesWithInvalidMultiplicity](#), [EnumLiteralsWithoutEnumStereotype](#), [EnumLiteralsWithSuperfluousType](#), [EnumClassesWithDuplicateCodes](#), [EnumClassesWithSomeCodesMissing](#), [Iec61850LNCClassesMalformedName](#), [Iec61850LNCClassesInWrongGroup](#), [CimClassesNameShouldBeSingular](#), [CimClassesNameStartingWithLowerCase](#), [CimDatatypeClassesWithInvalidAttributes](#), [Iec61850ClassesWithMissingCondIDTextInConstraints](#), [Iec61850ClassesWithInvalidConstraints](#), [CimClassesNeverUsedInRelationships](#), [Iec61850ClassesThatShouldHaveTaggedValuesForDocgen](#), [Iec61850ClassesThatShouldHaveAliasAsTitle](#), [ClassesThatShouldNotHaveNestingThroughAttribute](#), [CimClassesThatShouldNotHaveExplicitDependencies](#), [CimClassesThatShouldNotHaveOperations](#), [CimClassesThatShouldNotBeAbstract](#), [CimClassesUsedForAttributesButHaveSuperclasses](#), [CimClassesUsedForAttributesButHaveSubclasses](#), [CimClassesUsedForAttributesButHaveAssociations](#), [CimClassesWithOldDatatypeStereotype](#), [ClassesWithUnallowedStereotype](#), [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

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 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

SaxErrorData	getParsingErrors() Returns errors collected during parsing; may be empty but never null.
org.w3c.dom.Document	readAndValidate (java.io.File xmlFile) Reads and validates xmlFile and returns it as DOM document.
org.w3c.dom.Document	readAndValidate (org.xml.sax.InputSource source) Reads and validates source and returns it as DOM document.
org.w3c.dom.Document	readAndValidate (XmlString xmlText) Reads and validates xmlText and returns it as DOM document.

Methods

getParsingErrors

public [SaxErrorData](#) **getParsingErrors()**

Returns errors collected during parsing; may be empty but never null.

readAndValidate

public 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 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 org.w3c.dom.Document readAndValidate(org.xml.sax.InputSource source)
    throws XmlParsingException
```

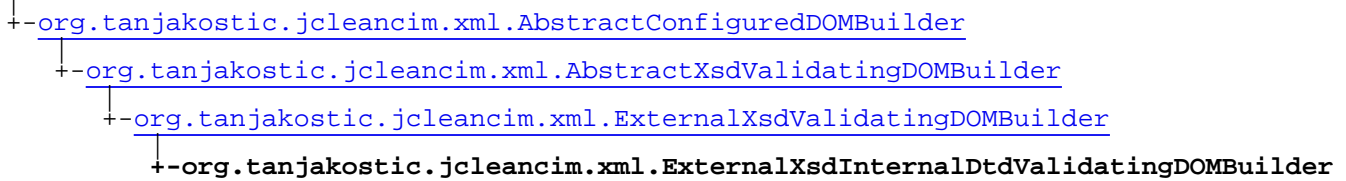
Reads and validates source and returns it as DOM document.

Throws:

[XmlParsingException](#)

org.tanjakostic.jcleancim.xml Class ExternalXsdInternalDtdValidatingDOMBuilder

java.lang.Object



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

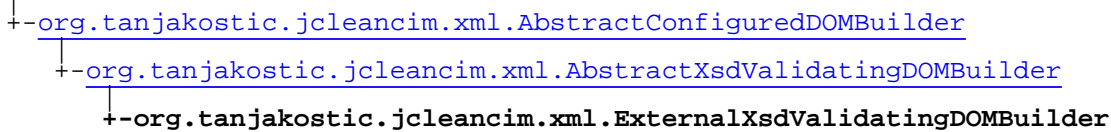
(continued from last page)

ExternalXsdInternalDtdValidatingDOMBuilder

```
public ExternalXsdInternalDtdValidatingDOMBuilder( java.io.InputStream externalSchema )
```

org.tanjakostic.jcleancim.xml Class ExternalXsdValidatingDOMBuilder

java.lang.Object



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)
```

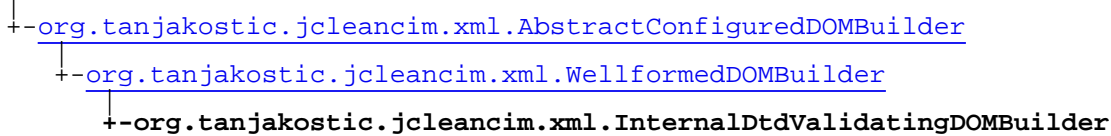
Methods

configureBuilderFactoryWithSchema

```
protected void  
configureBuilderFactoryWithSchema(javax.xml.parsers.DocumentBuilderFactory  
builderFactory)  
    throws XmlParsingException
```

org.tanjakostic.jcleancim.xml Class InternalDtdValidatingDOMBuilder

java.lang.Object



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](#)

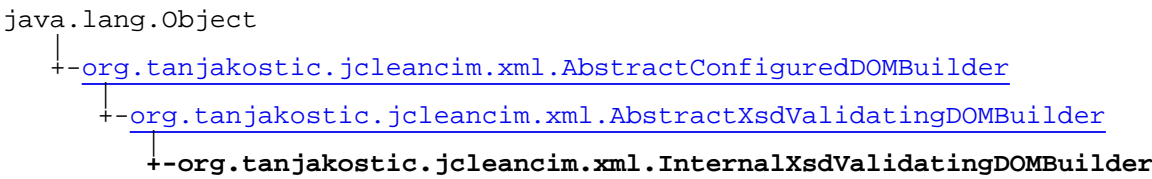
Constructors

InternalDtdValidatingDOMBuilder

public **InternalDtdValidatingDOMBuilder()**

org.tanjakostic.jcleancim.xml

Class InternalXsdValidatingDOMBuilder



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	<u>INDENT</u> Value: 2
---------------------	---

Method Summary

static org.w3c.dom.Element	<u>addCDATA</u> (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	<u>addNamespace</u> (org.w3c.dom.Document dom, <u>XmlNs</u> ns) Adds the namespace binding definition to DOM dom (to define multiple namespace bindings on the root element).
static org.w3c.dom.Element	<u>addQSubElement</u> (org.w3c.dom.Element el, java.lang.String qname, org.w3c.dom.Document document) Adds sub-element to el and returns modified el.
static <u>XmlString</u>	<u>asPrettyXml</u> (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 <u>XmlString</u>	<u>asPrettyXml</u> (<u>XmlString</u> xmlText, java.io.File xmlFile) Returns xmlText as pretty-print string and saves to xmlFile if not null.
static <u>XmlString</u>	<u>asXml</u> (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.XPath Expression	<u>compileXPath</u> (java.lang.String xpathExpression, <u>XmlNs[]</u> namespaces) Returns compiled xpath expression that recognises all namespaces.
static org.w3c.dom.Document	<u>createDocumentWithRoot</u> (java.lang.String comment, java.lang.String rootName, <u>XmlNs</u> ns) Returns a document with rootName element.
static org.w3c.dom.Element	<u>createQRoot</u> (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	<u>createQSubElement</u> (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 e1) Returns all attributes of e1, potentially empty list.
static org.w3c.dom.Element	getFirstNamedSubElement (org.w3c.dom.Element e1, java.lang.String name) Returns first element under e1 with the name if found, null otherwise.
static org.w3c.dom.Attr	getNamedAttribute (org.w3c.dom.Element e1, java.lang.String name) Returns attribute of e1 with the name if found, null otherwise.
static java.util.List	getNamedSubElements (org.w3c.dom.Element e1, java.lang.String name) Returns elements under e1 with the name if found, empty list otherwise.
static java.util.List	getSubElements (org.w3c.dom.Element e1) Returns all elements under e1, 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.DocumentF ragment	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)

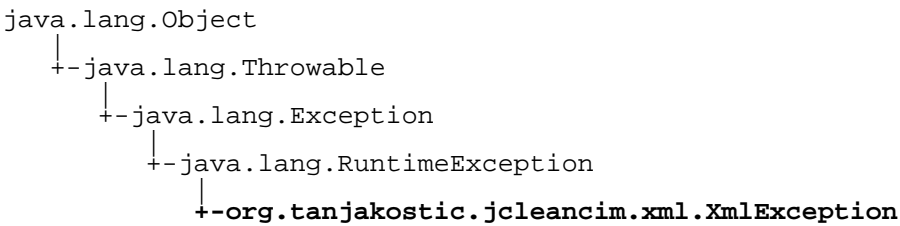
Methods

emptyDocument

public org.w3c.dom.Document **emptyDocument**()

org.tanjakostic.jcleancim.xml

Class XmlException



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](#)