Politecnico Di Milano

A.A. 2015/2016

Software Engineering 2:

Inspection Document

Paramithiotti Andrea (Matr. 788794) Rompani Andrea (Matr. 854052)

Zoia Lorenzo (Matr. 852392)

v1.0

5/1/2016

# Assigned Methods

All the methods assigned to the group belong to the class **DOLUtils**, in the package **com.sun.enterprise.deployment.util.**

1. readAlternativeRuntimeDescriptor
2. readRuntimeDeploymentDescriptor
3. getSniffersFromModule
4. getTypeFromModuleType
5. getConfigurationDeploymentDescriptorFiles
6. setElementValue

# Functional Role of the Methods

The Javadoc for the classes and the methods assigned was almost non-existent, thus the functional role of the methods was understood only by analysing the code line by line, trying to extrapolate the meaning of the actions performed also by reading the code of other classes as well.

## readAlternativeRuntimeDescriptor

### Parameters

* ReadableArchive appArchive
* ReadableArchive embeddedArchive
* Archivist archivist
* BundleDescriptor descriptor
* String altDDPath

### Functional Role

The Archivist contains the info for the deployment of determined modules. If the deployment method exists, then it is the one that is used. Otherwise this method is called, which after some control on the parameters takes an alternative descriptor from the parameters at runtime and saves it in the Archivist.

## readRuntimeDeploymentDescriptor

### Parameters

* List<ConfigurationDeploymentDescriptorFile> confDDFiles,
* ReadableArchive archive
* RootDeploymentDescriptor descriptor
* Archivist main
* final boolean warnIfMultipleDDs

### Functional Role

In the case that the parameter warnIfMultipleDDs is true, the method writes a warning on the depLogger for each element in the list confDDFiles. Then it takes the first ConfigurationDDFile in the list and sets its XML validation and its level of validation to true if confDD.isValidating() is true, otherwise it is set to false. The validation is taken from the Archivist that is passed as a parameter. In the end it calls the method Read() on the Descriptor passed as a parameter and on the InputStream that is taken from the Archivist passed as a parameter.

## getSniffersFromModule

### Parameters

* ServiceLocator habitat
* ReadableArchive archive,
* ModuleDescriptor md
* Application app

### Functional Role

The method gets the right sniffer from the snifferManager according to the parameters and defines the MainSniffer: it can be either the sniffer associated to the type of the ModuleDescriptor passed as a parameter or it can be taken directly from the snifferManager. Then it erases all the sniffers that are not compatible with the mainSniffer from the list of the sniffers, writes on the depLogger with Warning level, saves a hash table of the selected sniffers and returns it.

OPPURE PIU FACILMENTE

The method saves in the Archive passed as a parameter the list of the sniffers that are compatible with the module passed as a parameter, then returns it.

## getTypeFromModuleType

### Parameters

* ArchiveType moduleType

### Functional Role

The method returns a String that defines the type of the document passed as a parameter.

## getConfigurationDeploymentDescriptorFiles

### Parameters

* ServiceLocator habitat
* String containerType

### Functional Role

The method returns the configuration files useful for the deployment descriptors. In the list that is returned it saves only the elements for whom the indexedType of the descriptor of each handles of the habitat is equals to the type passed as a parameter.

## setElementValue

### Parameters

* XMLElement element
* String value
* Object o

### Functional Role

Regardless of what the method does, it is never called by any function in all the project, so it is a useless dead part of the code. Sometimes another method similar to this, but with different parameters, is called in the project

# List of Issues

# Other Problems

* As said before, all the method “setElementValue” inspected is a dead code, as it is never called in all the project. It may cause a misunderstanding for a developer with the task of improving the code, and it wpuld be necessary at least to underline this issue with a proper Javadoc or commented out lines of code