

University of Michigan - Dearborn

ECE591 – Directed Studies

Fall 2010

Installation and User Manual

Fuzzy System for Software Reliability Classification using Object Oriented Metrics

Presented by Juan Garcia

December 1, 2010

Dr. Adnan Shaout

Fuzzy System for Software Reliability Classification using Object Oriented Metrics

1. Installation & Execution of the program

This Program requires JDK 1.6 or newer version because it uses the Java Compiler tool to analyze the java source code. Please install the latest version of this program before you continue with installation and execution of this software. Please follow the steps below in order to install the required JDK and Java Swing. If you have the required software already installed in your computer, proceed with step 2.

Step 1. Install the latest version of Java J2SE 6 (JDK)

In the link below you'll find the different versions of JDK, click on the latest version to download and install it in your computer.

<http://www.oracle.com/technetwork/java/javase/downloads/index.html>

For any question see the installation instructions below

<http://www.oracle.com/technetwork/java/javase/index-137561.html>

Add the JavaCompiler package (usually the tools.jar file in your lib directory of the JDK version) to your classpath before executing the software.

Step 2. Install the program on your PC

Unzip the package BinCodeJuanCGarciaEce695.zip under any of the directories on your PC. Once the software is unzipped you are ready to configure and run the application.

Step 3.

Configure the software on your PC

Edit the file config_system_v2.0.xml and modify the xml tags accordingly to reconfigure the application. Below is the ex

Configuration of Metrics

Go to tag <metrics> and edit each of the sub tag for the metric

Each OO metric contains its own <metric> tag

Attributes

id = identifies the current metric

class-name = utilized to set the class for the current metric

Tags

<name> name of the OO metric

<definition> definition of the metric, include description, what and how is measure and the rationale of the metric

<solution> How the class can be modified to correct the problem

Configuration of fuzzy sets

Go the `<fuzzy-sets>` tag and for each of the OO metrics add also corresponding fuzzy set.

id = identifier for the fuzzy set

type = four types are currently available ascendent-set (2 values) descendent-set (2 values) triangle-set (3 values) and trapezoide-set(4 values)

label = label of the fuzzy set normal, medium, high, etc

value = depending on the type a number of values is needed.

max-value = max value allowed for the fuzzy set

Example:

```
<!-- WMC -->
<fuzzy-set id="normal-wmc" type="descendent-set" label="normal" value="10,20" max="50" />
<fuzzy-set id="medium-wmc" type="triangle-set" label="medium" value="10,20,30" max="50" />
<fuzzy-set id="high-wmc" type="ascendent-set" label="high" value="20,30" max="50" />
```

Fuzzy rules

Go to `</fuzzy-rules-app>` and/or `<fuzzy-rules-class>` tags and modify the fuzzy rules accordingly.

id= identifies the fuzzy rule, it is a unique value.

If, and, or, not and then tags are available, below an example of a fuzzy rule

```
<fuzzy-rule id='fuzzyrule1'>
<if>
<and>
    <condition metric-id='LCOM2' operation='is' fuzzy-set-id='high-lcom2' />
    <or>
        <condition metric-id='CBO' operation='is' fuzzy-set-id='high-cbo' />
        <condition metric-id='CBO' operation='is' fuzzy-set-id='medium-cbo' />
        <condition metric-id='CBO' operation='is' fuzzy-set-id='normal-cbo' />
    </or>
    <condition metric-id='LOC' operation='is' fuzzy-set-id='high-loc' />
    <condition metric-id='WMC' operation='is' fuzzy-set-id='high-wmc' />
</and>
</if>
<then fuzzy-set-id='critical' />
</fuzzy-rule>
```

Step 4. Execution Parameters

-mode= Required option that accepts two values command or gui

-projectName = Required option and identify the name of the project

-configurationFile = Required option for the location and file name of the configuration file.

-workingDirectory = Required option for the directory that the application will use as a working directory to compile the classes and store logs related to the application

-sourceDirectory = Required option for the directories that contain the source code. It accepts directories and java files with full path separated by commas.

-outputFile = Required option for the command mode to generate the fuzzy report for the classification of the application and the java classes.

-metricOutputLocation = optional option to generate the OO metric report

-filter = optional option to filter class like test, exception, etc

-additionalClasspath= optional option to include packages needed for the application to compile its source code.

Example command

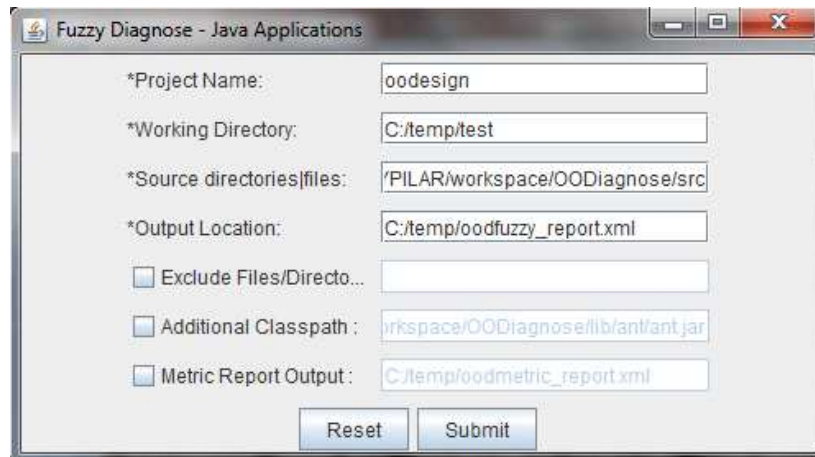
```
java -classpath oodiagnosis.jar
```

```
edu.umd.msswe.ece591.diagnose.fuzzysystem.OODiagnose -  
mode=command -projectName=oodesign -  
configurationFile=C:/Users/JUANYPILAR/workspace/OODiagnose/c  
onfig_system_v2.0.xml -workingDirectory=C:/temp/test -  
sourceDirectory=C:/Users/JUANYPILAR/workspace/OODiagnose/src  
-outputFile="C:/temp/oodfuzzy_report.xml" -  
metricOutputLocation="C:/temp/oodmetric_report.xml" -  
filter=test,Test,Exception -additionalClasspath="C:/Program  
Files  
(x86)/Java/jdk1.6.0_21/lib/tools.jar;C:/Users/JUANYPILAR/wor  
kspace/OODiagnose/lib/junit/junit-  
4.8.2.jar;C:/Users/JUANYPILAR/workspace/OODiagnose/lib/mocki  
to/mockito-all-  
1.8.5.jar;C:/Users/JUANYPILAR/workspace/OODiagnose/lib/ant/a  
nt.jar"
```

2. User Manual

Below is the initial window of the GUI application. The labels with start prefiz are required. Once all the parameters are added then click on submit button. Reset button will blank out the fields.

Initial Window - OO Diagnose Application



Fuzzy Diagnose - Java Applications

*Project Name: oodesign

*Working Directory: C:/temp/test

*Source directories/files: /PILAR/workspace/OODiagnose/src

*Output Location: C:/temp/oodfuzzy_report.xml

☐ Exclude Files/Directo...

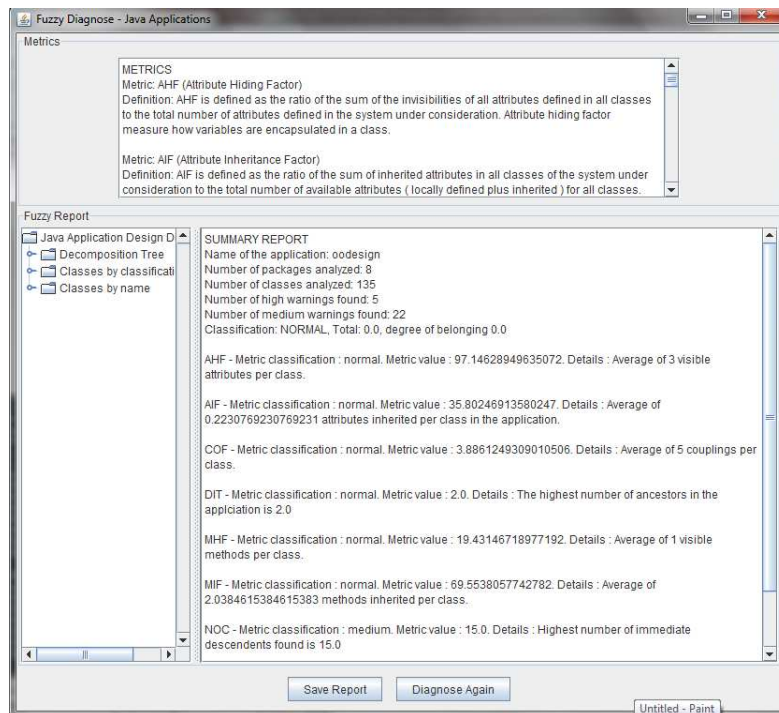
☐ Additional Classpath : /rkspase/OODiagnose/lib/ant/ant.jar

☐ Metric Report Output : C:/temp/oodmetric_report.xml

Reset Submit

Report Window - OO Diagnose Application

Once submit button is hit then the application will take a couple of minutes to compile and analyze the code, extract the metrics and generate the fuzzy report.



Fuzzy Diagnose - Java Applications

Metrics

METRICS

Metric: AHF (Attribute Hiding Factor)
Definition: AHF is defined as the ratio of the sum of the invisibilities of all attributes defined in all classes to the total number of attributes defined in the system under consideration. Attribute hiding factor measure how variables are encapsulated in a class.

Metric: AIF (Attribute Inheritance Factor)
Definition: AIF is defined as the ratio of the sum of inherited attributes in all classes of the system under consideration to the total number of available attributes (locally defined plus inherited) for all classes.

Fuzzy Report

Java Application Design D

- Decomposition Tree
- Classes by classificati
- Classes by name

SUMMARY REPORT

Name of the application: oodesign
Number of packages analyzed: 8
Number of classes analyzed: 135
Number of high warnings found: 5
Number of medium warnings found: 22
Classification: NORMAL, Total: 0.0, degree of belonging 0.0

AHF - Metric classification : normal. Metric value : 97.14628949635072. Details : Average of 3 visible attributes per class.

AIF - Metric classification : normal. Metric value : 35.80248913580247. Details : Average of 0.2230769230769231 attributes inherited per class in the application.

COF - Metric classification : normal. Metric value : 3.8861249309010506. Details : Average of 5 couplings per class.

DIT - Metric classification : normal. Metric value : 2.0. Details : The highest number of ancestors in the application is 2.0

MHF - Metric classification : normal. Metric value : 19.43146718977192. Details : Average of 1 visible methods per class.

MIF - Metric classification : normal. Metric value : 69.5538057742782. Details : Average of 2.0384615384615383 methods inherited per class.

NOC - Metric classification : medium. Metric value : 15.0. Details : Highest number of immediate descendents found is 15.0

Save Report Diagnose Again

Untitled - Paint

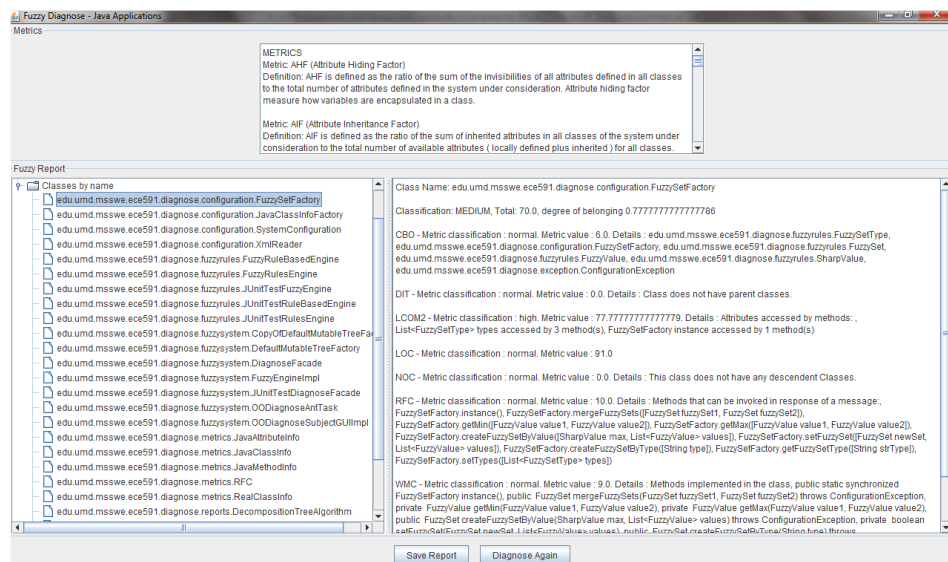
Decomposition tree – OO Diagnose Application

The screenshot displays the 'FuzzyDiagnose Java Application' window. The 'Metrics' tab is active, showing two metrics:

- Metric: AHF (Attribute Hiding Factor)**: Defined as the ratio of the sum of the invisibilities of all attributes defined in all classes to the total number of attributes defined in the system under consideration. Attribute hiding factor measure how variables are encapsulated in a class.
- Metric: AIF (Attribute Inheritance Factor)**: Defined as the ratio of the sum of inherited attributes in all classes of the system under consideration to the total number of available attributes (locally defined plus inherited) for all classes.

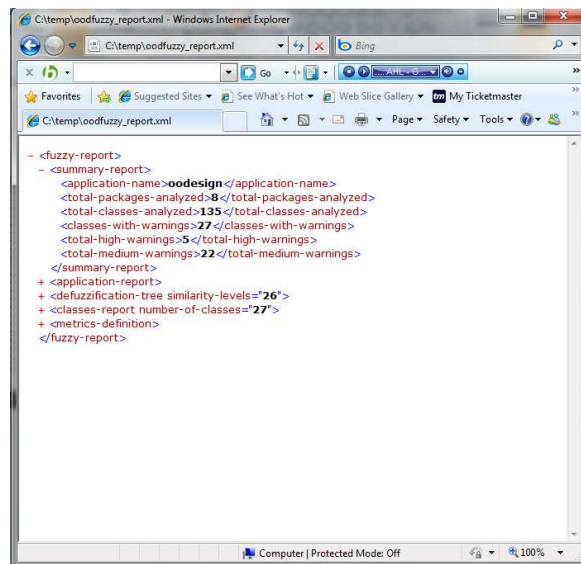
Below the metrics, the 'Fuzzy Report' section shows a 'Decomposition Tree' with various similarity levels and a 'LEVEL SUMMARY:' box containing classification information and class lists.

The “classes by name” tree provides a list of classes order by full path and the “classes by classification” tree provides a list of classes per classification.

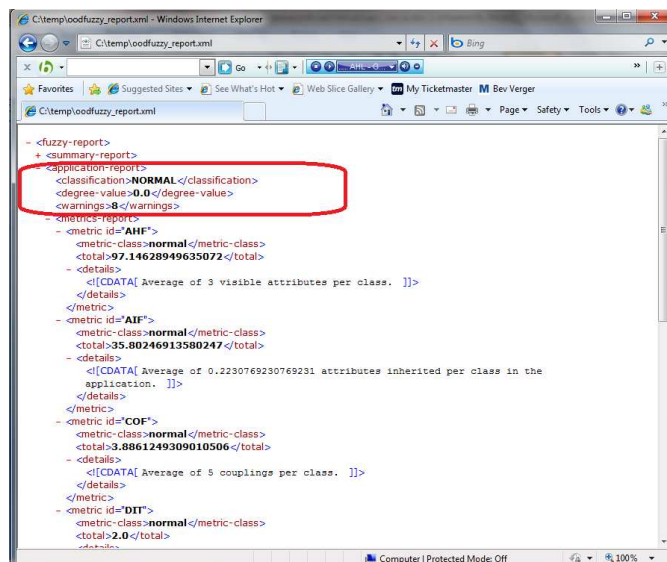


XML Fuzzy Report

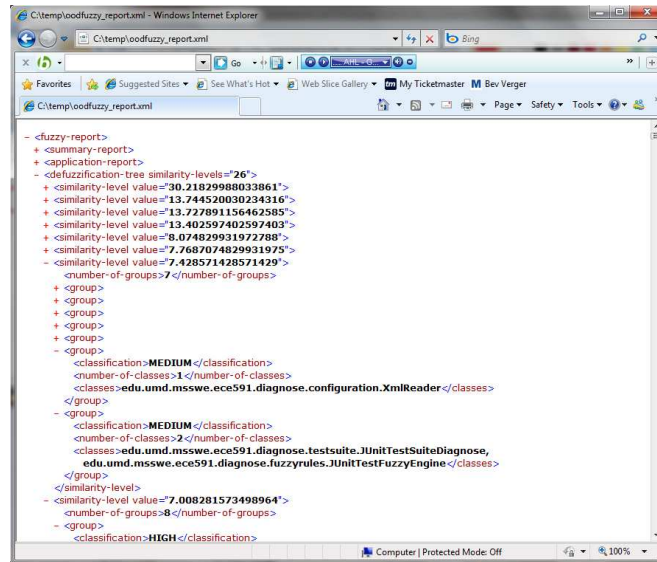
The user can save the report in XML format. This report is also generated by the command mode. It contains 5 main tags: summary report, application report, defuzzification tree report and metrics definition. The summary report provides information of the application being diagnosed like number of packages, number of classes, classes with critical, high and medium classification, etc



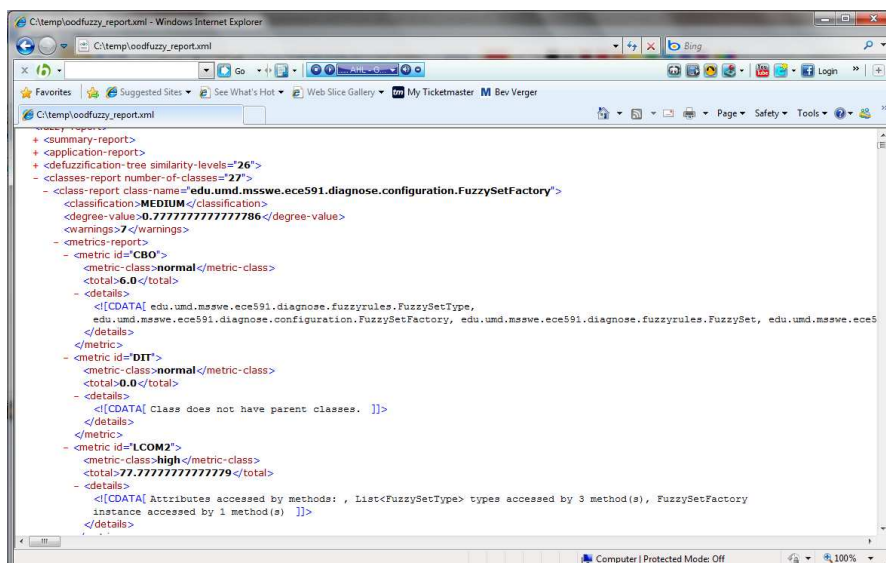
The application report provides detail information of the application, like global classification, values of the metrics, classification per metric and details reported by the metric.



The defuzzification tree provides a tree structure with the similarity levels generated and the classes for each of the similarity groups.



The class report provides information of the classification of the class like categorization of the class, value reported, and a detailed information for each of the metrics utilized to classify the class.



Finally the metrics definition provides description for each of the metrics utilized within the fuzzy system.

