

SOFTWARE ENGINEERING 2 PROJECT

Code Inspection



**POLITECNICO
DI MILANO**

TEAM: NICO MONTALI, ENRICO FINI

Contents

1	Class identification and role	4
2	Found Issues	5
3	Checklist	6
3.1	Naming Conventions	6
3.2	Indention	6
3.3	Braces	6
3.4	File Organization	7
3.5	Wrapping lines	7
3.6	Comments	7
3.7	Java Source Files	7
3.8	Package and Import Statements	7
3.9	8
3.10	Initialization and Declarations	8
3.11	Method Calls	8
3.12	Arrays	8
3.13	Object Comparisons	8
3.14	Output Format	8
3.15	Computation, Comparisons, Assignments	8
3.16	Exceptions	8
3.17	Flow of Control	8
3.18	Files	8

Document Status

Document Title	Code Inspection
Document ID	CI/1.0
Authors	N. Montali, E. Fini
Version	1.0

Changelog

Version	Date	Changes
1.0	01/02/2017	Initial version

Class identification and role

The class we were assigned to inspect is:

../org/apache/ofbiz/base/start/Config.java

Using the **Javadoc** (click on the link to open the URL) we found out that the class main objective was to

OFBiz server parameters needed on system startup and retrieved from one of the properties files in the start component

So it is a simple configuration class, where many parameters are loaded from files (.properties that can be found in the directory) and can be accessed through public and final instance variables.

Found Issues

We found ***** issues, listed here (and available at the corresponding checklist entry).

Checklist

In this Chapter, every step of the Code Inspection checklist is documented.

3.1 Naming Conventions

- 1 Variables names are meaningful
- 2 One-char variables are used only as exceptions.
- 3 Class names are mixed case with first uppercase letter.
- 4 No interface is declared here
- 5 All method names start with verbs and have mixed case
- 6 All class variable have mixed with starting lower case char (no underscores either)
- 7 This class has no constants

3.2 Indention

- 8 Four space are used to indent every time
- 9 No tab are used in the file

3.3 Braces

- 10 All braces are in K&R style.
- 11 Every single statement block is braced correctly

3.4 File Organization

- 12 Section are well separated by spaces
- 13 **Many lines exceed the 80 char limit even if not necessary.** Lines: 74, 86, 114, 116, 121, 198, 201, 228, 249, 268, 270, 278, 280 can be written more elegantly. Other lines exceed 80 char limit but can be accepted.
- 14 **Some lines also exceed 120 char limit.** Lines: 74, 228, 268, 270, 280

3.5 Wrapping lines

- 15 **Some operators are inserted after the line break.** Lines: 163, 164, 165, 166
- 16 High-level breaks are used.
- 17 **New statements are not aligned with the beginning of the previous line at the same level** Lines: 162, 163, 164, 165, 166

3.6 Comments

- 18 **Comments could explain more.** Some method's function could be better explained even if quite simple.
- 19 There is no commented out code.

3.7 Java Source Files

- 20 This java contains only one Java Class.
- 21 The public class is the first class of the file
- 22 As in the Javadoc, this Class does not expose any external interface, only public variables to be read-only.
- 23 The Javadoc is practically complete.

3.8 Package and Import Statements

- 24 The first non comment line is the package, all the imports follows.

3.9

Class and Interface Declarations

- 25 Declarations follow the order provided.
- 26 Since methods perform very different tasks, it is difficult to find a correct order.
- 27 Code is free of duplicates, long methods and big classes.

3.10 Initialization and Declarations

28

3.11 Method Calls

3.12 Arrays

3.13 Object Comparisons

3.14 Output Format

3.15 Computation, Comparisons, Assignments

3.16 Exceptions

3.17 Flow of Control

3.18 Files