Prof. Lichter

Andreas Steffens, Konrad Fögen

OOSC
SoSe 2019

Submission 2

oosc@swc.rwth-aachen.de

Submitted by Group 09

Ulfet CETIN	391819
Saud KHAN	392365
Samuel ROY	391822
Charulekha, Besta Venkateswara RAO	391844
Deepak SATEESH	391813

(ordered on lastname basis)

1.1. Apply Metrics to a complex software project

java-design-patterns:

- McCabe Cyclometric Complexity

Cyclomatic Complexity	1.39	1.06	1.00

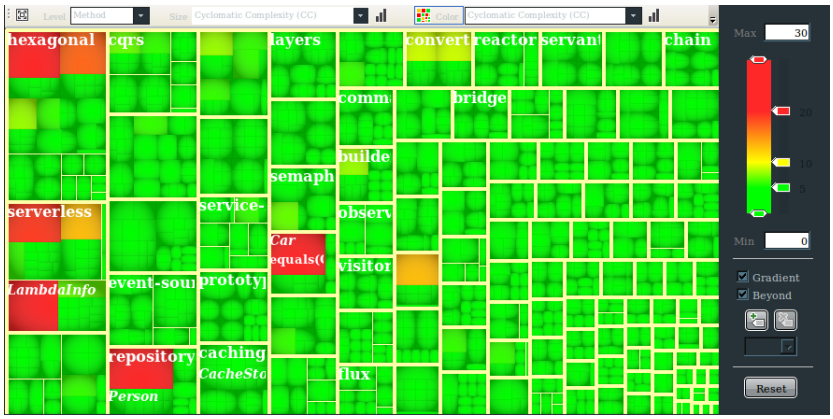

Figure 1.1: java-design-patterns - Cyclomatic Complexity (taken using jArchitect)

- Chidamber & Kemerer:

Metric Name	Arithmetic Avg.	StdDev.	Median
Number of Methods (per Class) (in place of Weighted Methods per Class, as we take system average)	1.78	1.31	1.00
Fan-In Visibility (System) (in place of Number of Children -NOC-) (percentage of internal components in the system that depend directly or indirectly on other components, system-wise)	0.48	0.39	0.41
Depends Upon (System) Average Component Dependency (ACD) (both in place of Coupling between Objects -CBO-)	3.57 2.92	2.90 1.29	3.0 2.71
Response For a Class (RFC) (There is no counterpart in SonarGraph)			
Lack of Cohesion in Methods (LCOM) (There is no counterpart in SonarGraph)			

- Robert Martin Metrics:

Metric Name	Arithmetic Avg.	StdDev.	Median
Number of Incoming Dependencies (System)	0.74	1.22	0.0
Number of Outgoing Dependencies (System)	0.74	1.15	0.0
Instability	0.77	0.33	1.0
Abstractness	0.18	0.22	0.14

* Unless stated otherwise, data is gathered from SonarGraph Explorer.