



HELLENIC
REPUBLIC

UNIVERSITY
OF MACEDONIA

Software Engineering Group
se.uom.gr



Who is Producing More Technical Debt?

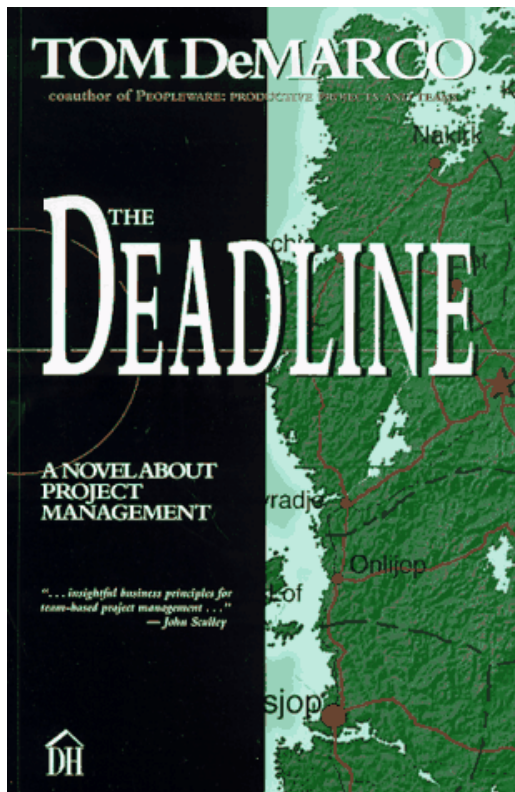
A Personalized Assessment of TD Principal

*Theodoros Amanatidis, Alexander Chatzigeorgiou,
Apostolos Ampatzoglou and Ioannis Stamelos*

University of Macedonia, Aristotle University of Thessaloniki

Context & Goal

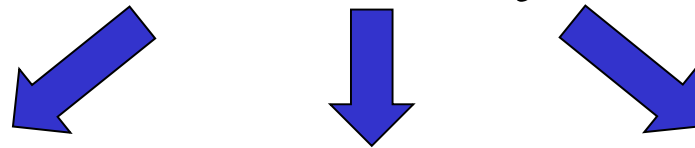
- *TD reduces the velocity during evolution*
- *TD can be assessed on artifacts*
- *However, it's **people** that do projects*



..and it's people that introduce TD

Context & Goal

Case Study



**distribution of TD
among developers**
RQ1

**Violation types
per developer**
RQ2

**Relation between
TD and maturity**
RQ3

- *Managers can steer the allocation of tasks*
- *Developers get input for self-improvement*

A note on Ethics

Processing information at the level of individual developers should be performed with care.

In this study gathered personal data has been de-identified.



Assessing the contribution of developers to the system's TD should not share any kind of personal data with third parties.

Any type of performance analysis should respect ethics, ensuring for example that:

- developers are aware of the relevant process**
- any feedback will be accessible by the employees**
- and will remain confidential.**

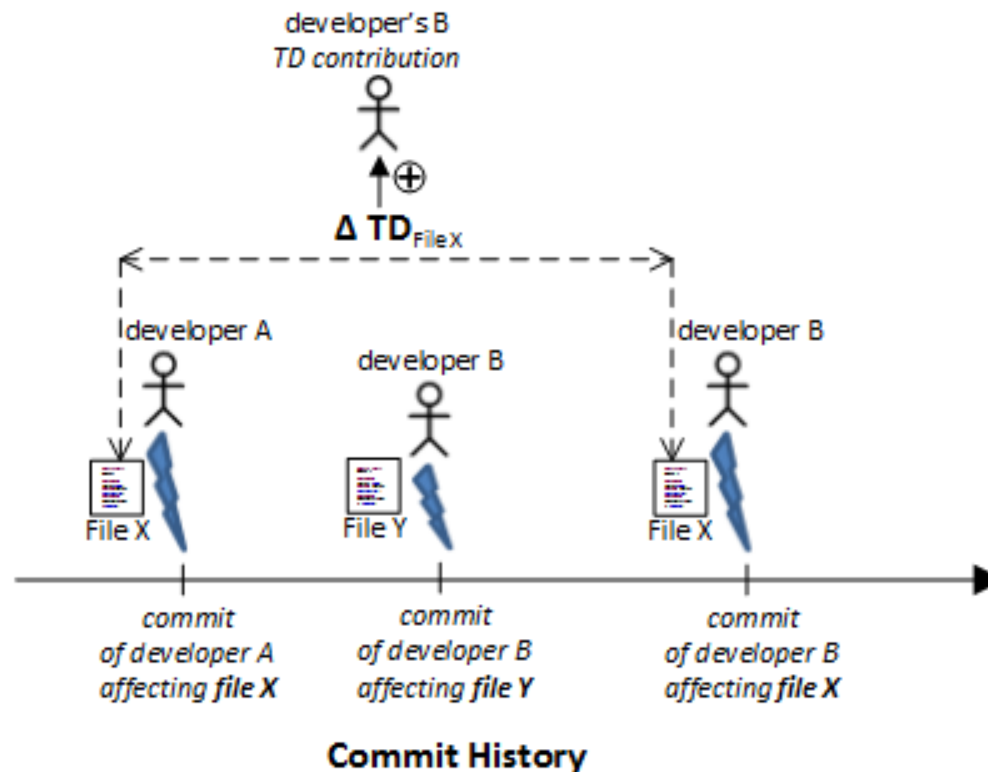
Case Study Design - Projects

Project	#Commits	#Developers (considered)	Size of last version (LOC)	year of 1st release
Laravel (core)	1136	11	149K	2013
Composer	807	7	8K	2012
Yii2	2097	19	406K	2013
Cakephp	1677	23	297K	2008

Case Study Design - Variables

[V1] DevID

[V2] Total TD **sonarqube**



[V3] Number of modified lines → [V4] Normalized TD

Case Study Design - Variables

[V5] Types of TD violations

S104	Files should not have too many lines	✓	brain-overload
S1066	Collapsible "if" statements should be merged	✓	clumsy
S1067	Expressions should not be too complex	✓	brain-overload
S1068	Unused private fields should be removed	✓	unused
S107	Functions should not have too many parameters	✓	brain-overload
S1117	Local variables should not have the same name as class fields		pitfall
S1134	"FIXME" tags should be handled	✓	
S1142	Functions should not contain too many return statements	✓	brain-overload

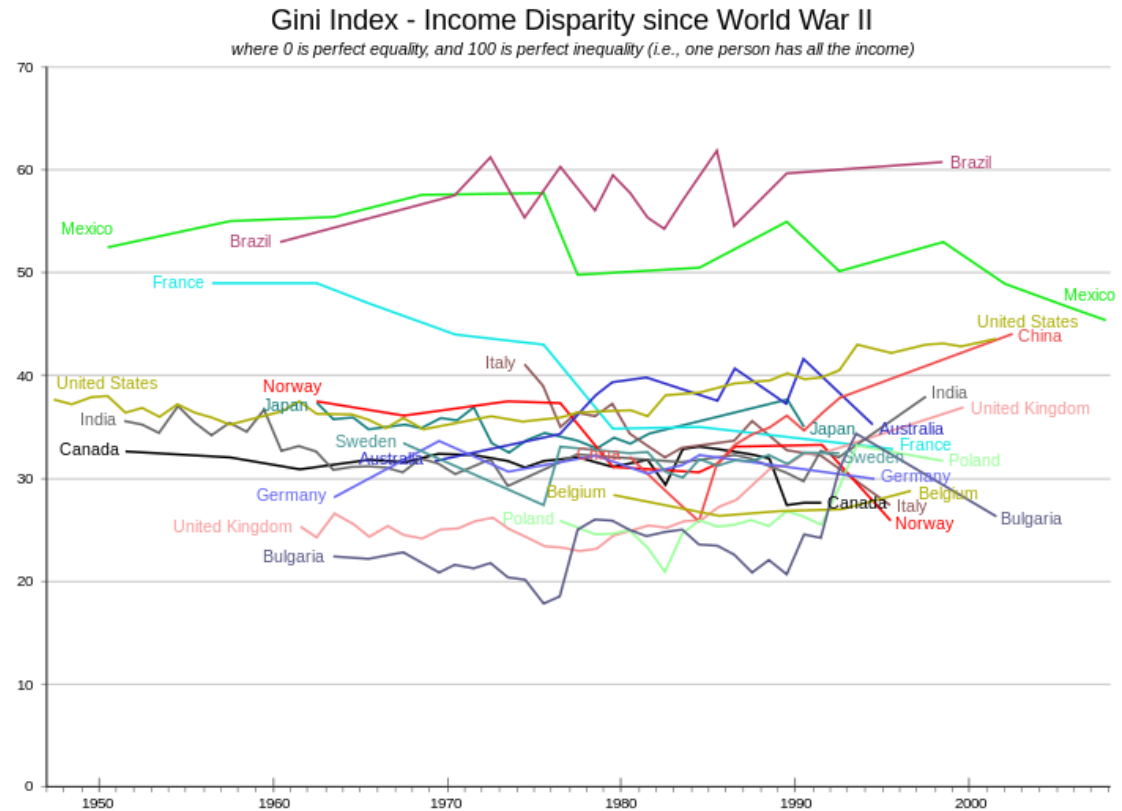
[V6] Developer Maturity



Case Study Design – Data Analysis

RQ1 Distribution of TD among developers

- *Bar charts*
- *Gini Coefficient*



- *Anderson Darling test*

Case Study Design – Data Analysis

RQ2: Which TD violations are introduced by the developers of a software project?

- *HeatMap*

RQ3: relation between TD and maturity

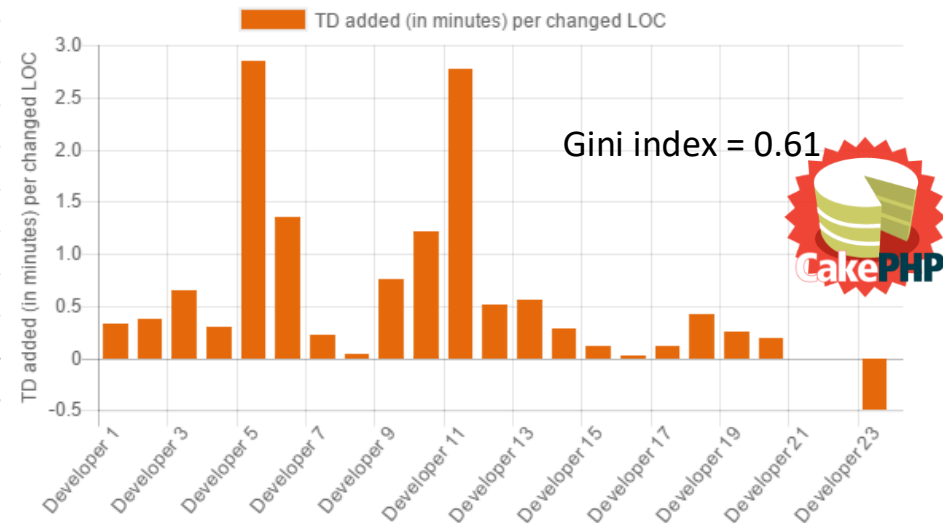
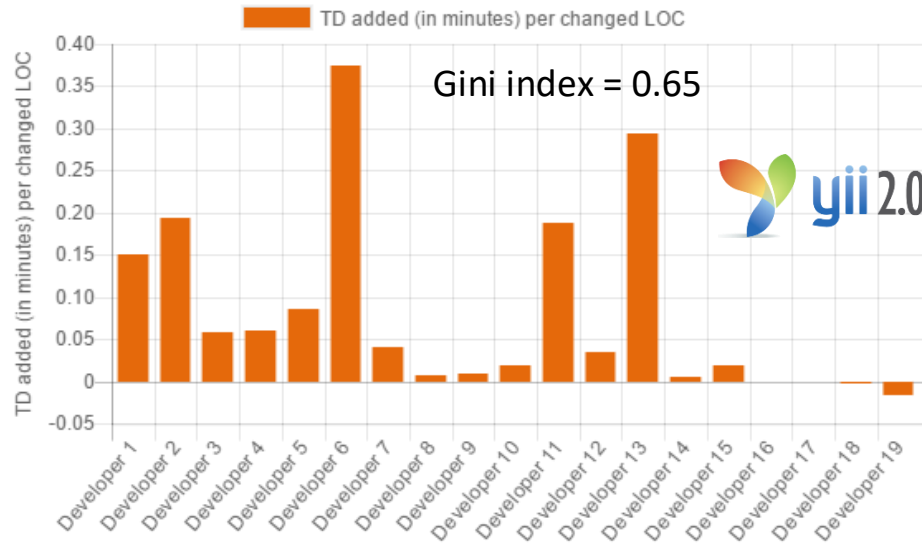
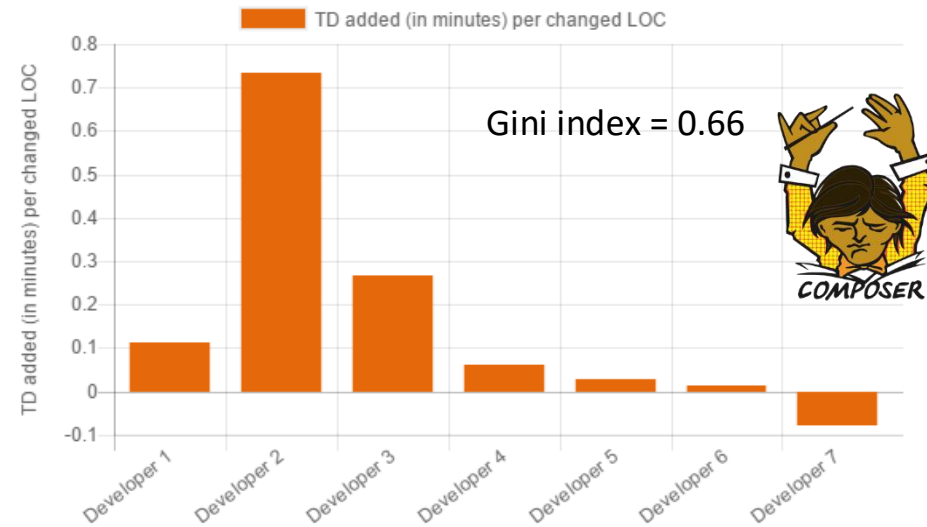
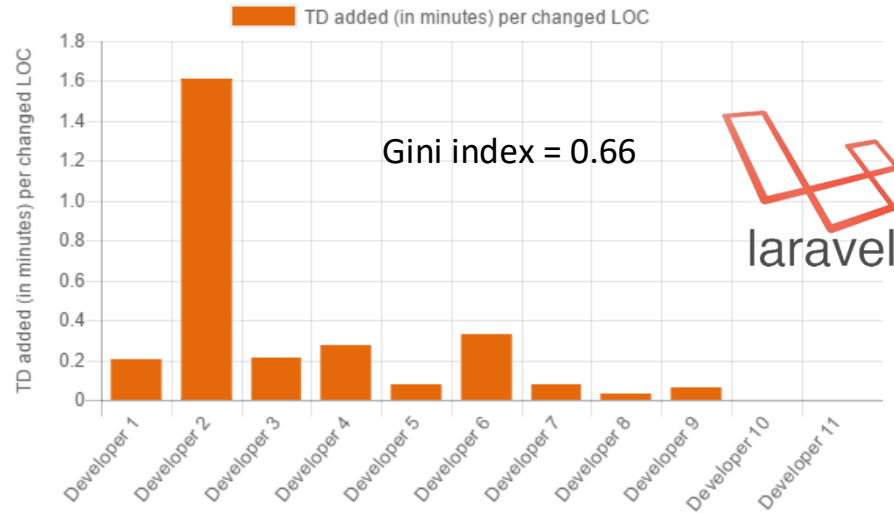
Combined Dataset for all projects

To avoid bias variables are expressed as % :

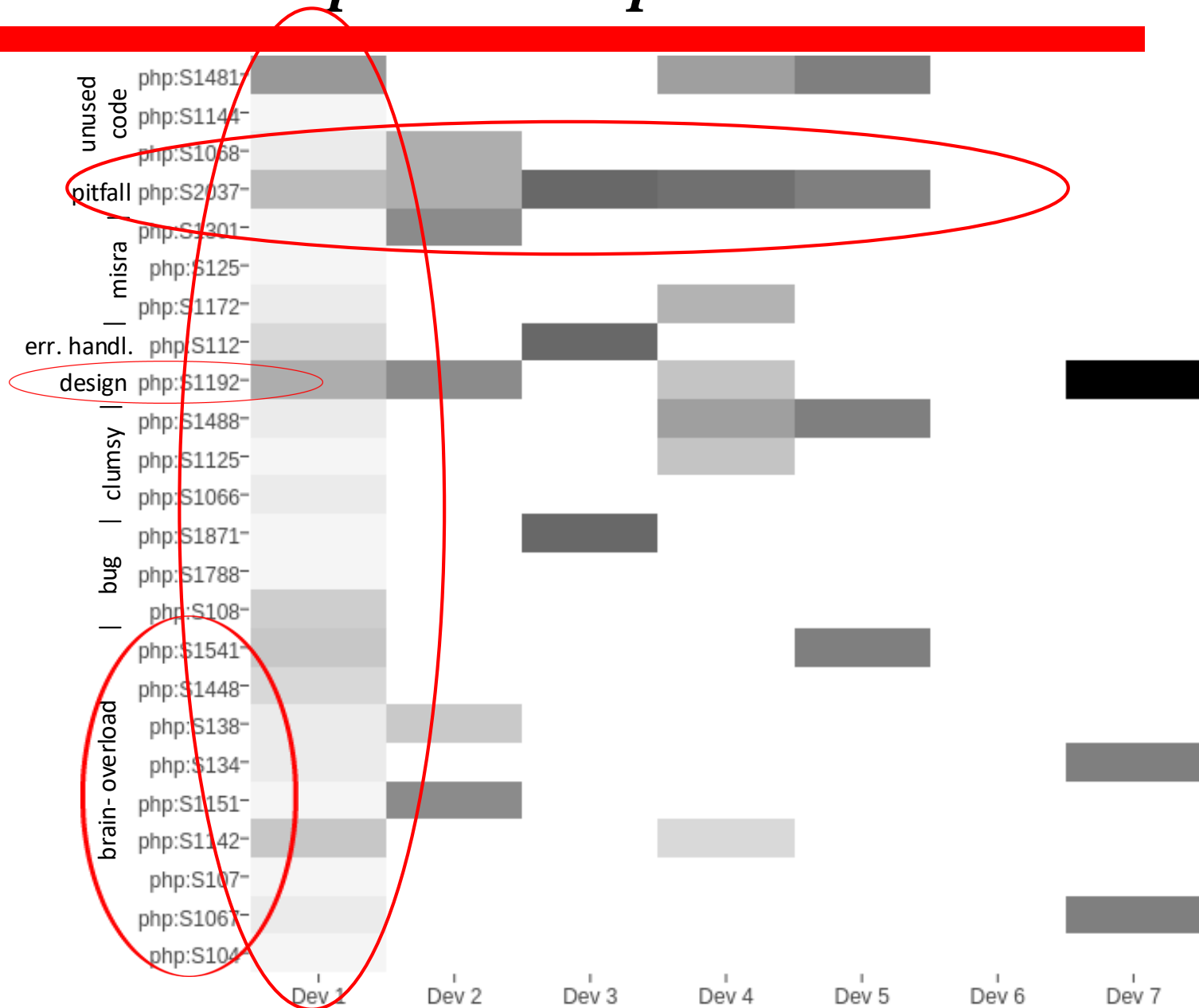
maturity of each developer divided with the maturity of the most experienced one
normalized TD for each developer divided by the maximum normalized TD

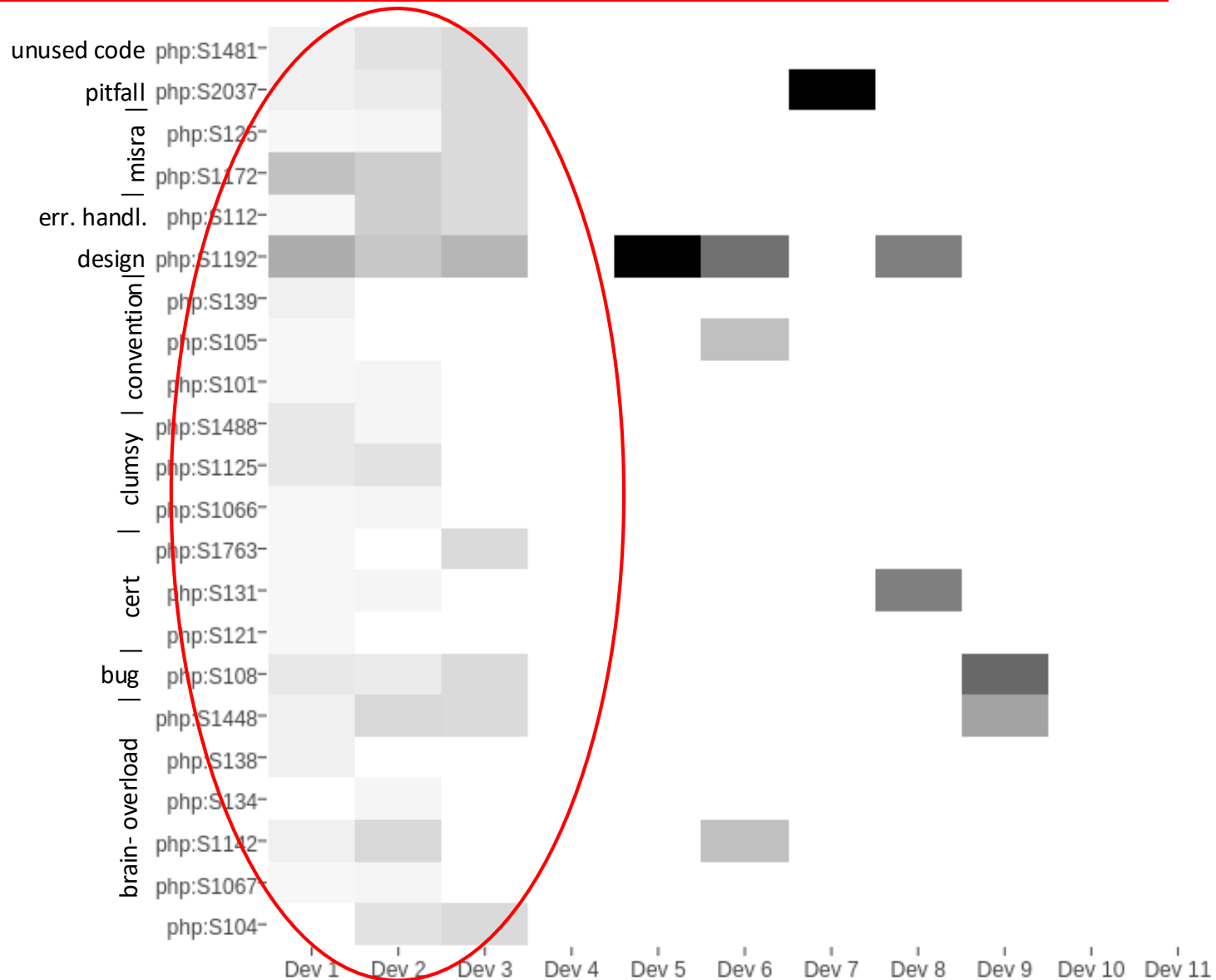
- *ScatterPlot*
- *Correlation Analysis*
- *Independent sample t-test*

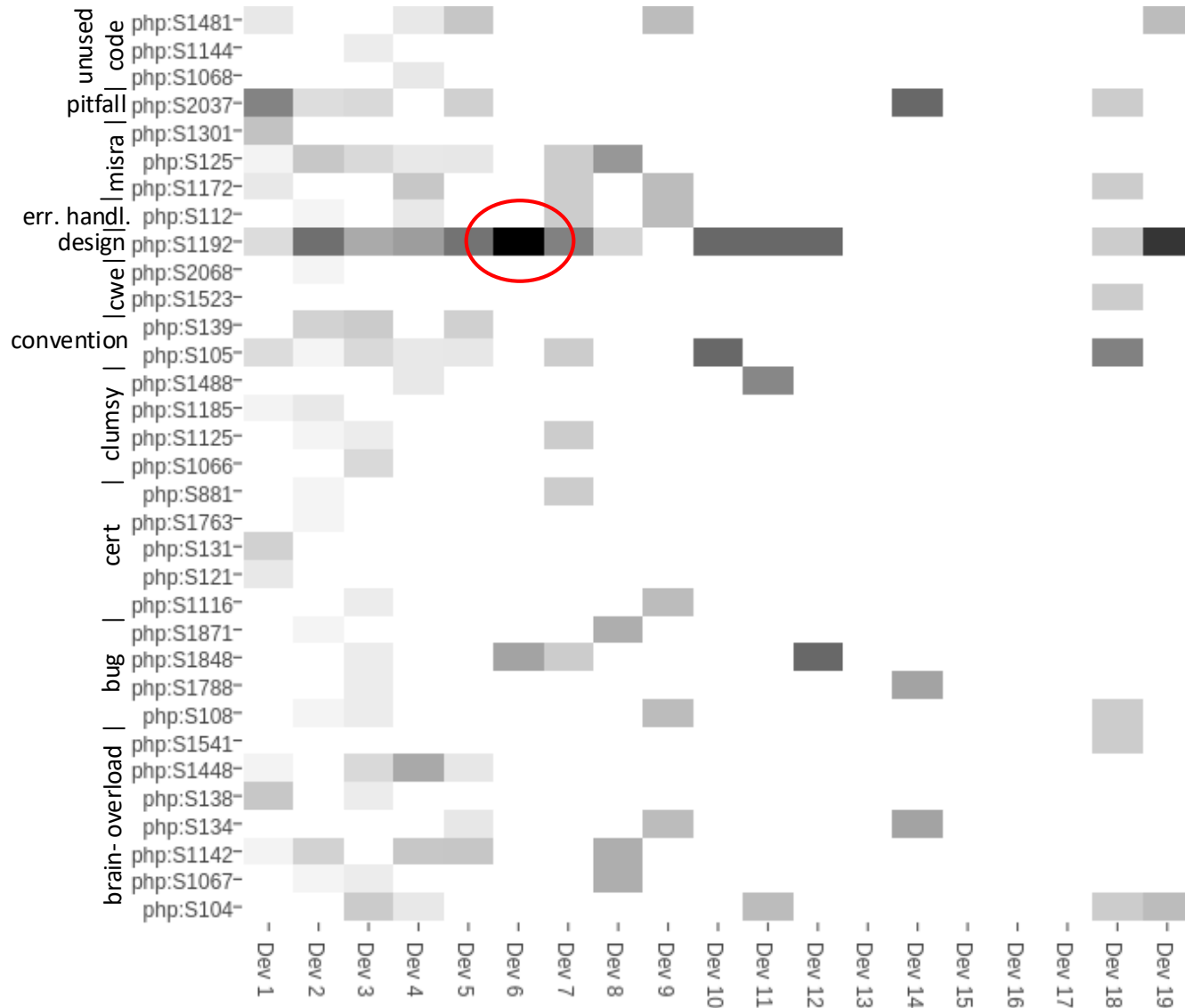
Distribution of TD among Developers



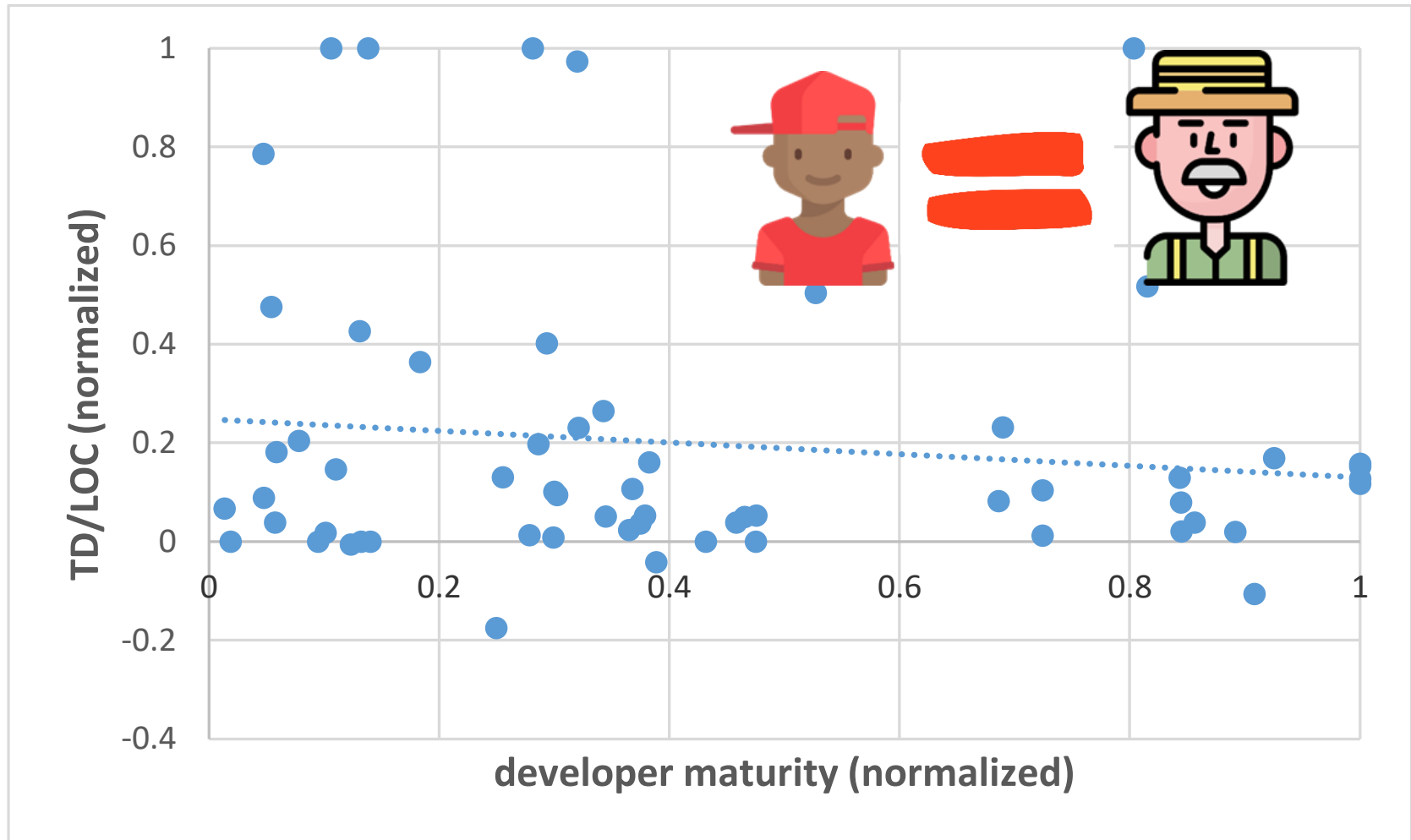
TD violations per Developer







TD vs. Developer Maturity



to conclude

- developers' competencies vary, since the **distribution of TD is highly imbalanced**
- different developers introduce different TD violations
- some recurring violations can be identified across developers and projects
- there is no statistically significant evidence that more experienced developers introduce less TD



HELLENIC
REPUBLIC

UNIVERSITY
OF MACEDONIA

Software Engineering Group
se.uom.gr



Thank you for your attention!