

Noname manuscript No.
(will be inserted by the editor)

Causal inference of server- and client-side code smells in web apps evolution - Appendix 1 - Causal Inference Extended Data I - Graphics and tables up to lag4

Replication package

Address(es) of author(s) should be given

1 RQ1 - Server- and client-side Code Smell evolution

1.1 Server- and client-side CS + metrics evolution extended graphics

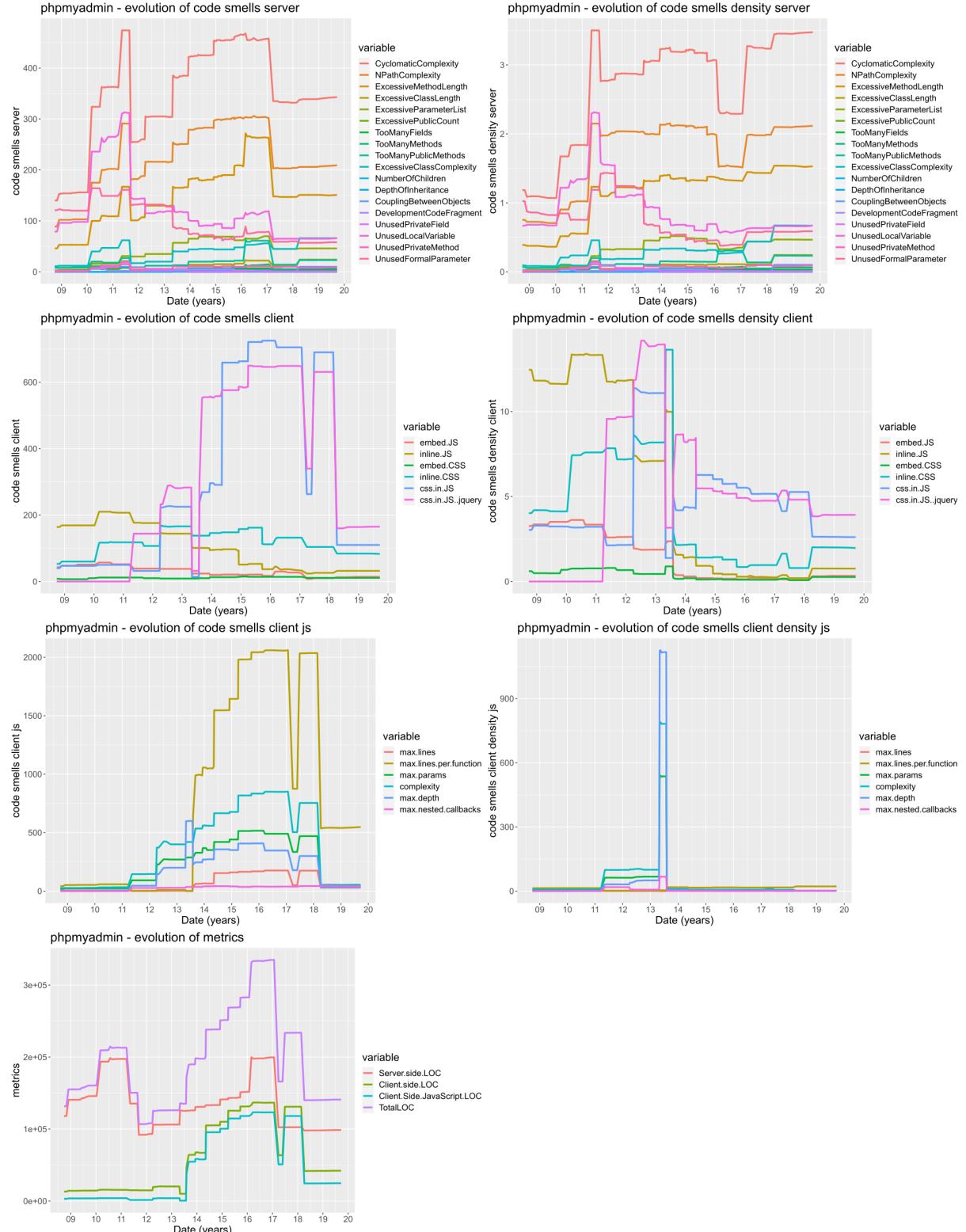


Fig. 1: CS and metrics evolution for phpMyAdmin

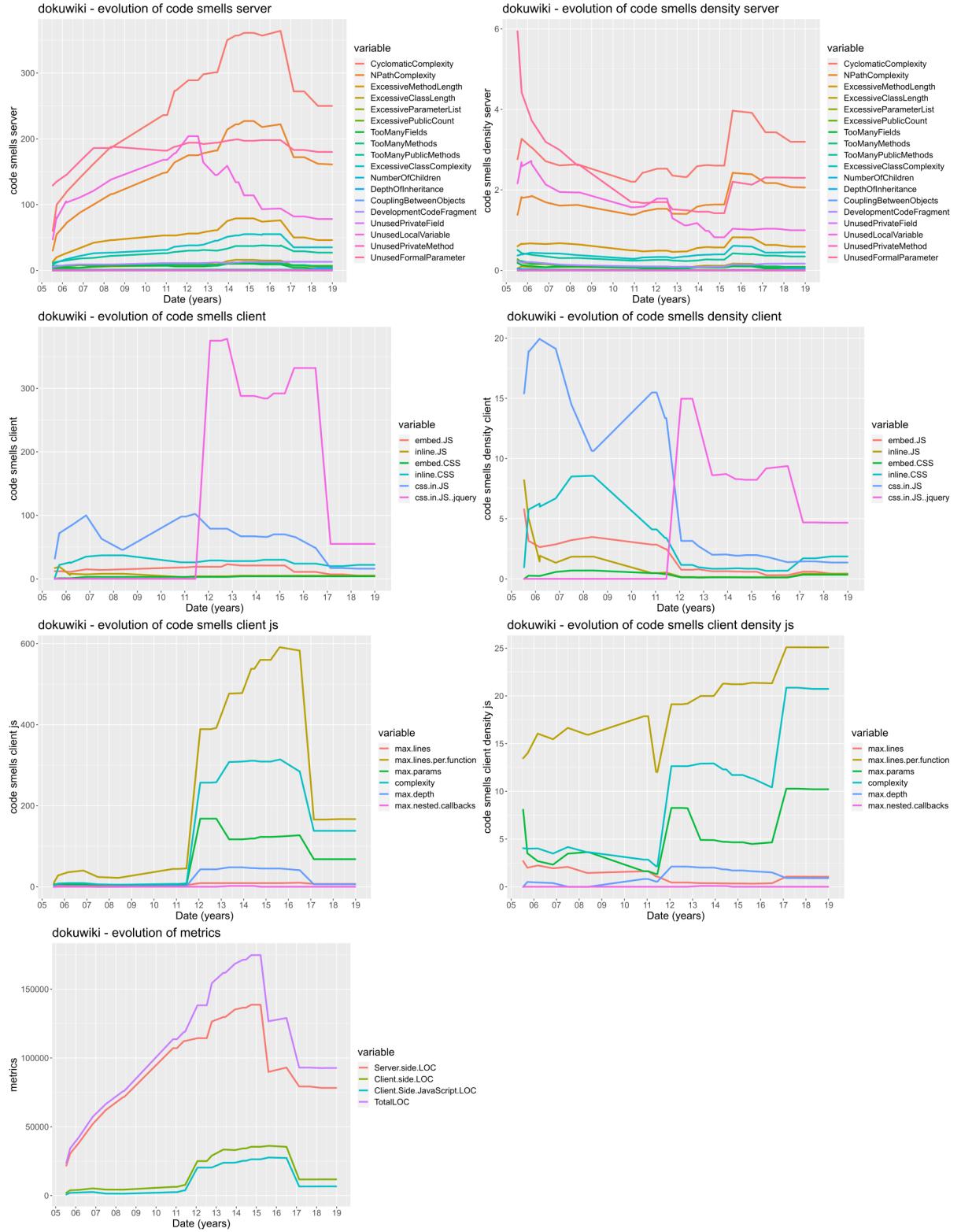


Fig. 2: CS and metrics evolution for DokuWiki

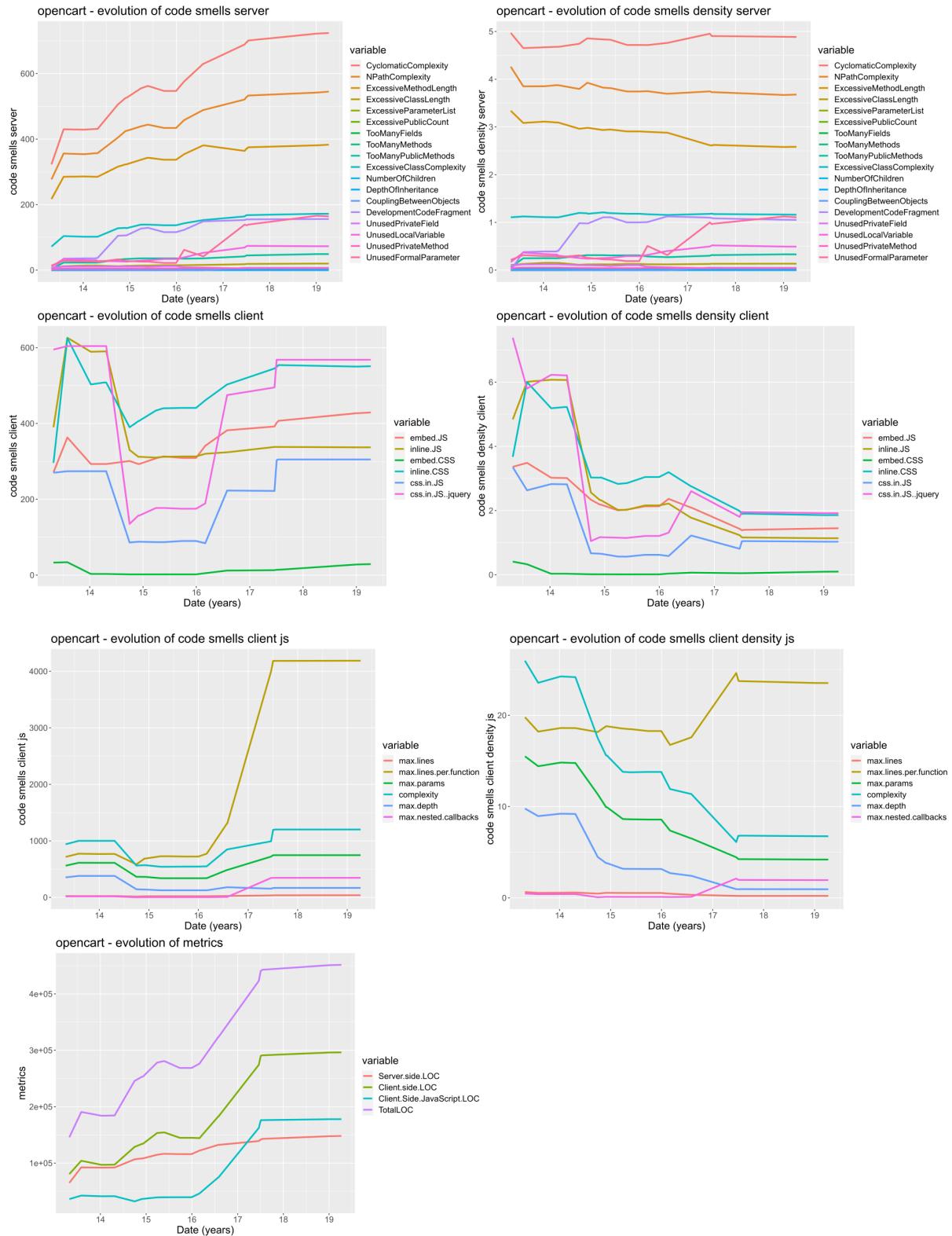


Fig. 3: CS and metrics evolution for OpenCart

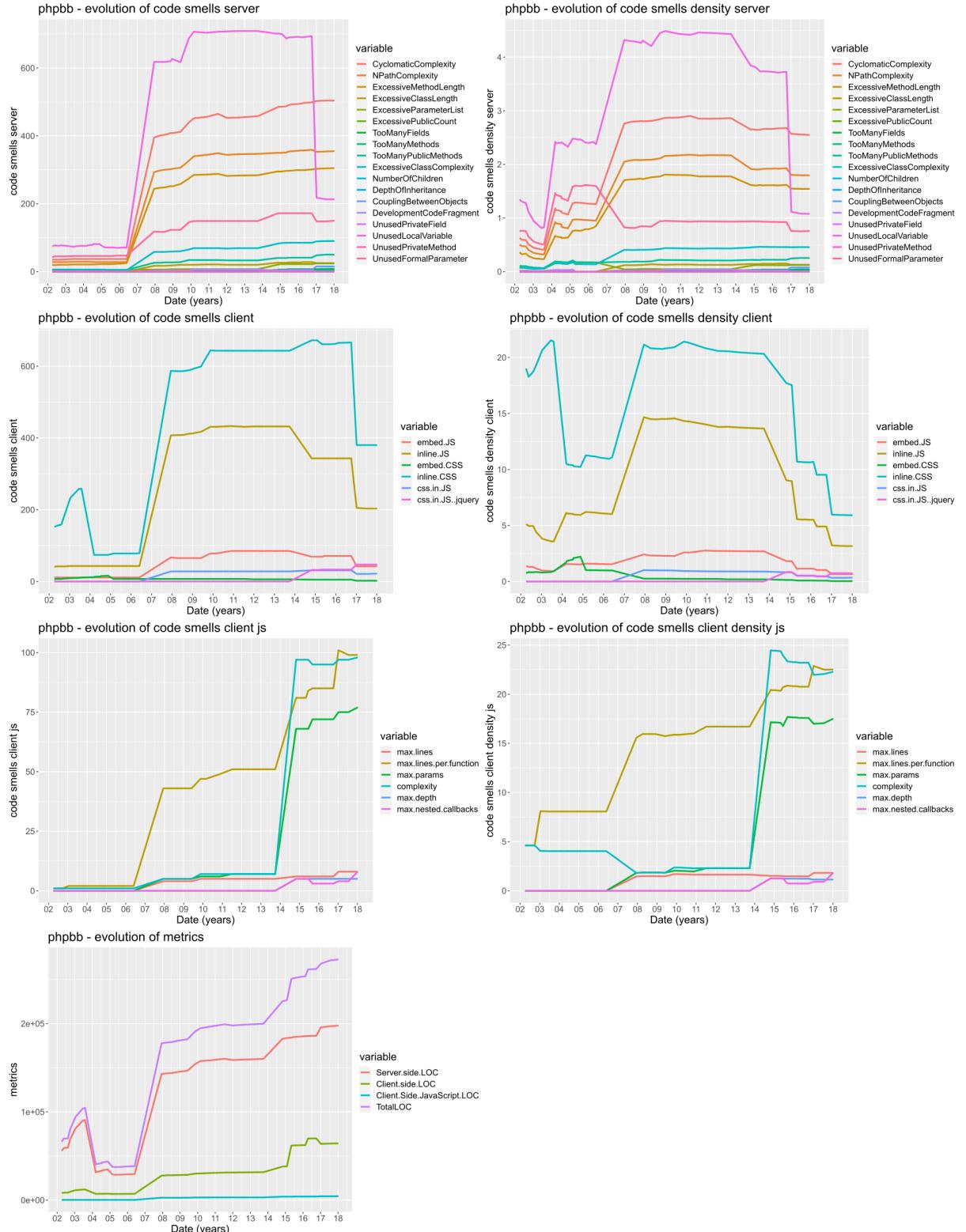


Fig. 4: CS and metrics evolution for phpBB

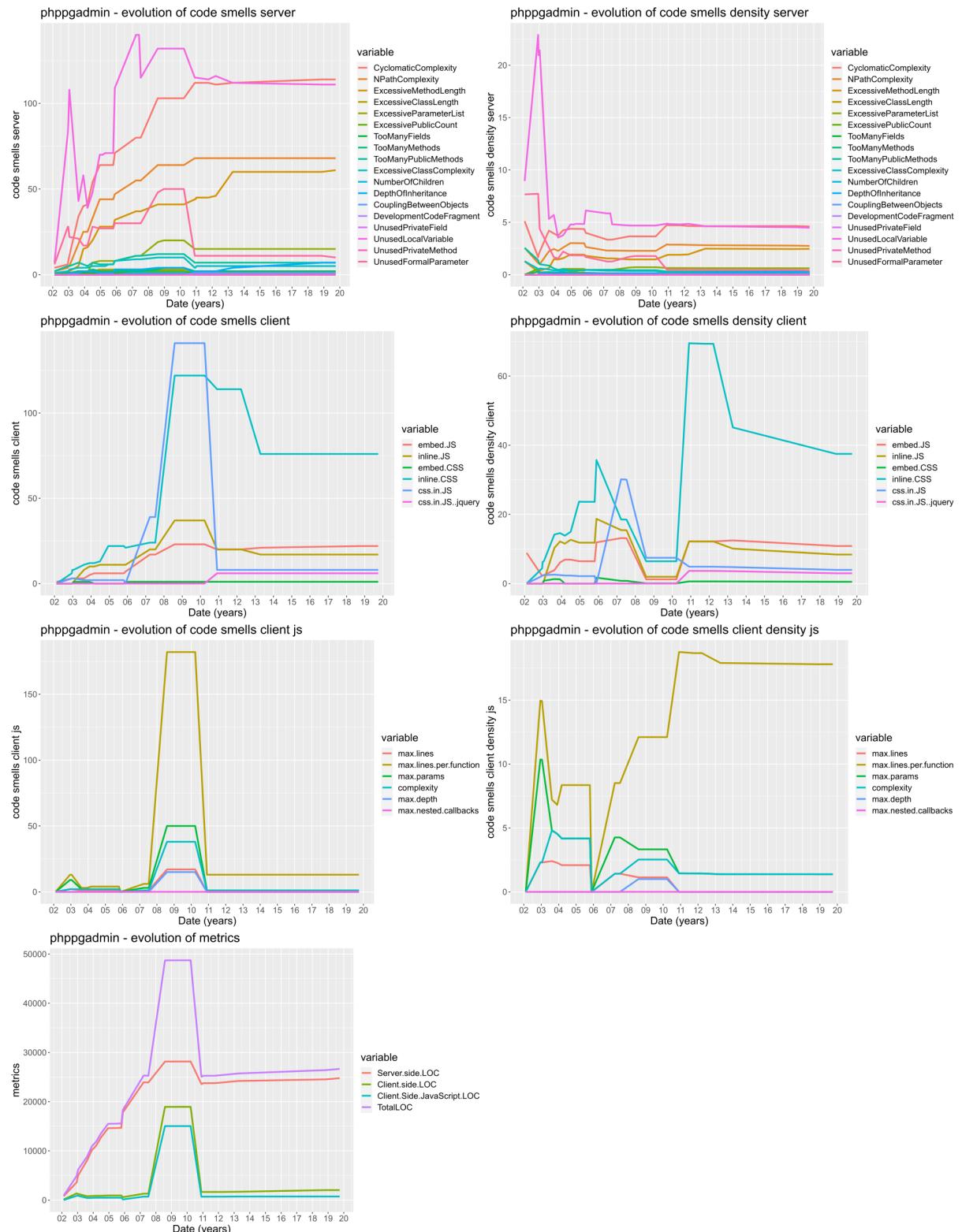


Fig. 5: CS and metrics evolution for phpPgAdmin

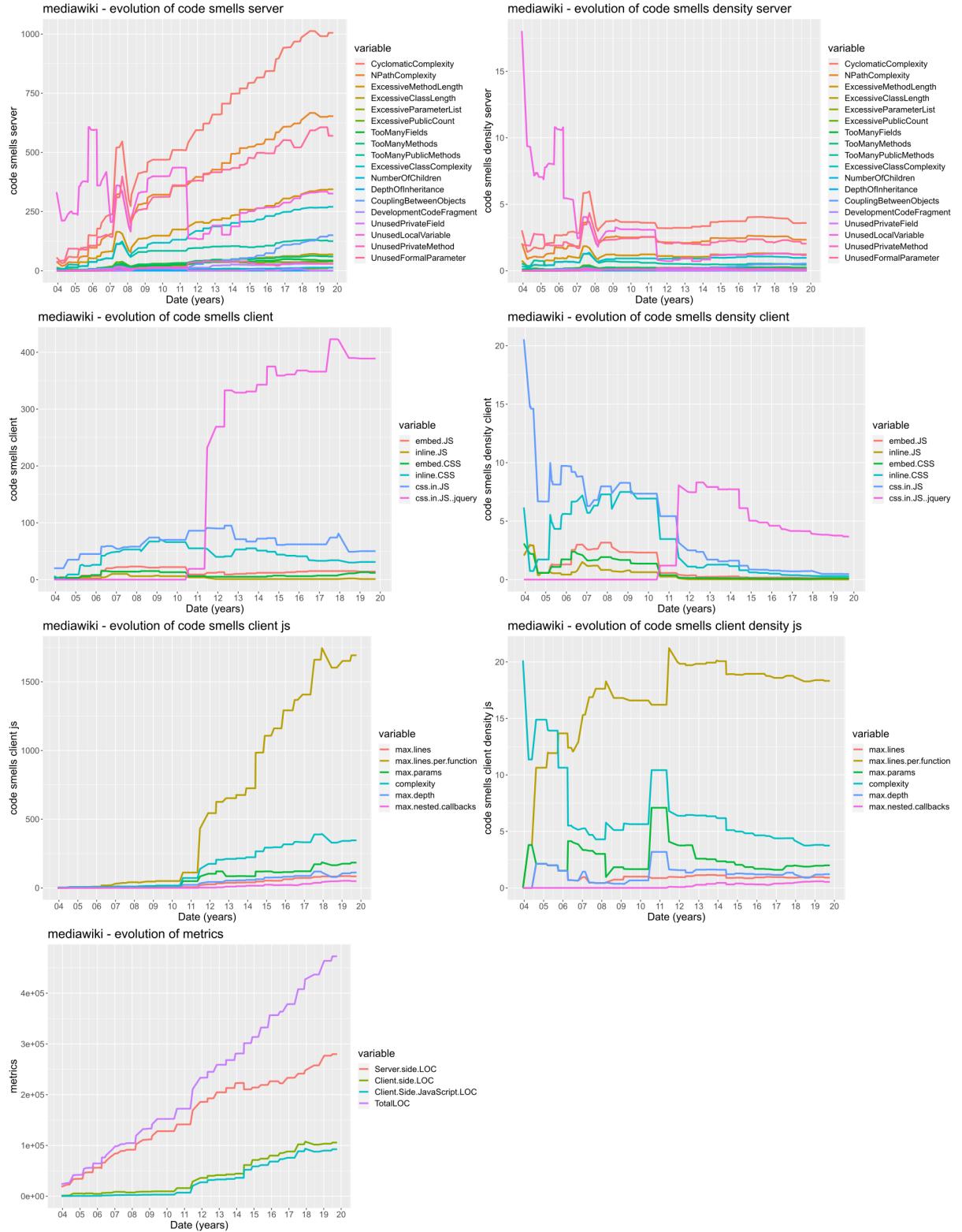


Fig. 6: CS and metrics evolution for MediaWiki

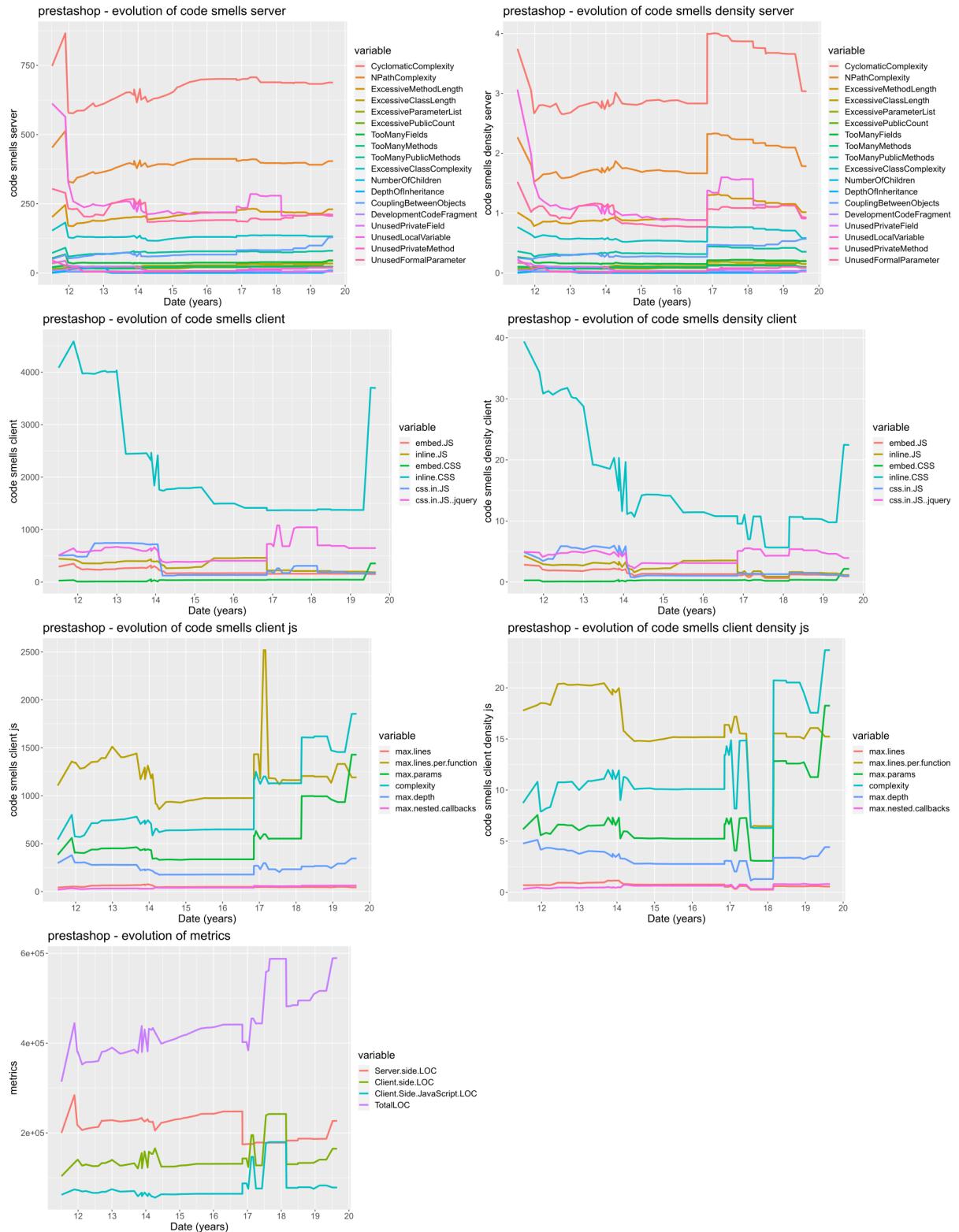


Fig. 7: CS and metrics evolution for PrestaShop

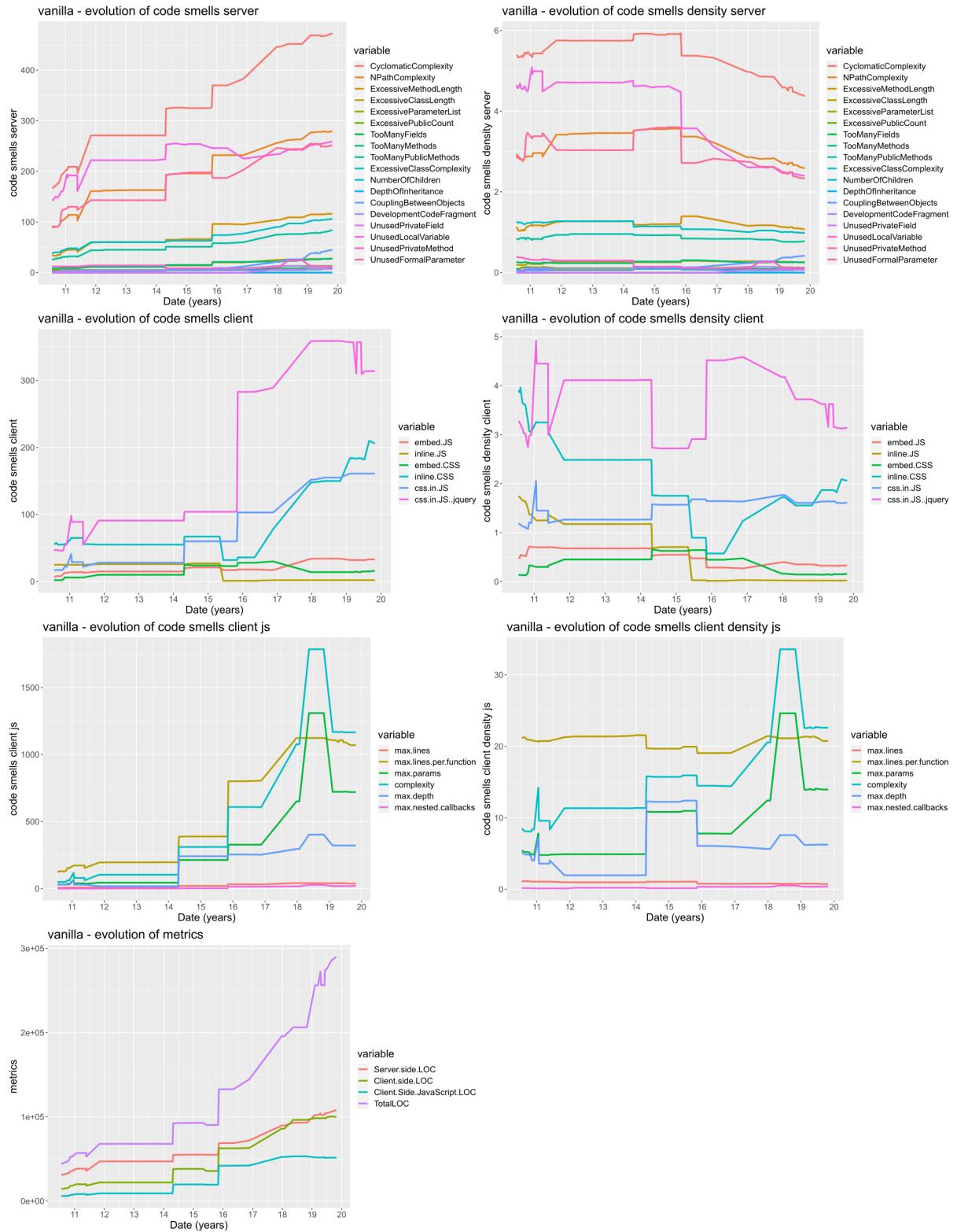


Fig. 8: CS and metrics evolution for Vanilla

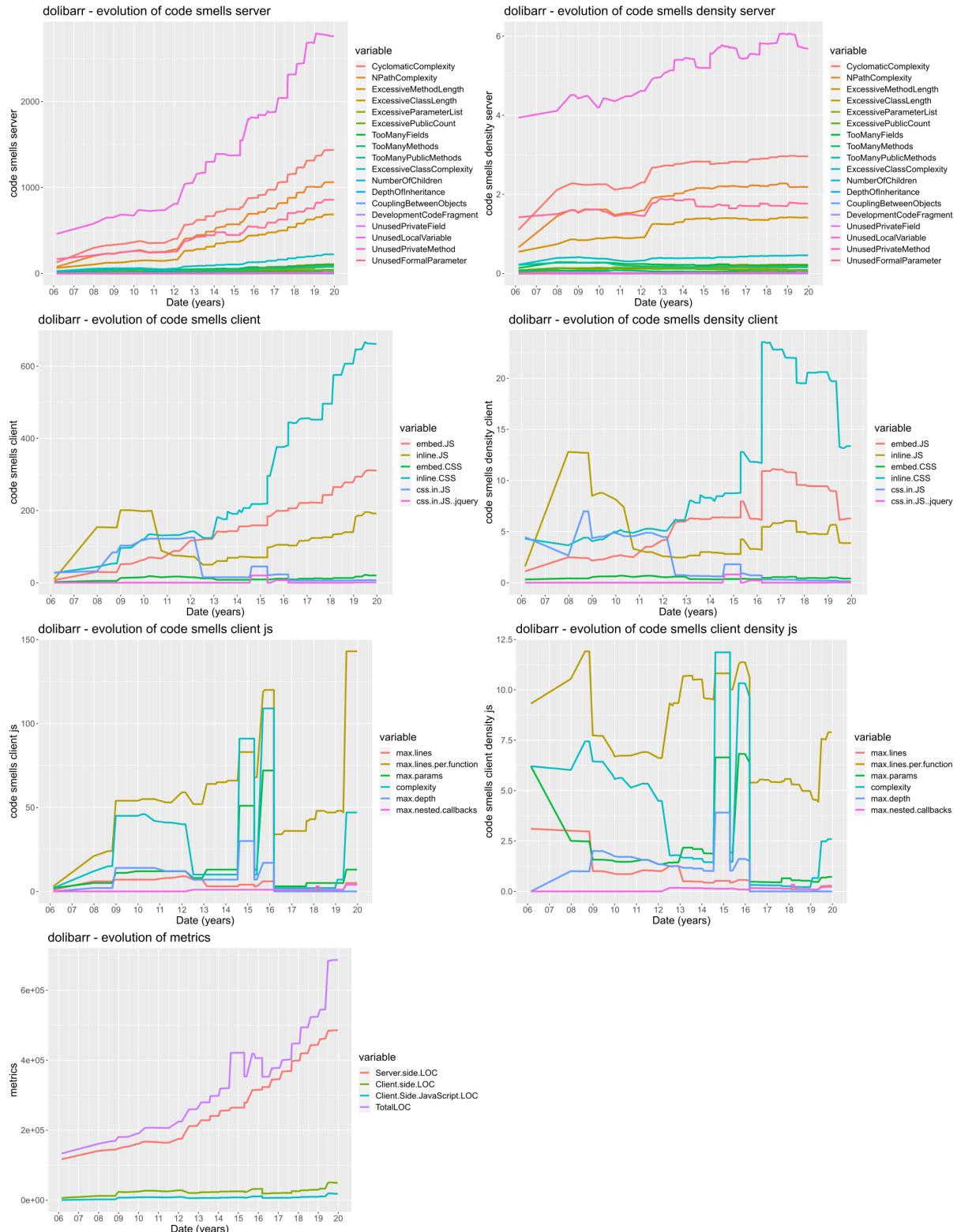


Fig. 9: CS and metrics evolution for Dolibarr

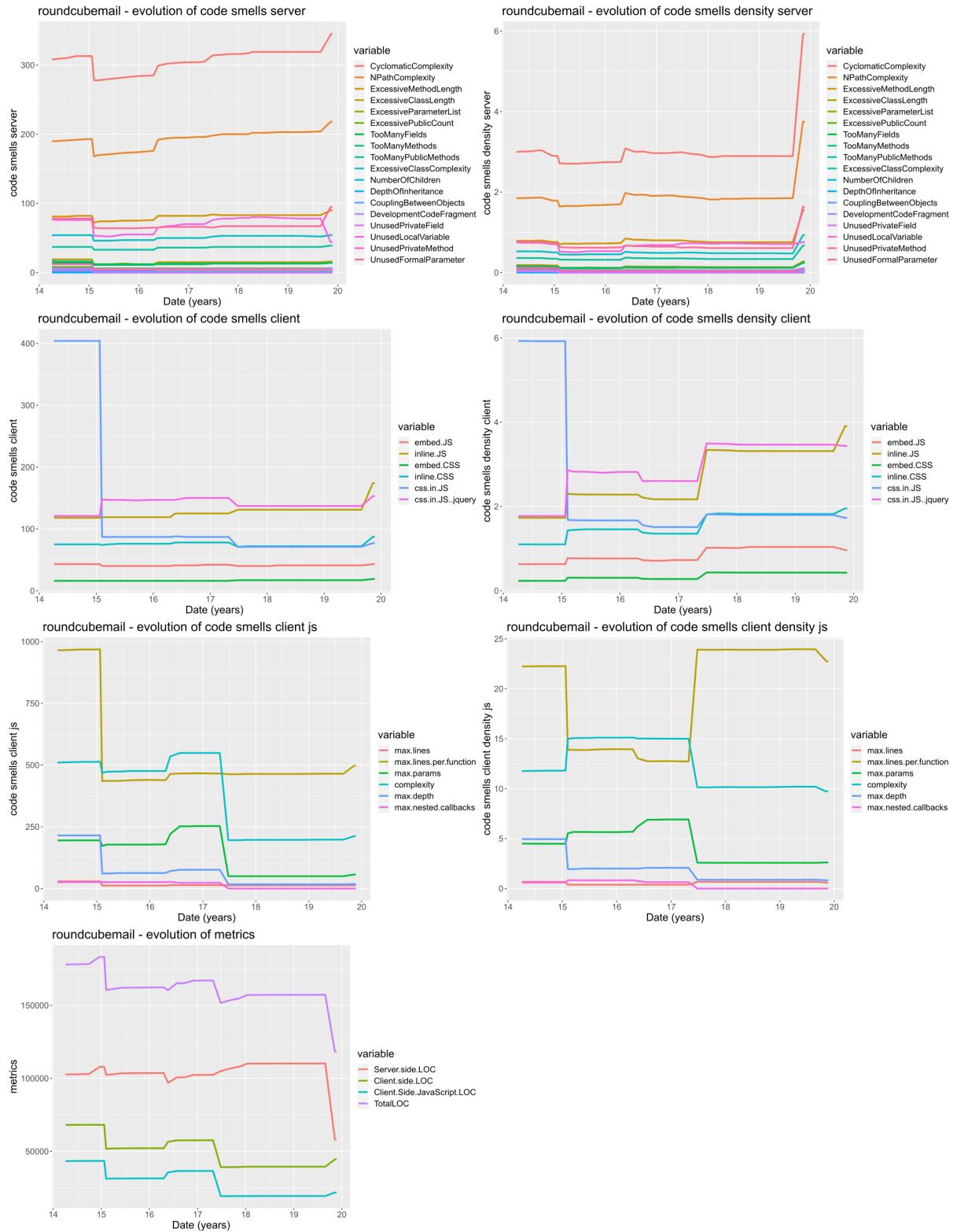


Fig. 10: CS and metrics evolution for Roundcube

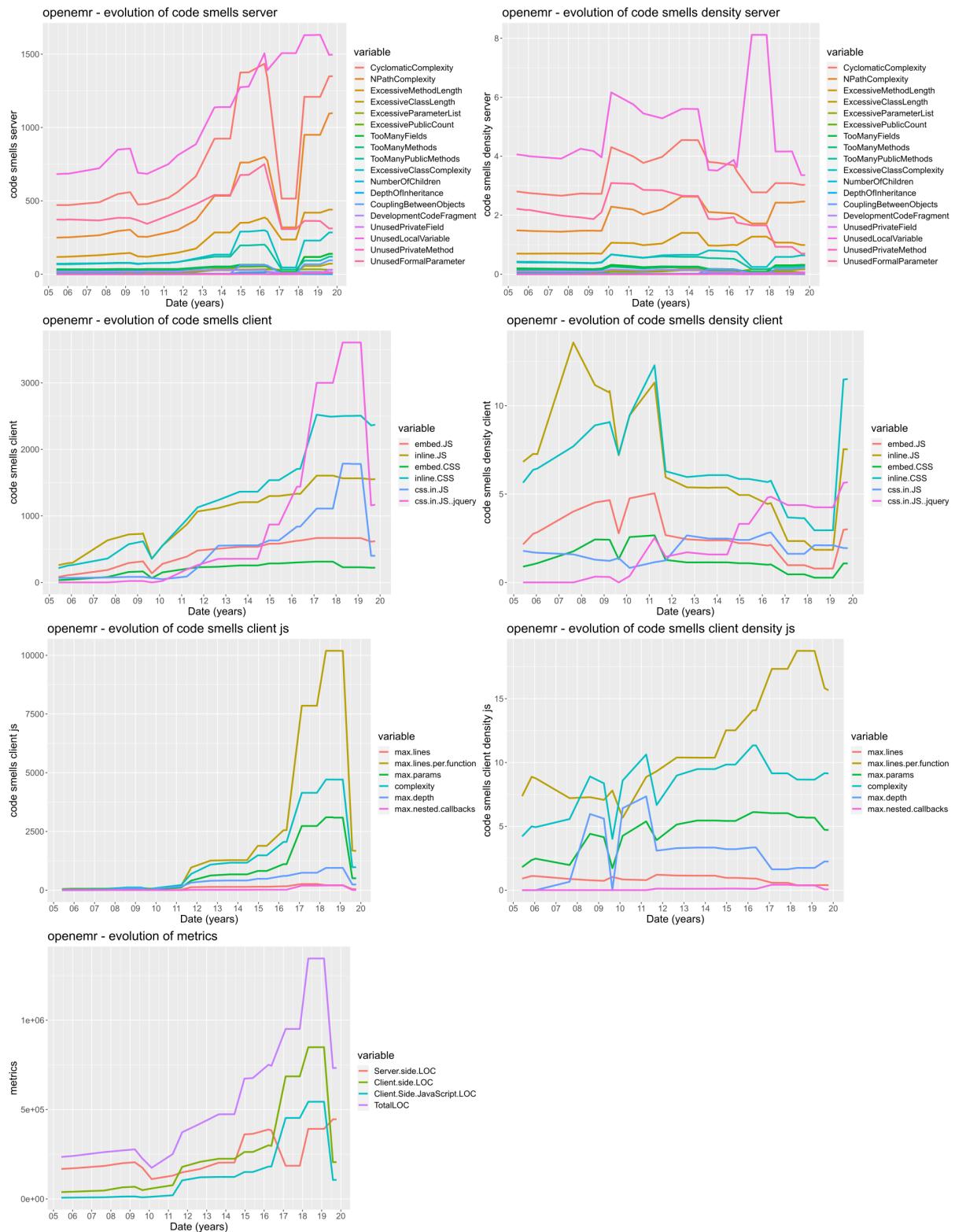


Fig. 11: CS and metrics evolution for OpenEMR

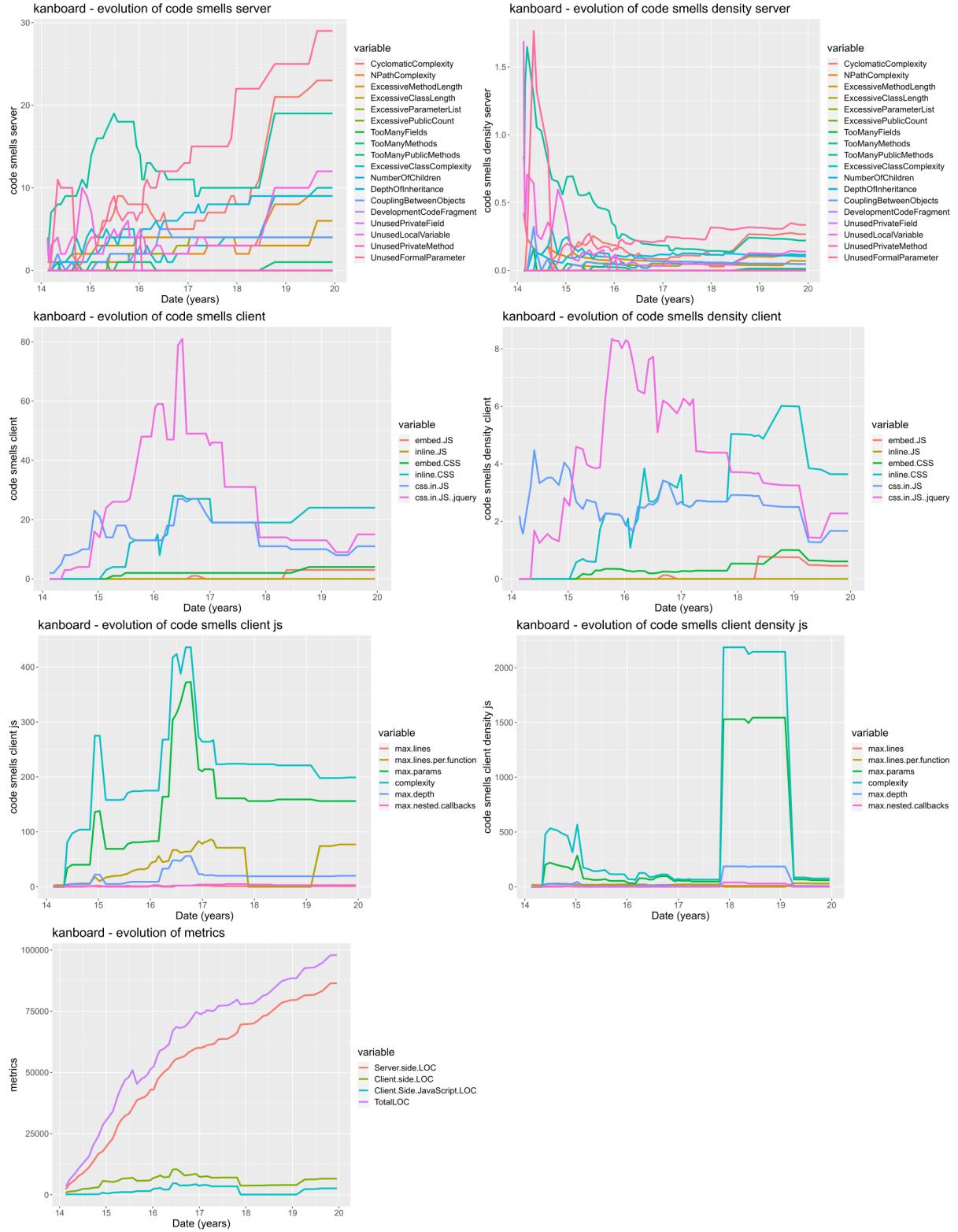
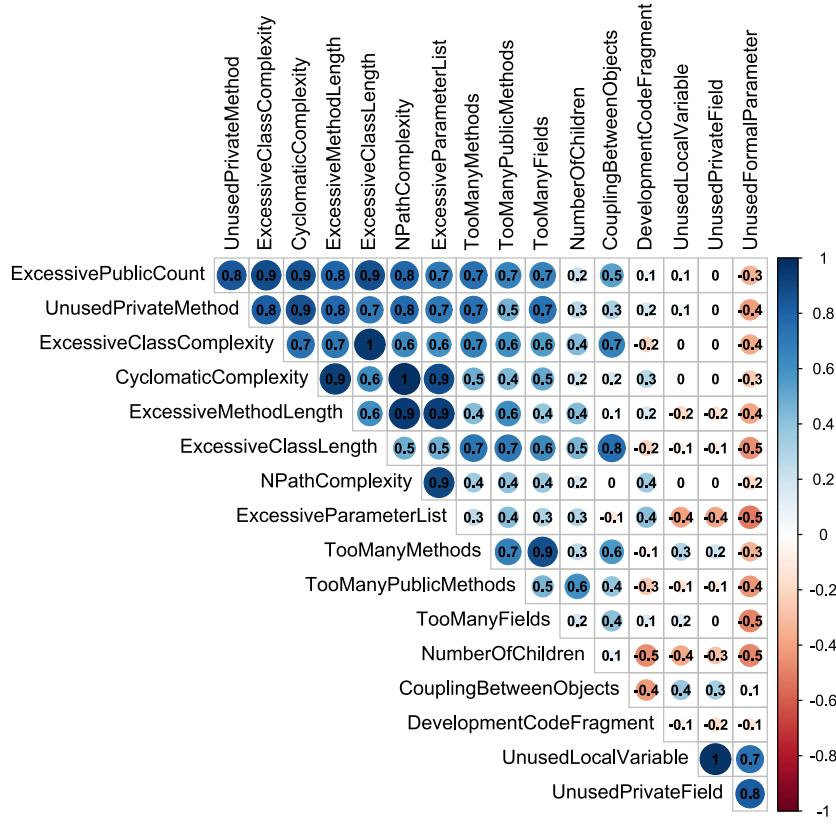
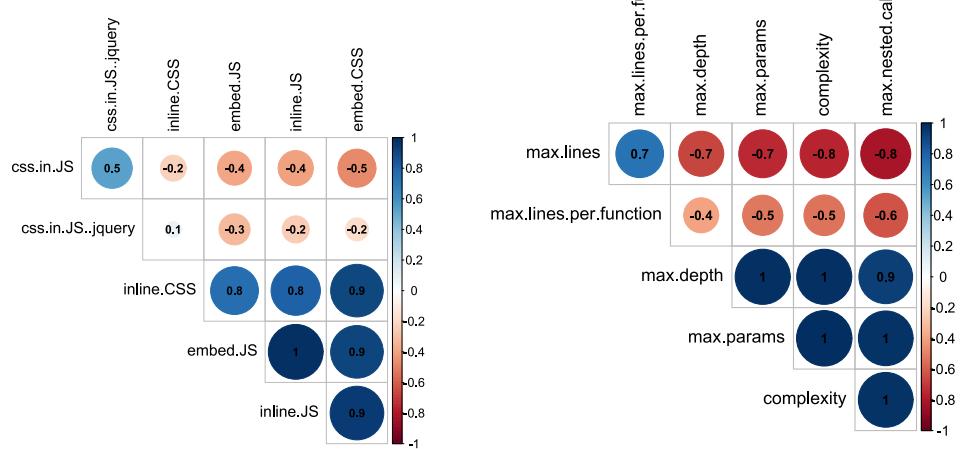


Fig. 12: CS and metrics evolution for Kanboard

1.2 Correlation matrices for CS in the same group



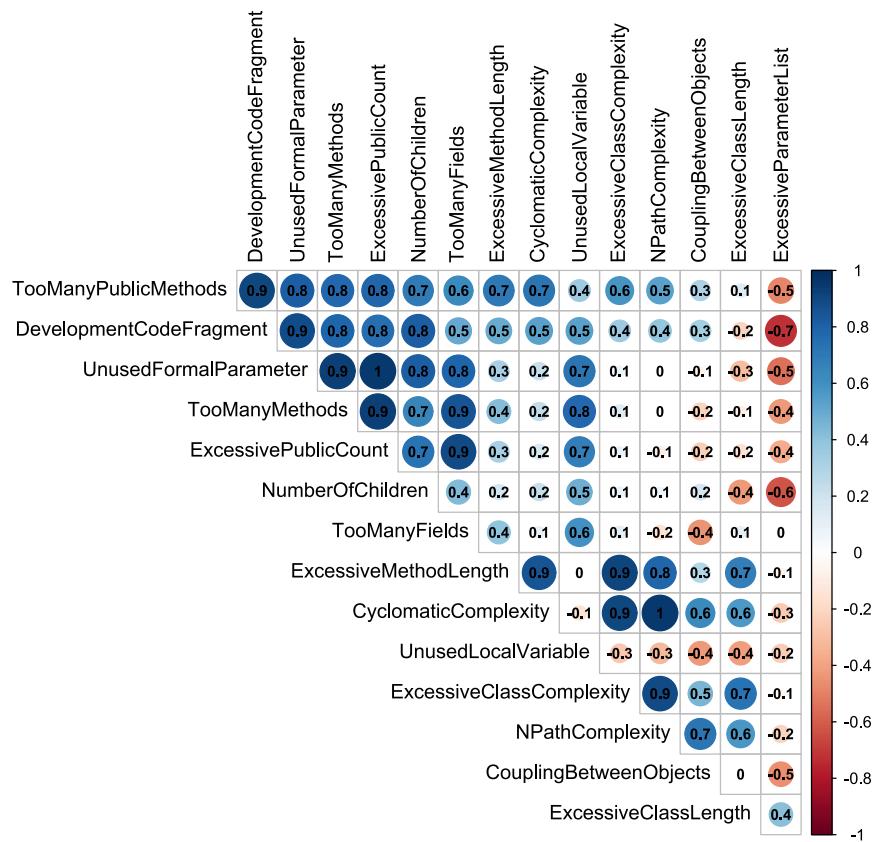
(a) CS server-side x CS server-side



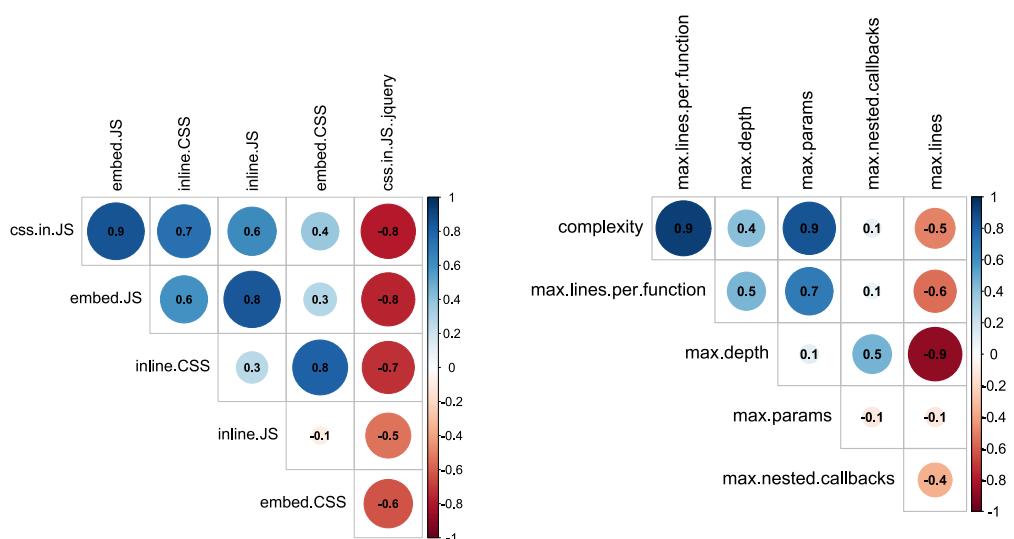
(b) CS client-side x CS client-side

(c) CS client-side js x CS client-side js

Fig. 13: phpMyAdmin - Correlation matrices for CS in the same group



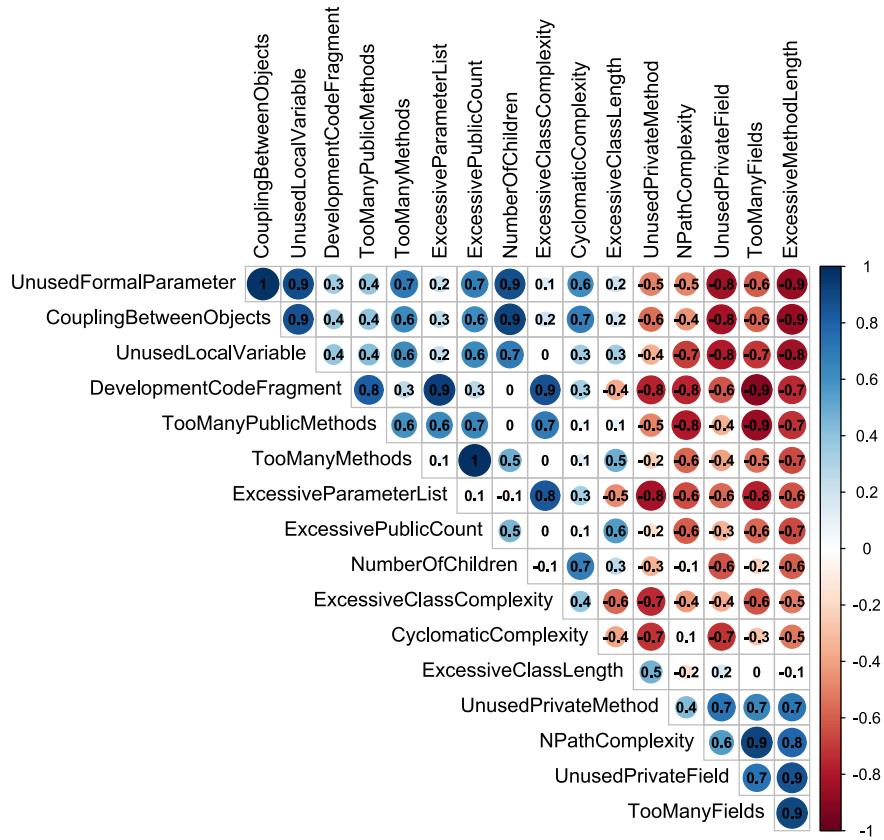
(a) CS server-side x CS server-side



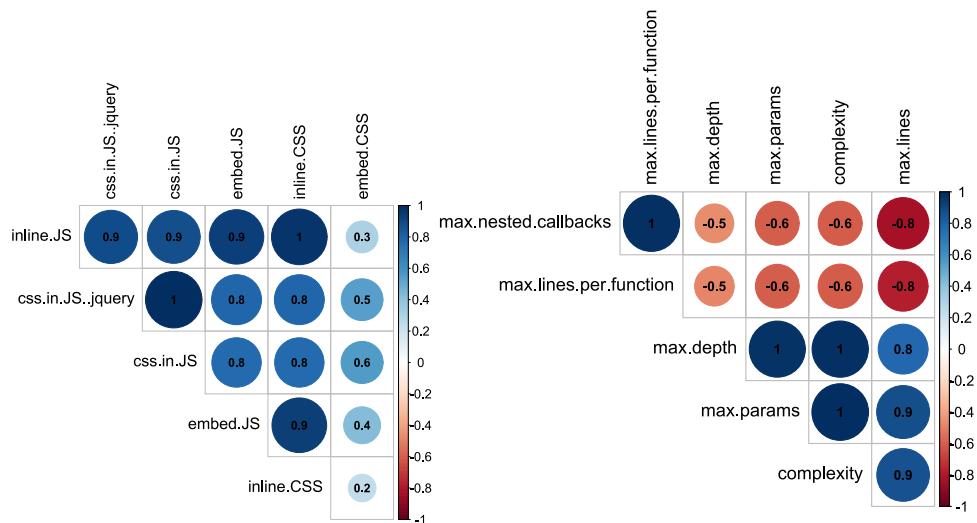
(b) CS client-side x CS client-side

(c) CS client-side js x CS client-side js

Fig. 14: DokuWiki - Correlation matrices for CS in the same group



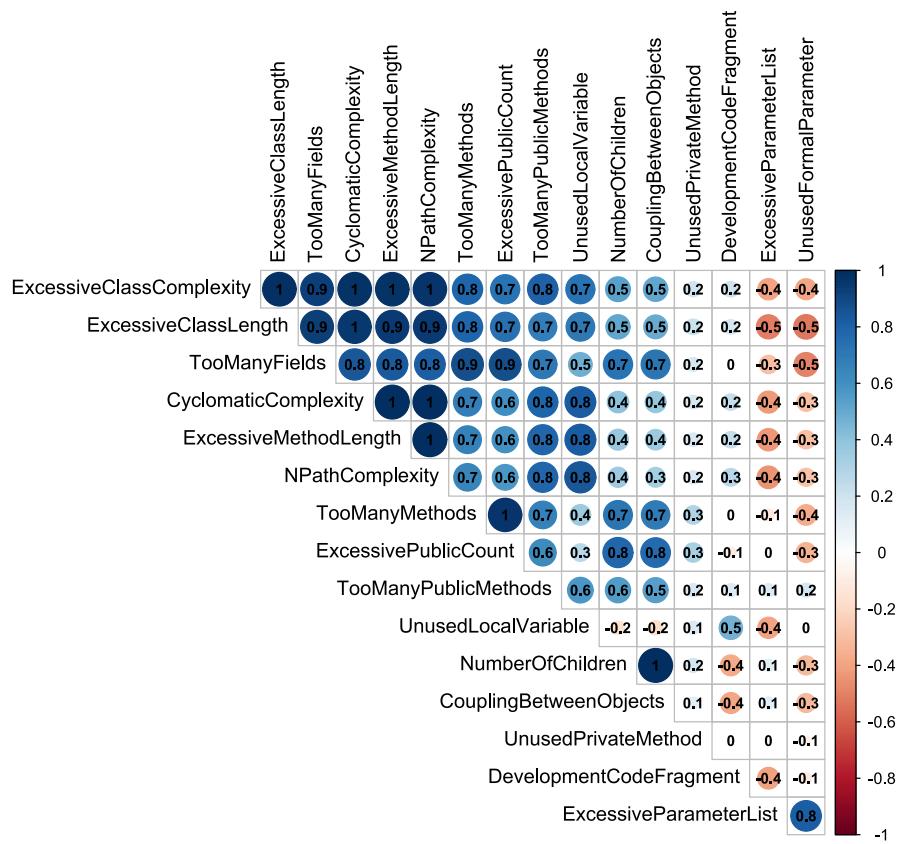
(a) CS server-side x CS server-side



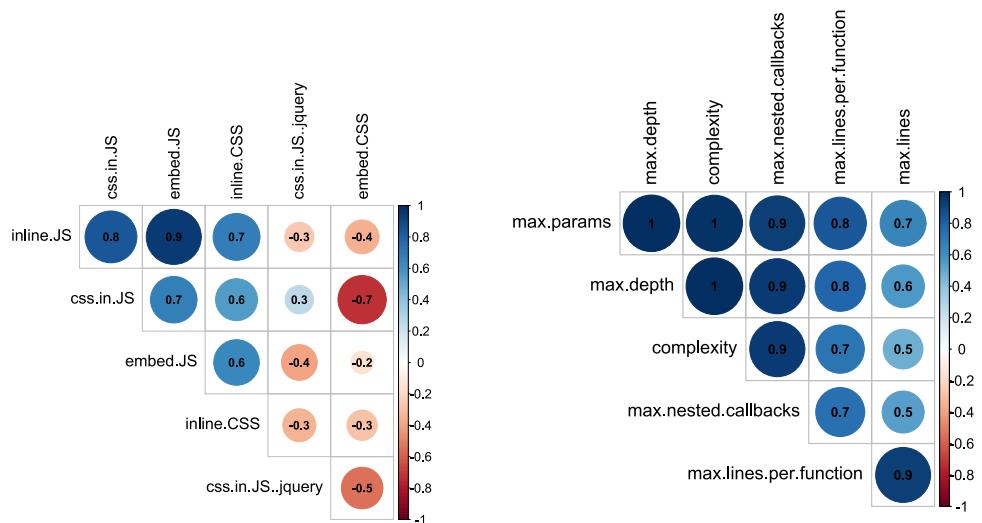
(b) CS client-side x CS client-side

(c) CS client-side js x CS client-side js

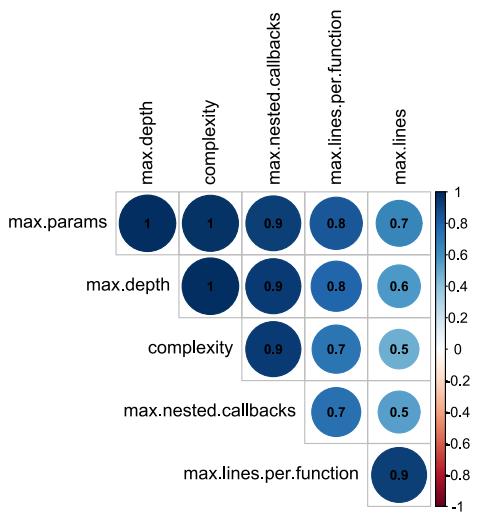
Fig. 15: OpenCart - Correlation matrices for CS in the same group



(a) CS server-side x CS server-side

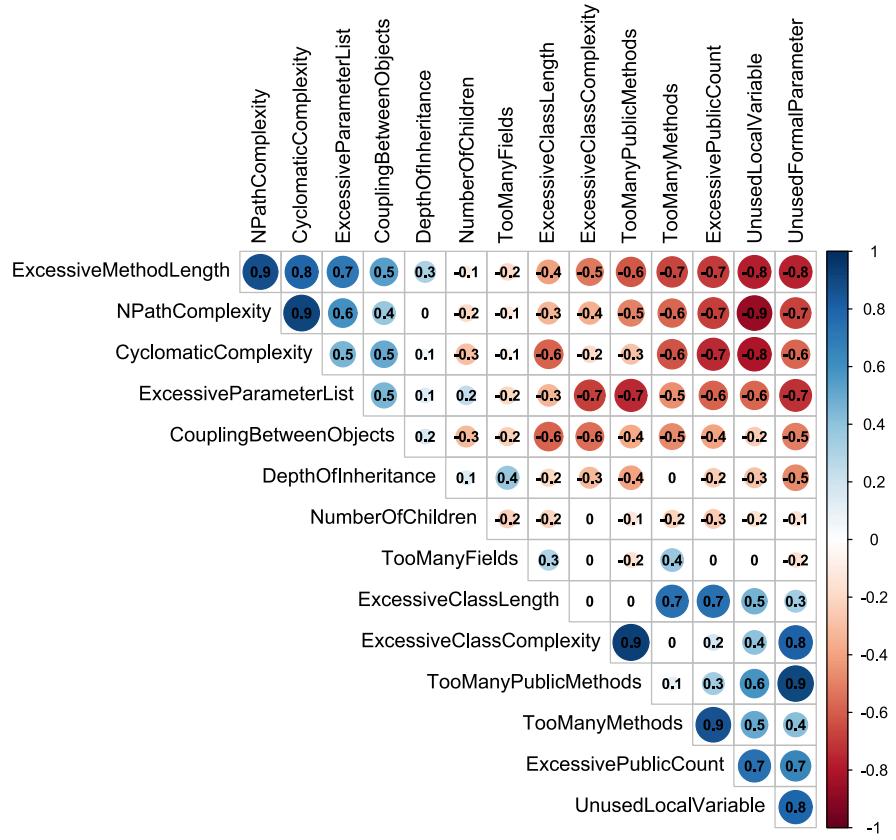


(b) CS client-side x CS client-side

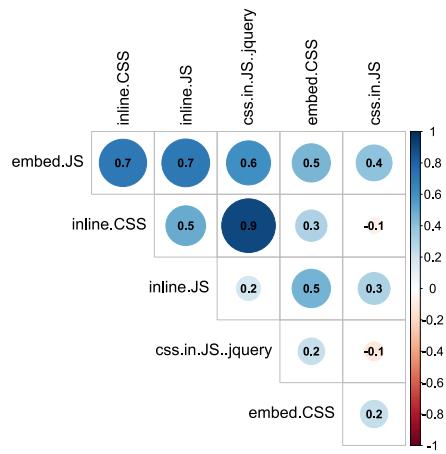


(c) CS client-side js x CS client-side js

Fig. 16: phpBB CS - Correlation matrices for CS in the same group



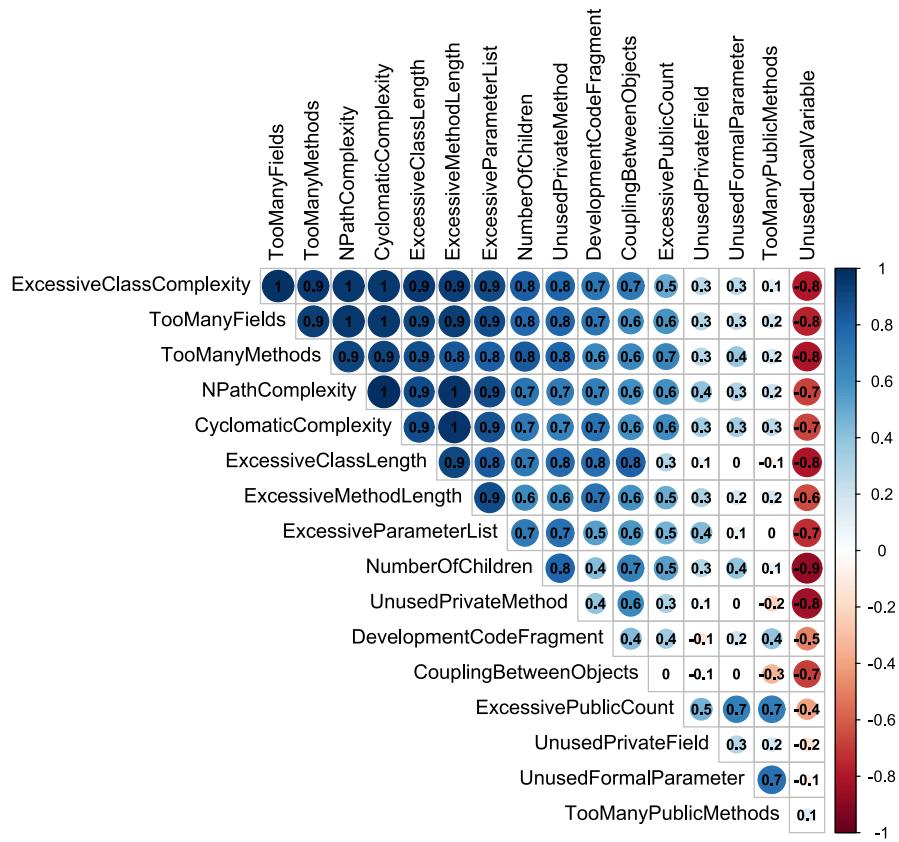
(a) CS server-side x CS server-side



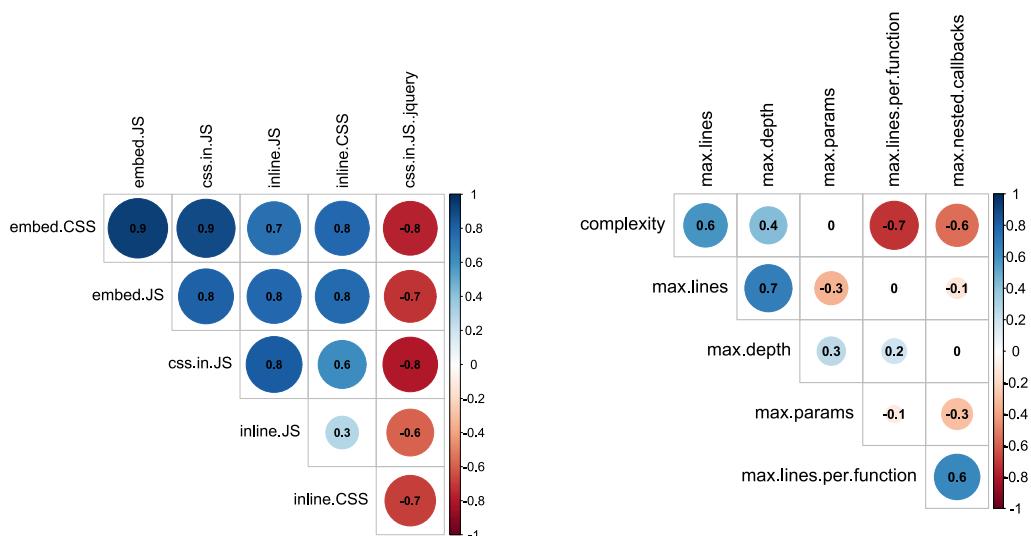
(b) CS client-side x CS client-side

(c) CS client-side js x CS client-side js

Fig. 17: phpPgAdmin - Correlation matrices for CS in the same group



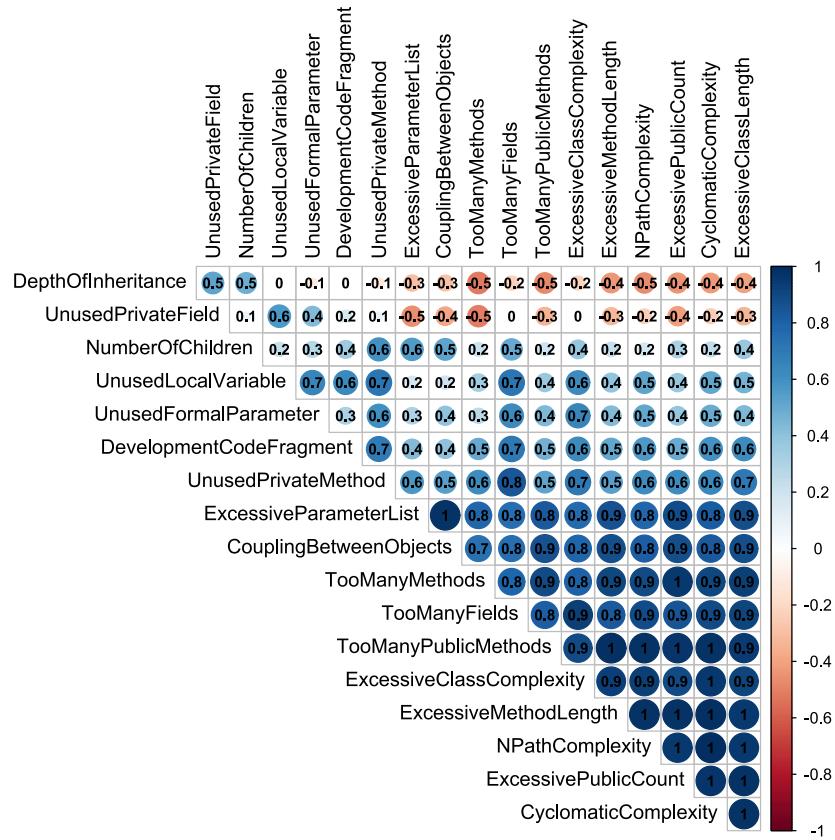
(a) CS server-side x CS server-side



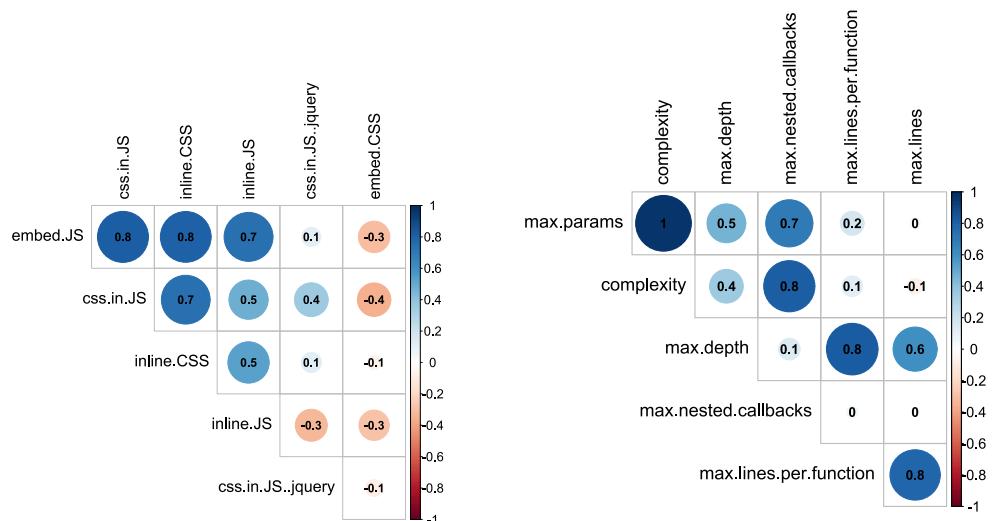
(b) CS client-side x CS client-side

(c) CS client-side js x CS client-side js

Fig. 18: MediaWiki - Correlation matrices for CS in the same group



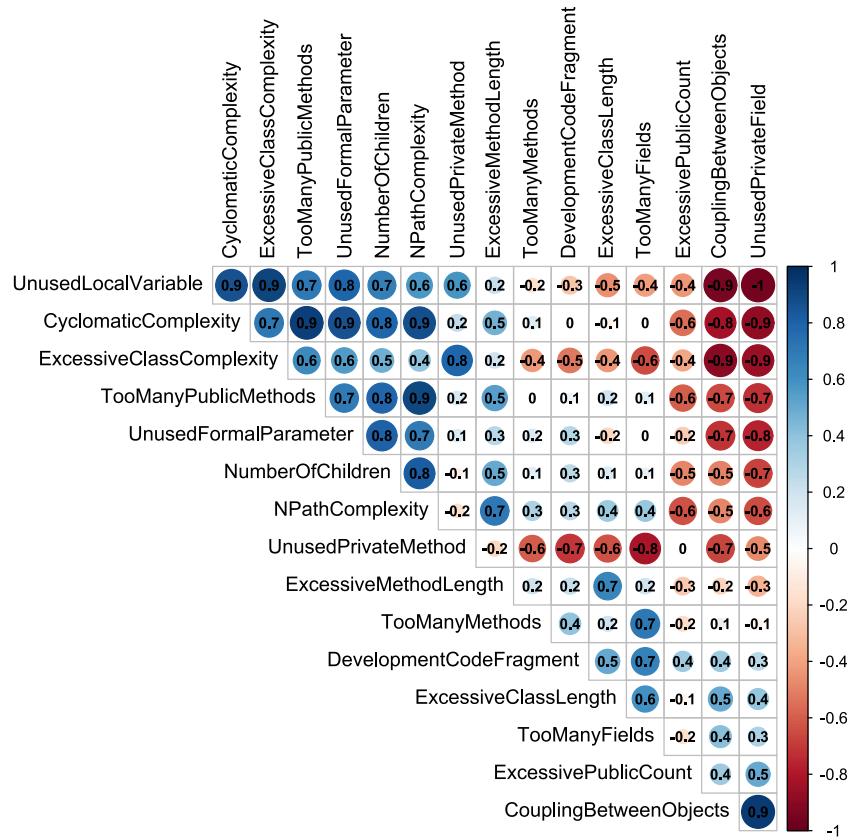
(a) CS server-side x CS server-side



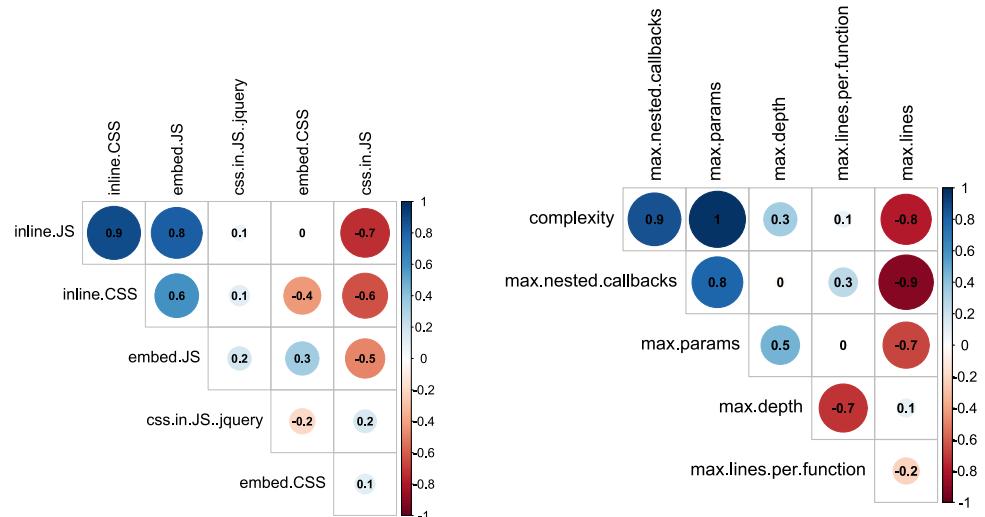
(b) CS client-side x CS client-side

(c) CS client-side js x CS client-side js

Fig. 19: PrestaShop - Correlation matrices for CS in the same group



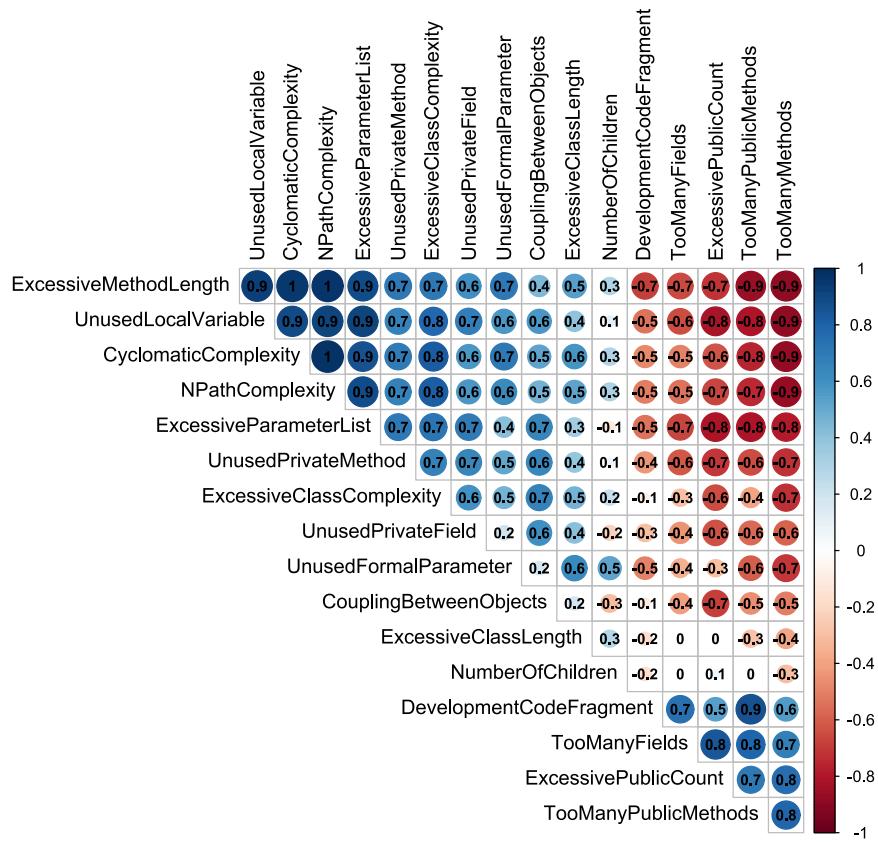
(a) CS server-side x CS server-side



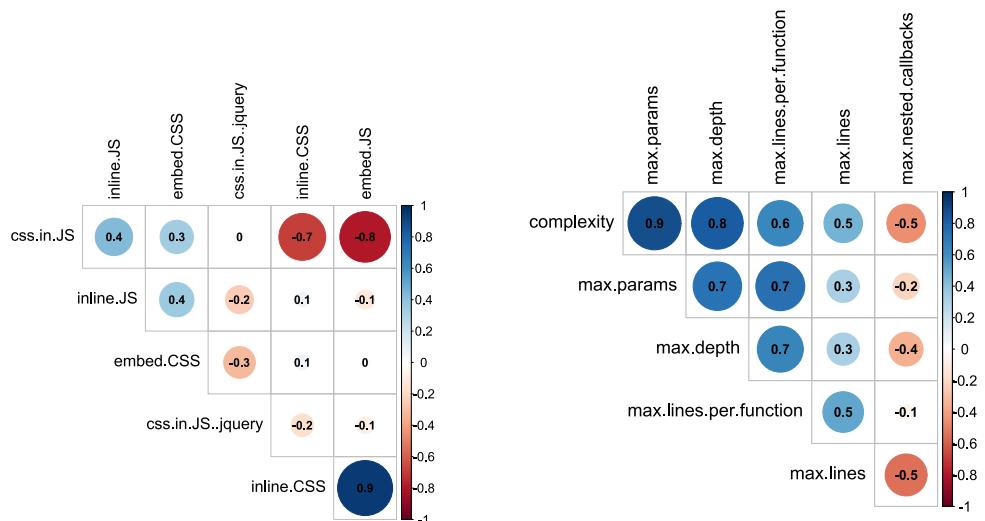
(b) CS client-side x CS client-side

(c) CS client-side js x CS client-side js

Fig. 20: Vanilla - Correlation matrices for CS in the same group



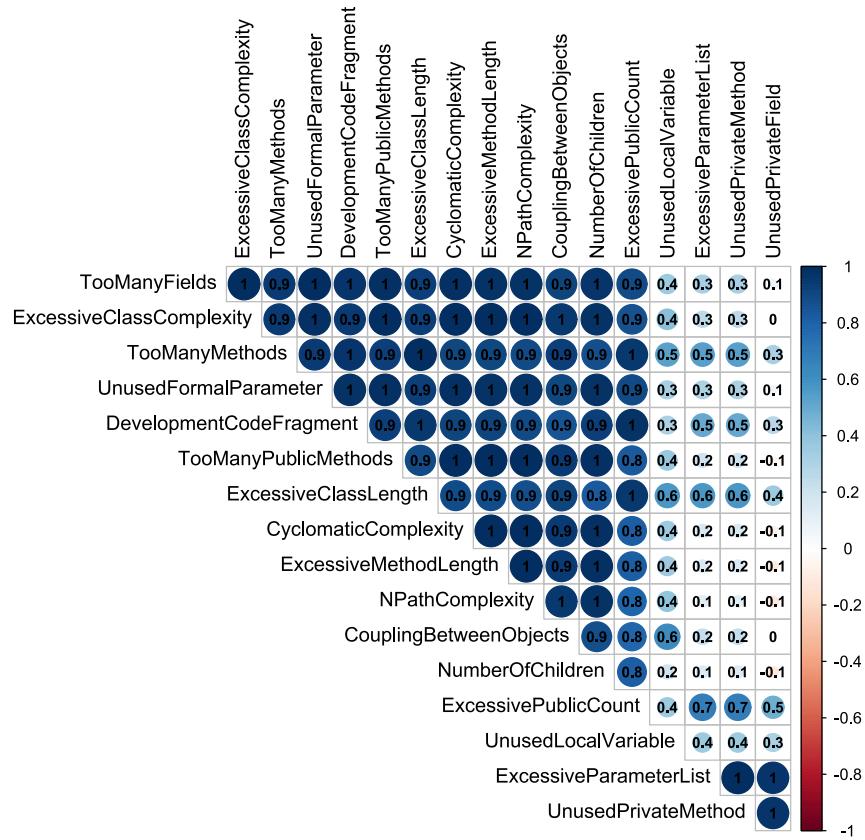
(a) CS server-side x CS server-side



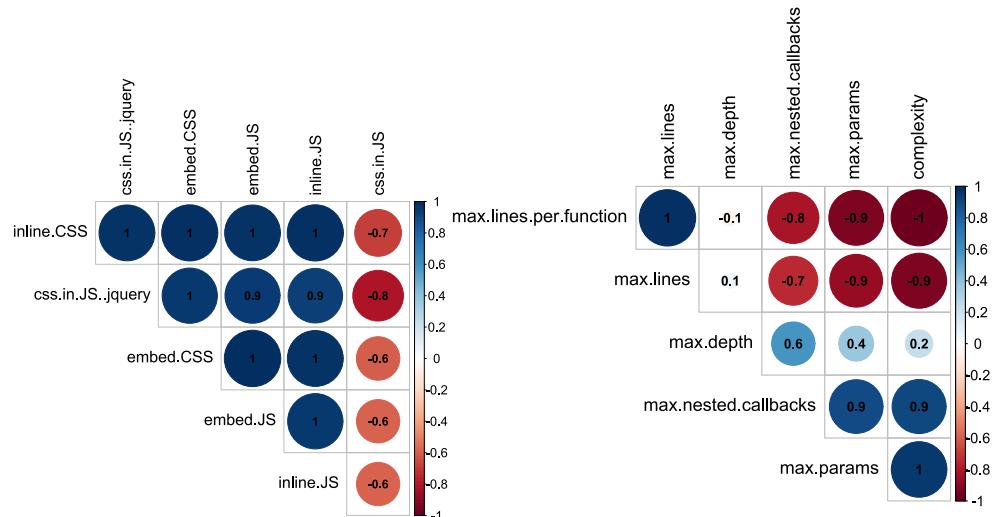
(b) CS client-side x CS client-side

(c) CS client-side js x CS client-side js

Fig. 21: Dolibarr - Correlation matrices for CS in the same group



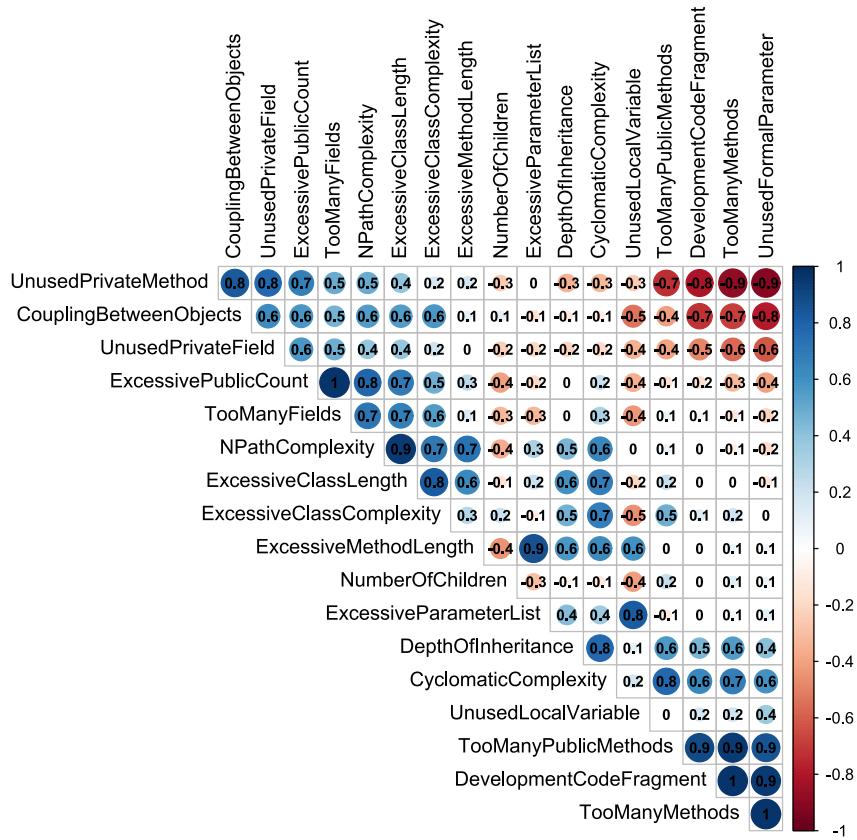
(a) CS server-side x CS server-side



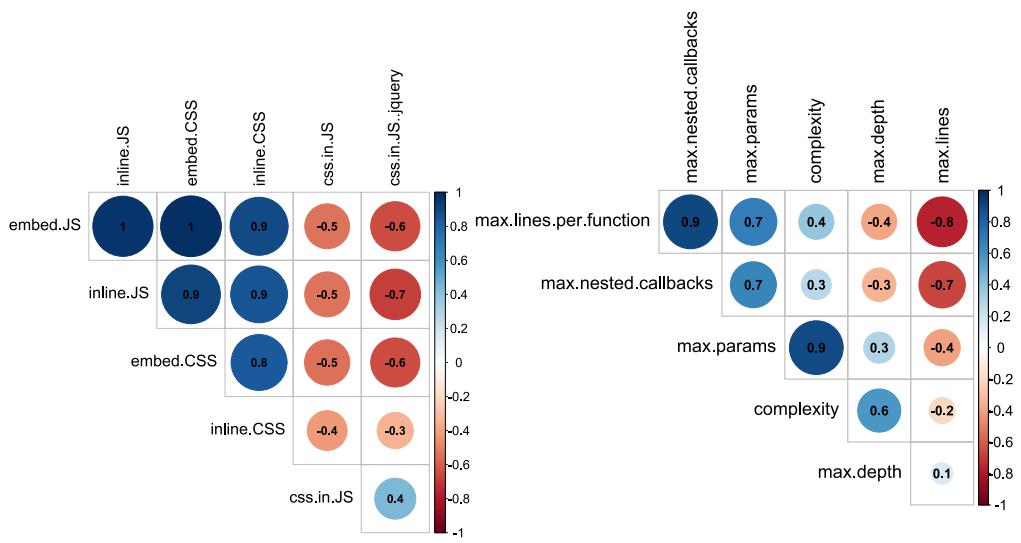
(b) CS client-side x CS client-side

(c) CS client-side js x CS client-side js

Fig. 22: Roundcube - Correlation matrices for CS in the same group



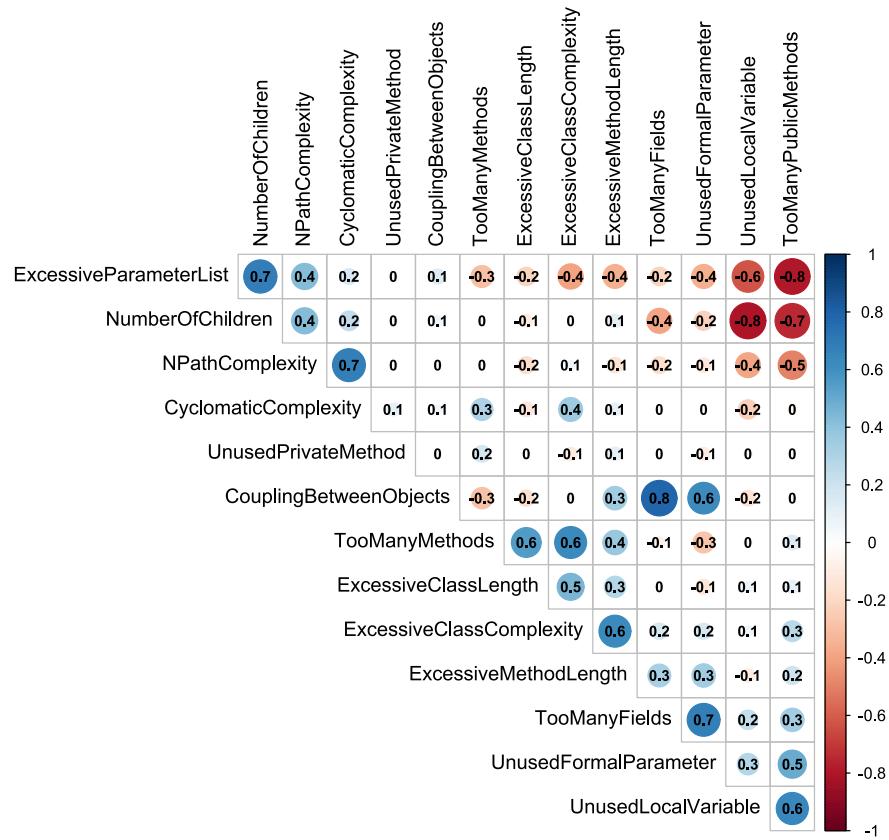
(a) CS server-side x CS server-side



(b) CS client-side x CS client-side

(c) CS client-side js x CS client-side js

Fig. 23: OpenEMR - Correlation matrices for CS in the same group



(a) CS server-side x CS server-side

Fig. 24: Kanboard - Correlation matrices for CS in the same group

2 RQ2 - Relationship between server- and client-side CS evolution

2.1 Correlation - Tables with more detail in data

Table 1: Correlation (cor_ts) between CS groups in apps - detail

Correlation	phpMyAdmin	DokuWiki	OpenCart	phpBB	phpPgAdmin	MediaWiki	PrestaShop	Vanilla	Dolibarr	Roundcube	OpenEMR	Kanboard
server,client cor	-0.05	0.53	-0.78	0.46	-0.54	0.47	-0.34	0.4	0.57	0.37	-0.03	-0.76
server,client p-value	0.533	0	0	0.001	0.002	0	0.003	0.001	0	0.042	0.887	0
server,client_js cor	0.14	-0.17	-0.79	0.52	0.28	-0.25	0.03	-0.53	-0.42	-0.13	0.2	-0.16
server,client_js p-value	0.071	0.285	0	0	0.142	0.004	0.824	0	0	0.475	0.265	0.201
client,client_js cor	0.35	-0.81	0.95	-0.26	-0.08	-0.53	0.27	-0.76	-0.68	0.26	-0.35	0.34
client,client_js p-value	0	0	0	0.068	0.692	0	0.021	0	0	0.16	0.044	0.006

2.2 Linear regression between Code Smell groups

Table 1 shows the detail of correlation cor_ts, with all applications represented. The line cor represents the value of the correlation, and the following line the p-value. The correlation shows only relations on the same release of the app.

Table 2: Linear regression (X=>Y) between Code Smell groups. A dot represents statistical significance.

Linear regression	phpMyAdmin	DokuWiki	OpenCart	phpBB	phpPgAdmin	MediaWiki	PrestaShop	Vanilla	Dolibarr	Roundcube	OpenEMR	Kanboard
server =>client	•	•	•	•	•	•	•	•	•	•	•	•
server =>client_js	○	•	•	•	•	•	•	•	•	•	•	•
client =>server	•	•	•	•	•	•	•	•	•	•	•	•
client =>client_js	•	•	•	○	•	•	•	•	•	•	•	•
client =>server	○	•	•	•	•	•	•	•	•	•	•	•
client_js =>client	•	•	•	○	•	•	•	•	•	•	•	•

The table 2 shows the linear regressions from groups A->B, with all apps represented. We used the "lm" R function to capture regression models with statistical significance, represented with the dot in the table. For the applications *phpBB*, *PrestaShop*, *Vanilla* and *roundcube*, the regression models have statistical significance in the regression among all CS groups. This significance can indicate a strong relation among all the groups of CS, in these apps, in the same release. If, for example, the server-side CS increase, the other two follow the same pattern in that same release.

In the next table 3 the first column represents the sum of the apps with statistical significance in the linear regressions among groups (a resume from table 2). The relations that have significance in most apps are: Server =>Client and Client =>Server; Client =>Client_js and Client_js =>Client.

2.3 Linear regression, Granger causality, Transfer Entropy up to lag 4

Table 3: Statistical causality from between CS groups (12 apps) using Linear regression, Granger-causality (lag 1-4) and Transfer Entropy (lag 1-4).

rel	LM	GC1	GC2	GC3	GC4	TE1	TE2	TE3	TE4
Server =>Client	●●●●●●●●●●	○	○○	●	●	●●○○○	○○	●	●
Server =>client _js	●●●●○	●		○	●	●●	●	●	●
Client =>Server	●●●●●●●●●●	●●●	●●○	●●●	●●	●●●●●○○	●●●●	●●	●●●
Client =>Client _js	●●●●●●●●○	●●○○	●●○	●	●○	●●●●○	●●●●	●●○	●●●
Client =>Server	●●●●○	○○	●	●	●	●●●●○	●●	●●	●●
Client _js =>Client	●●●●●●●●○	●●○○○	●●○	●●	●●○	●●●●○○	●●	●●	●●

Black dots and white dots represent statistical significance of 0.05 and 0.10, respectively, in one application

3 RQ3 - Impact of server- and client-side CS intensity evolution on web app' reported issues

3.1 Time-series correlation (cor_ts) between CS and issues per app

Table 4: Resume of time-series correlation (cor_ts) between CS and issues, per app

cs type	phpMyAdmin	DokuWiki	OpenCart	phpBB	PrestaShop	Vanilla	Dolibarr	Roundcube	OpenEMR	Kanboard
server	8	7	1	0	12	15	17	10	12	0
client	4	2	0	1	1	5	4	4	5	2
client _js	6	4	0	0	4	6	1	0	6	1
total	18	13	1	1	17	26	22	14	23	3

Number indicates the number of positive correlations greater than 0.3 between CS and issues.

Table 4 resumes the correlations by application. Seven of the ten applications have many timeseries correlations between CS and issues. However, three of the applications have almost no correlation at all (*OpenCart* and *phpBB* with just one, and *Kanboard* with only three). These applications behave differently than the others regarding the issues.

3.2 Causality relationships between CS and issues - up to lag 4

Reproduction of the extended tables in the chapter 6, but with more lags, both for Granger-causality and Transfer Entropy.

3.2.1 Causality relationships between CS and issues density

Table 5: Statistical causality from CS density to Issues density relations (10 apps) using Linear regression, Granger-causality (lag 1-4) and Transfer Entropy (lag 1-4).

CS	LM	GC1	GC2	GC3	GC4	TE1	TE2	TE3	TE4
CyclomaticComplexity	●●●○○	●	●●●	●●●	●●	●●●○○	●●●○	●	●●
NPathComplexity	●●○	●○○	●	●●	○	●●●○	●●●	●●	●●○
ExcessiveMethodLength	●●●○	●●●	●●●	●●○	●●	●●●●	●●○	●●○	●●
ExcessiveClassLength	●●●	●●○	●○	●○	●●	●●●●○	●●		
ExcessiveParameterList	●○	●●●○	●●●○	●●●	●●○	●●●○	●●○		
ExcessivePublicCount	●				○	●●○	●●○		
TooManyFields	●●○	●●○	●●○	●○	○	●●○	●●○		
TooManyMethods	●○	●	●●		○	●●●○	●●○	●	●
TooManyPublicMethods	●●●○○	●●●	●○	●○	●	●●●	●●●	●●	●●
ExcessiveClassComplexity	●●●○	●●	●●●○	●○	●○	●●●○	●●●●○○	●●	●●
NumberOfChildren	●●●	●●●○	●●	●	●○	●●●○	●●●		
DepthOfInheritance							○		
CouplingBetweenObjects	●●	●●●●	●●	●●○	●●●	●●●○	●●		
DevelopmentCodeFragment	●●●○	●●○	●	●	●○	●●●●●○	●●	●	●
UnusedPrivateField	●●	●●	●			●●●		●	●
UnusedLocalVariable	●●	●●	●●	●●●	●●●	●●●●	○	●	●
UnusedPrivateMethod	●○○				○	●●●	●○○	●	●
UnusedFormalParameter	●●●○○	●●○	●●●●	●●	●●	●●●○	●●●●	●●●●	●●●●
embed.JS	●●●○		●	●○	●	●●●	●○		
inline.JS	●●		●○	●○	●○	●●●●	●●●	●●	●●
embed.CSS	●●●●	●●●	●●●	●●●	●●●	●●●●	●●●	●●	●●
inline.CSS	●●●●●	●●●○	●●●○	●	●○	●●●●●	●●●●○○	●●○	●●
css.in.JS	●○○	●	●●		○	●●○	●	○	
css.in.JS..jquery	●●●○○	●○○	●●	●●○○	●○○	●●●○○	●●○	●	
max.lines	●●●●	●●	●●	●	●	●●●○	●	●○	●
max.lines.per.function	●●●●	●●●	●●●○	○		●●○	●●	●○	●
max.params	●●●●●	●●●○	●○	●○	○	●●●●●●●○○	●●●●●	●●●	●●
complexity	●●●○	●●●	●	●●○	●○	●●●●●●●○○	●●●●●	●○	●●
max.depth	●●●●		●	●●●	●●●	●●●●●○○	●●●●●	●●●●	●●●●
max.nested.callbacks	●●●●	●	●	●●○	●●	●●●○○	●●●	●	●

Columns: Linear regression(LM), Granger-causality (lag1 to 4), and Transfer Entropy (lag 1 to 4). The first 18 CS are from the server-side/PHP, the following 6 are the client-side embed CS, and the last 6 are the client-side JavaScript CS. Black dots and white dots represent statistical significance of 0.05 and 0.10, respectively, in one application.

3.2.2 Causality relationships between CS and issues absolute number

Table 6: Statistical causality from CS density to Issues relations (10 apps) using Linear regression, Granger-causality (lag 1-4) and Transfer Entropy (lag 1-4).

CS	LM	GC1	GC2	GC3	GC4	TE1	TE2	TE3	TE4
(Ex.)CyclomaticComplexity	●●○○	○○	●●○	●	●	●●○○	●●	●	●●
(Ex.)NPathComplexity	●●	●●	●○	○	○	●●●	●●	●	●●
ExcessiveMethodLength	●●●●●	●●●●●	●●●○	●●○○	●○	●●●●	●●	●●	●●
ExcessiveClassLength	●●●	●●●	●●○	○	●	●○	●	○	
ExcessiveParameterList	●●	○○	●○	●	●●	●●○	●●○		
ExcessivePublicCount	●●	●○				●●●●○○	●●		
TooManyFields	●●	●○	●○	●		●●●	●○		
TooManyMethods	●●●○	●●	●		○	●●●●○○	●●●	●	●
TooManyPublicMethods	●●●○	●●	●●	●○	○	●●●	●●	●	●
ExcessiveClassComplexity	●●●●○○	●●○○	●●	●●	●●	●●○	●●○○	●●	●●
(Ex.)NumberOfChildren	●●○	●●●	●●○	●●	●●	●●○○	●●		
(Ex.)DepthOfInheritance									
(Ex.)CouplingBetweenObjects	●●○	●●	●	●	○	●●			●●
DevelopmentCodeFragment	●●○	○○	●●	●●	●●	●●●●●	●●	●●	●●
UnusedPrivateField	●●●	●●○	●○			●●●●	●●	●●	●●
UnusedLocalVariable	●●●●○○	●●○	●●	●●○○	●●	●●●●○○		●●	●●
UnusedPrivateMethod	●●●○	●●				●●●	●●○	●●	●●
UnusedFormalParameter	●●●○	●●	●●●	●●	●●	●●○○	●●○	●●●	●●●
embed.JS	●●●○	●○	●●○	●●	○	●●●		○	
inline.JS	●●	●●	●●	●○○	○○	●●●●○○	●●●	●●	●●
embed.CSS	●●○	●●	●●●○○	●●○○	●○	●●●●	●●	●●	●●
inline.CSS	●●	●●	●●	●●○	●○	●●●●●○○	●●●	●●	●●
css.in.JS	●●●○○	○○	○○			●●●	●●○	○○	
css.in.JS.jquery	●●●○○	●●	●●○	○○	●○	●●○○		●●	
max.lines	●●●●○○	●○	●●○	●●	●●	●●●●○○	●●	●●	●●
max.lines.per.function	●●●●○○	●○	●●○	○○	○	●●●●	●●	●●	●●
max.params	●○	●○	○○○			●●●●●○○	●●●●	●●	●●
(Ex.)complexity	●●○	●●○	●●○○			●●●●○○	●●●●○○	●●	●●
max.depth	●○	●●	●●○○		●●	●●●●●○○	●●●●○○	●●●●	●●●●
max.nested.callbacks	●●○	●●	●●	●●●●	●●●●	●●●●	●●●●	●●●●	●●●●

Columns: Linear regression(LM), Granger-causality (lag1 to 4), and Transfer Entropy (lag 1 to 4). The first 18 CS are from the server-side/PHP, the following 6 are the client-side embed CS, and the last 6 are the client-side JavaScript CS. Black dots and white dots represent statistical significance of 0.05 and 0.10, respectively, in one application.

4 RQ4 - Impact of CS intensity evolution on a web app's bugs

4.1 Time-series correlation (cor_ts) between CS and bugs per app

Table 7: Resume of time-series correlation (cor_ts) between CS and bugs, per app

	phpMyAdmin	DokuWiki	OpenCart	phpBB	PrestaShop	Vanilla	Dolibarr	Roundcube	OpenEMR	Kanboard
server	9	9	1	0	7	16	14	11	2	1
client	4	2	0	1	0	4	3	4	4	2
client_js	6	3	0	0	4	6	0	0	5	4
total	19	14	1	1	11	26	17	15	11	7

Number indicates the number of positive correlations greater than 0.3 between CS and bugs

Table 7 shows the correlations divided by app and by type of code smell (server, client, and client_js). The table shows that both the server and client CS are related to the bugs, except for two applications. These correlations can be not linear because cor_ts can measure non-linear correlations between 2 irregular time series with different granularity's/number of observations (check figure ??). This correlation only measures similarity between time series. For causality, we have to do some other studies that we show in the next section.

4.2 Causality relationships between CS and bugs - up to lag 4

4.2.1 Causality relationships between CS and bugs absolute number

Table 8: Statistical causality from CS density to Bugs relations (10 apps) using Linear regression, Granger-causality (lag 1-4) and Transfer Entropy (lag 1-4).

CS	LM	GC1	GC2	GC3	GC4	TE1	TE2	TE3	TE4
(Ex.)CyclomaticComplexity	●●●	●●○	●○○	●○	●	●●●○○○	●○	●	●●
(Ex.)NPathComplexity	●●●	●●●	●●○	○○		●●●○○○	●●	●	●●
ExcessiveMethodLength	●●●●●	●●●●	●●○	●●○○	●●	●●●●○	●○	●●	●●
ExcessiveClassLength	●●●	●●●	●●	●●○	●●	●●●●●	○		
ExcessiveParameterList	●●●	●○	●●	●○		●●●●○	●●●●	●●	●
ExcessivePublicCount	●●○	●●	○	●○	○	●●●○	●		
TooManyFields	●●●○	●○	●●	●		●●●	○		
TooManyMethods	●●●○○	●●○○				●●●●○	●●	●	●
TooManyPublicMethods	●●●●	●●○○	●●	●	○	●●○	●○	●	●
ExcessiveClassComplexity	●●●●●	●●●○○	●●○	○	○	●●●●●○	●●○○○	●●	●●
(Ex.)NumberOfChildren	●●●●	●●●	●●●	●●	●	●●●●●	●○		
(Ex.)DepthOfInheritance						●	●	●	●
(Ex.)CouplingBetweenObjects	●●○	●○○	●	●○	○	●●●●●●●	●●	●	●●
DevelopmentCodeFragment	○○	●●	●	●●	●○	●●●●●	●	●	●
UnusedPrivateField	●●●	●○	●			●●●●○	●●●○	●●	●●
UnusedLocalVariable	●●●●	●●○	●	●●○○	●○	●●○	○	●	●
UnusedPrivateMethod	●●●●○	●○				●●●●○	●	●	●
UnusedFormalParameter	●●●○	●	●○○	●●	●●●	●●○○	●●●	●●●	●●●
embed.JS	●●●○	●●●○	●●	●●	●●	●●●●●	○		
inline.JS	●●○	●●○	●●○	●●	●●	●●●●○	●●●	●●	●●
embed.CSS	●●○	●●	●●○	●●●	●●○	●●●●○	●●	●●	●
inline.CSS	●●●●	●●●	●●○	●●	●○	●●●●●●●	●●●○	●●●	●●●
css.in.JS	●●●○○	●●○	○			●●●●○	○	○	
css.in.JS.jquery	●●●	●●	●●	●●○	●●	●●○	●○	●	
max.lines	●●●	●●●	●●	●●	●●	●●●●○○○	●●	●	●
max.lines.per.function	●●●	●●●	●●	●○	●○	●●●	●●●	●●	●●
max.params	●○○	●○	●●	●●○	●○	●●●●●●●	●●●●	●●	●●
(Ex.)complexity	●●○	●●	●●○	○○	○○	●●●●○	●●○	●	●●
max.depth	●		●○	●●	●●	●●●●○○○	●●●●○○○	●●●●●	●●●●●
max.nested.callbacks	●●	●●○	●●	●●○○	●●●	●●●●○○○	●●●●	●●	●●

Columns: Linear regression(LM), Granger-causality (lag1 to 4), and Transfer Entropy (lag 1 to 4). The first 18 CS are from the server-side/PHP, the following 6 are the client-side embed CS, and the last 6 are the client-side JavaScript CS. Black dots and white dots represent statistical significance of 0.05 and 0.10, respectively, in one application.

4.2.2 Causality relationships between CS and bugs density

Table 9: Statistical causality from CS density to Bugs density relations (10 apps) using Linear regression, Granger-causality (lag 1-4) and Transfer Entropy (lag 1-4).

CS	LM n	GC1	GC2	GC3	GC4	TE1	TE2	TE3	TE4
(Ex.)CyclomaticComplexity	••	••	••	••••○	••	••••○	••••○	•○	••
(Ex.)NPathComplexity	••••	••○	•	○○		•••○	•	○○	••
ExcessiveMethodLength	•••••○	•••	•••	•••••○	•••	•••○	••○	•••	••
ExcessiveClassLength	••○	••	••	••••○○	○○	•••••○	○		
ExcessiveParameterList	•••○	••○	○○	••		••••○	•••	•	
ExcessivePublicCount	○○				○○	•○○	○		
TooManyFields	••	••	○○	••		••○	•		
TooManyMethods	•		•			••••○	••	•	
TooManyPublicMethods	••••	••••	○○	•	•	•••○	•○	•	•
ExcessiveClassComplexity	••••	•••○	••○	••○	○○	•••••○○	••○○	••○	••
(Ex.)NumberOfChildren	•••	••○	○○	•	○	••○	••		
(Ex.)DepthOfInheritance									
(Ex.)CouplingBetweenObjects	•••	••	•	•○○	••	•••	•○		••
DevelopmentCodeFragment	••○	••○	○○	•	••	•••••	••	•	•
UnusedPrivateField	•••	••○	•			••••○	••	•	•
UnusedLocalVariable	•••○		•	••••	•••○	••••○			
UnusedPrivateMethod	•	○			•	•••	••	•	•
UnusedFormalParameter	••••	••○○	○○	•	•○○	•○○	••	••○○	••
embed.JS	••○○	•	○○	•○	•	••••○			
inline.JS	○○	•	••	••	•	•••○	•••	••	••
embed.CSS	•••○	•••○	••	•••	••	•••	•	○○	•
inline.CSS	••••	••○	•••	•	•	•••••○	••○○	•••	••
css.in.JS	○○	••	••			••••○		○	
css.in.JS..jquery	••	••	••	•••	••	•••••○○	•••	○○	
max.lines	••○	••	•	•	•	••••○○	•○	•	•
max.lines.per.function	••○	••	••	••	•	••••	••○	•	•
max.params	••	••○	••○	••○○	••○	•••••••○○	••○	••	••
(Ex.)complexity	••	••	••○	••○○	•	•••○○	••••	•	••
max.depth	•			•○	•	•••••○○	••••	••••	••••
max.nested.callbacks	•	••	•	•••	••	••••○○	•••	•○	•

Columns: Linear regression(LM), Granger-causality (lag1 to 4), and Transfer Entropy (lag 1 to 4). The first 18 CS are from the server-side/PHP, the following 6 are the client-side embed CS, and the last 6 are the client-side JavaScript CS. Black dots and white dots represent statistical significance of 0.05 and 0.10, respectively, in one application.

5 RQ5 - Causality relationships between CS and time to release

Next table shows values with lags up to 4.

Table 10: Statistical causality from CS density to Time To Release relations (12 apps) using Linear regression, Granger-causality (lag 1-4) and Transfer Entropy (lag 1-4).

CS	LM	CG1	CG2	CG3	CG4	TE1	TE2	TE3	TE4
(Ex.)CyclomaticComplexity	●●●●●	●●●○○	○○	●●○	●●○	●●●○○○	○○	○○	○○
(Ex.)NPathComplexity	●●●●●	●●●○○	○○	○○	●●○	●●●○○	●●○○	●	●○
ExcessiveMethodLength	●●●●●●	●●●○○○	●●●	●●●○○	●●●	●●●●○○○	●●●	●●○○	●●
ExcessiveClassLength	●●●●	●●●●○○○	●●●	●●○○	●●●	●●●●○○○	●●●	○	○
ExcessiveParameterList	●●●●	●●●○○	●●●	●●●	●●●	●●●●○○○	●●●	○	○
ExcessivePublicCount	●●●○○	●●○○	●●●	●●●	●○	●●●○○	●	○	○
TooManyFields	●○○	●○	●●●	●●●	●●●	●●●●○○○	○		
TooManyMethods	●●●●○○	●●●●○○○	●●●○○	●●●○○	●●●●●	●●●●●○○○	●●●○○	●	●
TooManyPublicMethods	●●○○	●●○○	●●●	●●●	●○	●●●●●	●●○○	●●●	●●
ExcessiveClassComplexity	●●●●●●	●●●○○○	●●●●	●●●●	●●●●	●●●●●○○○○	●●●●	●●●	●●●
(Ex.)NumberOfChildren	●●○○	●●○○	●●○○	●●○○	●●○○	●●●●○○○	●	○	
(Ex.)DepthOfInheritance	●					○			
(Ex.)CouplingBetweenObjects	●●●●	●●●○○	●●●○	●●●○	●●●●	●●●●●	●●●	●	●●●
DevelopmentCodeFragment	●●●○○	●●●○○○	●●●	●●○○	●●●	●●○○	○	○	○
UnusedPrivateField	●●○○	●●○○	●●○○	●●○○	●●	●●●●●		●	●
UnusedLocalVariable	●●●○○	●●●○○○	●●●○○	●●●○○○	●●○	●●●●○○○○	○	●	●
UnusedPrivateMethod	●●●●	●●●●○○○	●●●●	●●●●	●●●●	●●●●●●●●	●●●●	●	●
UnusedFormalParameter	●●●●○○	●●●●○○○○	●●●●○○○	●●●●○○○○	●●●●●	●●●●●●●●	●●●●●	●●●●●	●●●●●
embed.JS	●●●●●●	●●●●●●○○○	●●●●○○○	●●●●○○○	●●●●●	●●●●●●○○○	●●●●○○○		
inline.JS	●●●●●●	●●●●●●○○○	●●●●○○○	●●●●○○○	●●●●●	●●●●●●○○○	●●●●○○○	●○○	●
embed.CSS	●●●●●●	●●●●●●○○○	●●●●○○○	●●●●○○○	●●●●●	●●●●●●○○○	●●●●○○○	●○○	●
inline.CSS	●●●●●●○○○	●●●●●●○○○○	●●●●○○○○	●●●●○○○○	●●●●●●	●●●●●●●●●●○○○○	●●●●●●○○○○	●●●●●●	●●●●●●
css.in.JS	●●●	●●●	●●●	●●●	●●●	●●●●●●●●	●●●●●●●●	●●○○	●
css.in.JS..jquery	●●○○	●●●●○○○○	●●○○	●●○○	●●○○	●●●●●●●●	●●●●●●●●	●○○	●
max.lines	●●●●○○○○	●●●●○○○○○	●●●●○○○○○	●●●●○○○○○	●●●●●●●●●●	●●●●●●●●●●○○○○	●●●●●●●●●●	●●○○	●●○○
max.lines.per.function	●●●●●●○○○○	●●●●●●○○○○○	●●●●●●○○○○○	●●●●●●○○○○○	●●●●●●●●●●	●●●●●●●●●●○○○○	●●●●●●●●●●	○	
max.params	○	●●○○	●●○○	●●○○	●○○	●●●●●●●●●●	●●●●●●●●●●	●	●○○
(Ex.)complexity	●●○○	●●○○	●●○○	●●○○	●●	●●●●●●●●●●	●●●●●●●●●●	●●○○	●●
max.depth	○○	○○	●●●○○○○○	●●●○○○○○○	●●●●●●●●●●	●●●●●●●●●●○○○○○	●●●●●●●●●●	●●●●●●●●●●	
max.nested.callbacks	●●●○○○	●●●○○○○○	●●●○○○○○○	●●●○○○○○○○	●●●●●●●●●●	●●●●●●●●●●○○○○○○	●●●●●●●●●●	●	●

Columns: Linear regression(LM), Granger-causality (lag1 to 4), and Transfer Entropy (lag 1 to 4). The first 18 CS are from the server-side/PHP, the following 6 are the client-side embed CS, and the last 6 are the client-side JavaScript CS. Black dots and white dots represent statistical significance of 0.05 and 0.10, respectively, in one application.