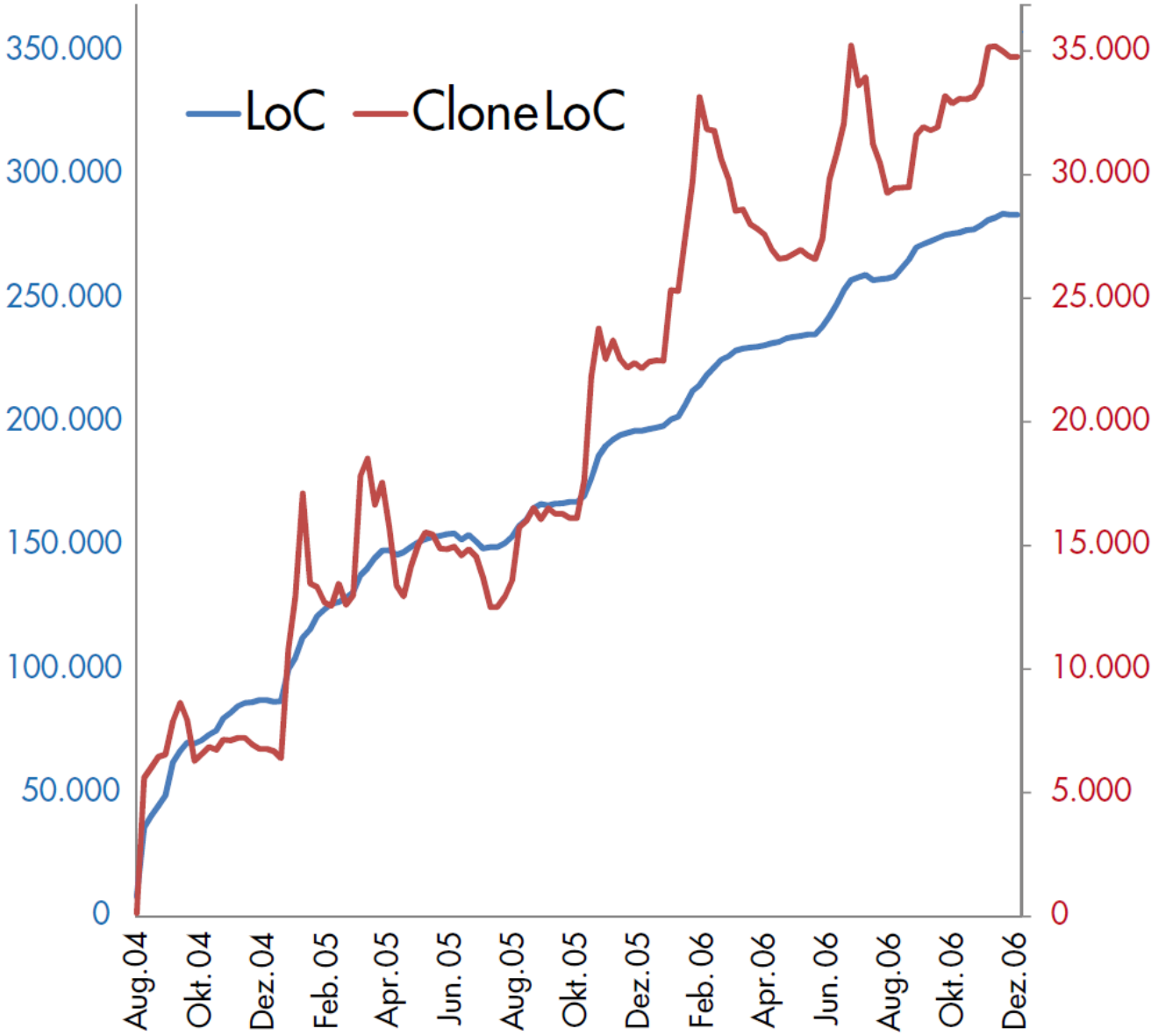
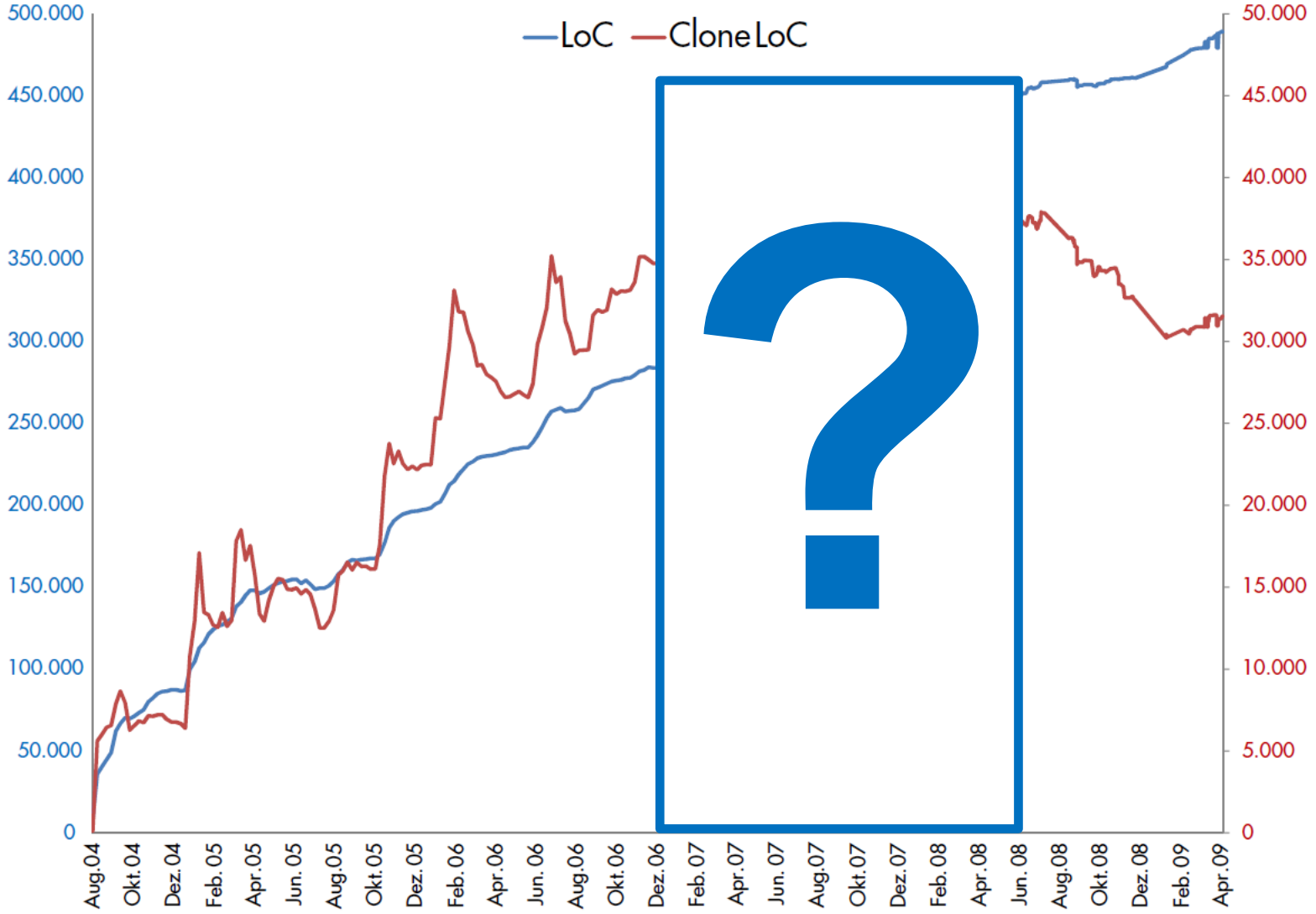
A Framework for Incremental Quality Analysis of Large Software Systems

Veronika Bauer
Lars Heinemann
Benjamin Hummel

Elmar Juergens

Michael Conradt


LoC
CloneLoC
350.000
300.000
250.000
200.000
150.000
100.000
50.000
0
35.000
30.000
25.000
20.000
15.000
10.000
5.000
0
Aug.04
Okt.04
Dez.04
Feb.05
Apr.05
Jun.05
Aug.05
Okt.05
Dez.05
Feb.06
Apr.06
Jun.06
Aug.06
Okt.06
Dez.06


LoC
Clone LoC
500.000
450.000
400.000
350.000
300.000
250.000
200.000
150.000
100.000
50.000
0
50.000
45.000
40.000
35.000
30.000
25.000
20.000
15.000
10.000
5.000
0
Aug.04
Okt.04
Dez.04
Feb.05
Apr.05
Jun.05
Aug.05
Okt.05
Dez.05
Feb.06
Apr.06
Jun.06
Aug.06
Okt.06
Dez.06
Feb.07
Apr.07
Jun.07
Aug.07
Okt.07
Dez.07
Feb.08
Apr.08
Jun.08
Aug.08
Okt.08
Dez.08
Feb.09
Apr.09
?

Continuous Quality Control

Continuous measurement and inspection of quality indicators…

…allows early removal (while removal is still cheap)

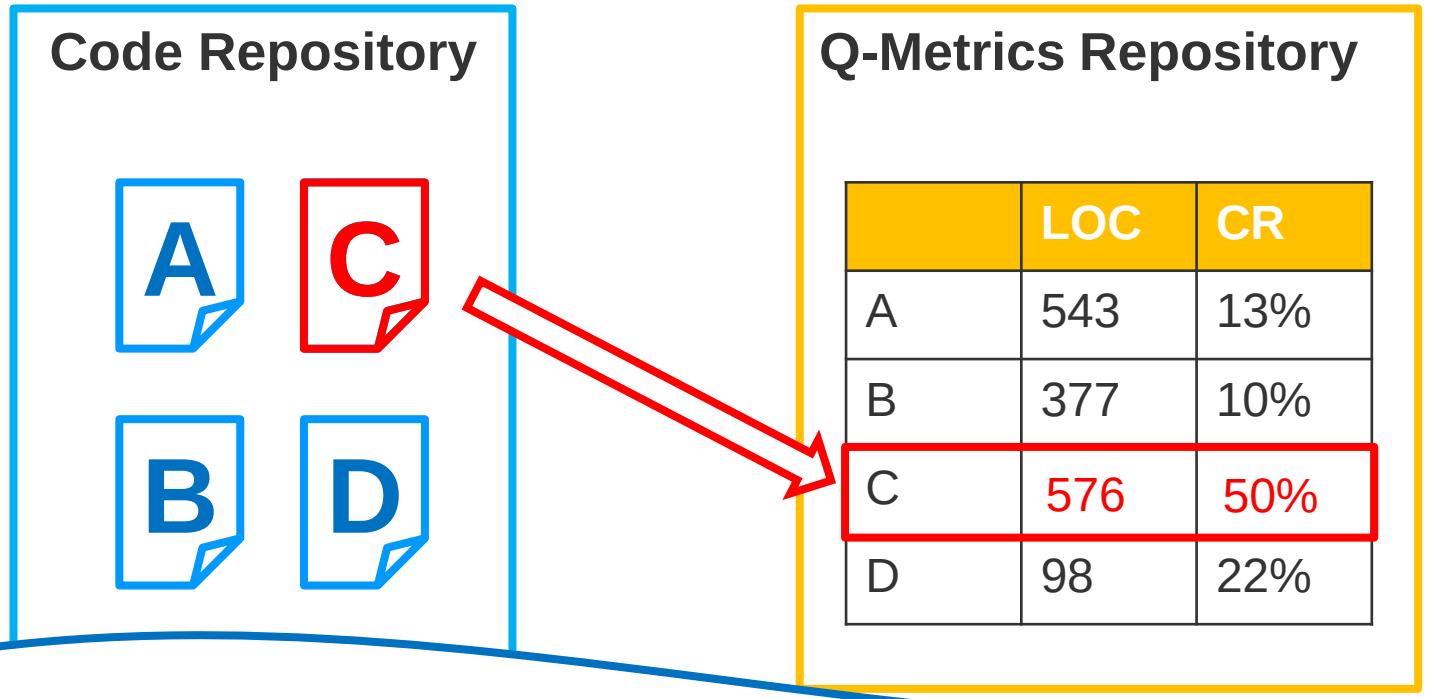
Requirements

Get immediate feedback upon commit, as an hour later the developer might already be working on a different task.

Provide historic data on all revisions, to allow root-cause analysis of quality problems.

Approach: Incremental Update

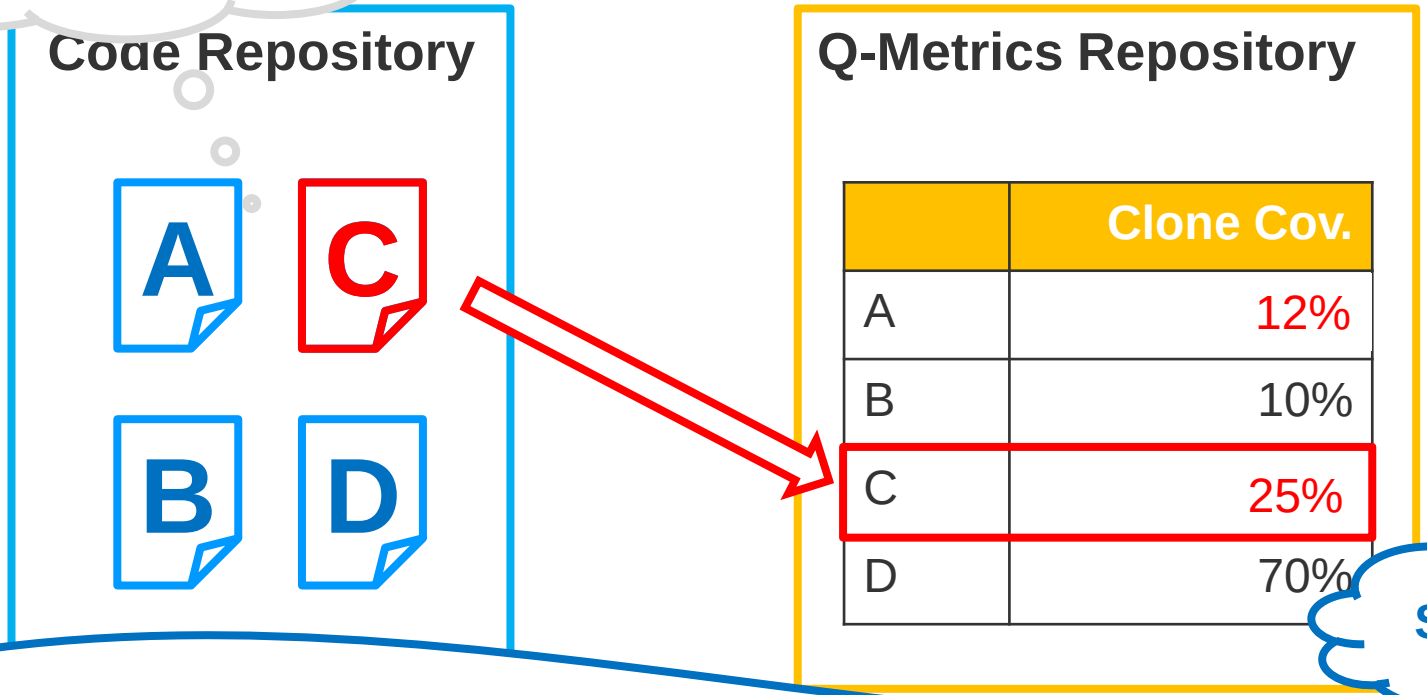
Per-File Metrics



Works for quite a large number of metrics!

Global Dependencies?

Developer copies from A to C



See paper!

How do we know to also update A?

Experiments

using ConQAT (http://www.conqat.org/)

Systems

	JabRef	ArgoUML	Chromium
Language	Java	Java	C/C++
kLOC	130	362	2,570
#files	626	1,866	13,005
rel. revisions	1,496	3,650	53,470

Calculate

LOC, SLOC, Nesting, Comment Ratio, Longest Method, Clone Coverage, Aggregated Values

for all files in all revisions

Results

	JabRef	ArgoUML	Chromium
Naive (est.)	11 hours	39 hours	134 days
Incremental	20 min.	66 min.	49 hours
Speedup	32	35	66
Per revision	0.8 sec	0.9 sec	3.3 sec

Conclusion

Summary

- Generic framework for incremental calculation of quality indicators
- Incremental update of metrics less than 4 second per revision on average
- Speedup of up to 66 compared to naïve metric computation