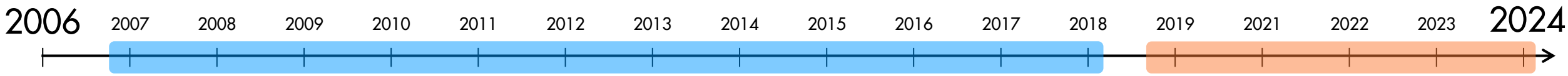
Are Code Reviews Worth It?
Insights from 16 Years of Review Data

We Are Developer 2024

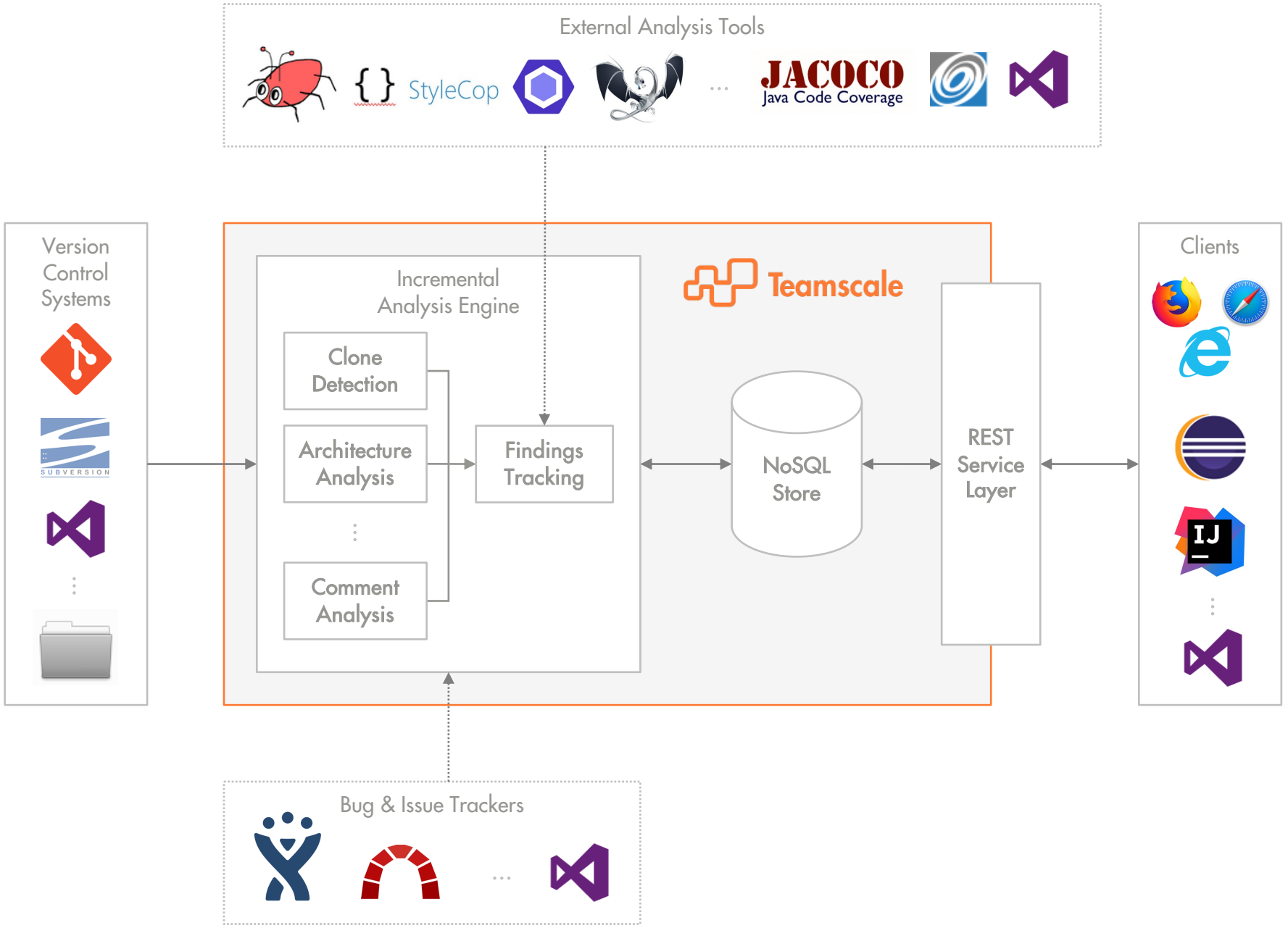
CQSE | Dr. Sven Amann

TECHNISCHE UNIVERSITÄT DARMSTADT

CQSE



Teamscale


External Analysis Tools
StyleCop
JACOCO
Java Code Coverage
Version Control Systems
SUBVERSION
Incremental Analysis Engine
Clone Detection
Architecture Analysis
Comment Analysis
Findings Tracking
Teamscale
NoSQL Store
REST Service Layer
Clients
IJ
Bug & Issue Trackers

Teamscale in Numbers

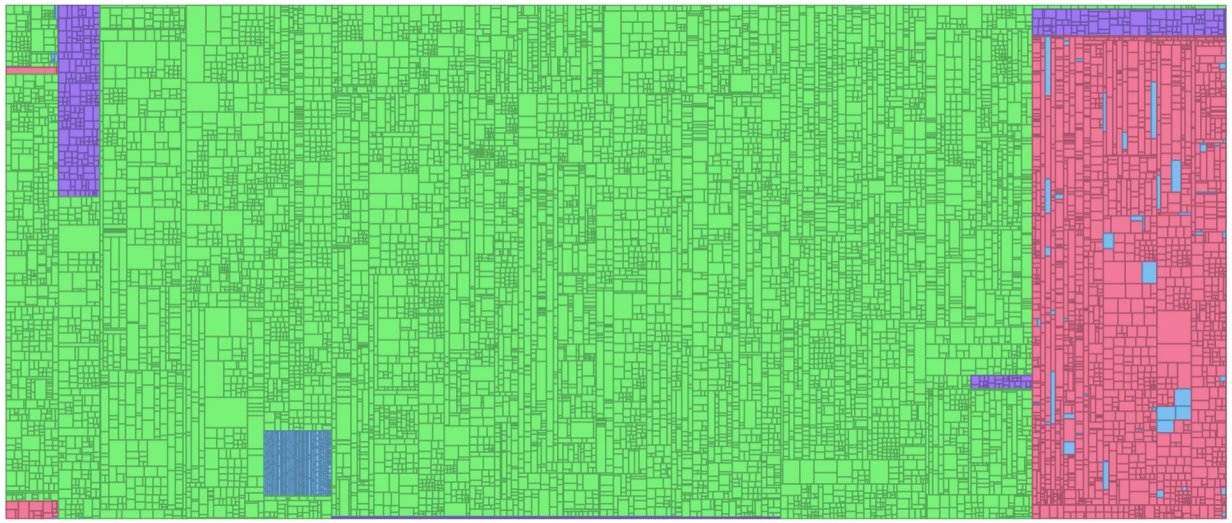
Lines of Code for teamscale

1.5M

Number of Files for teamscale

13.4k

Teamscale is Java (green), Typescript (pink) and C# (violet) for teamscale

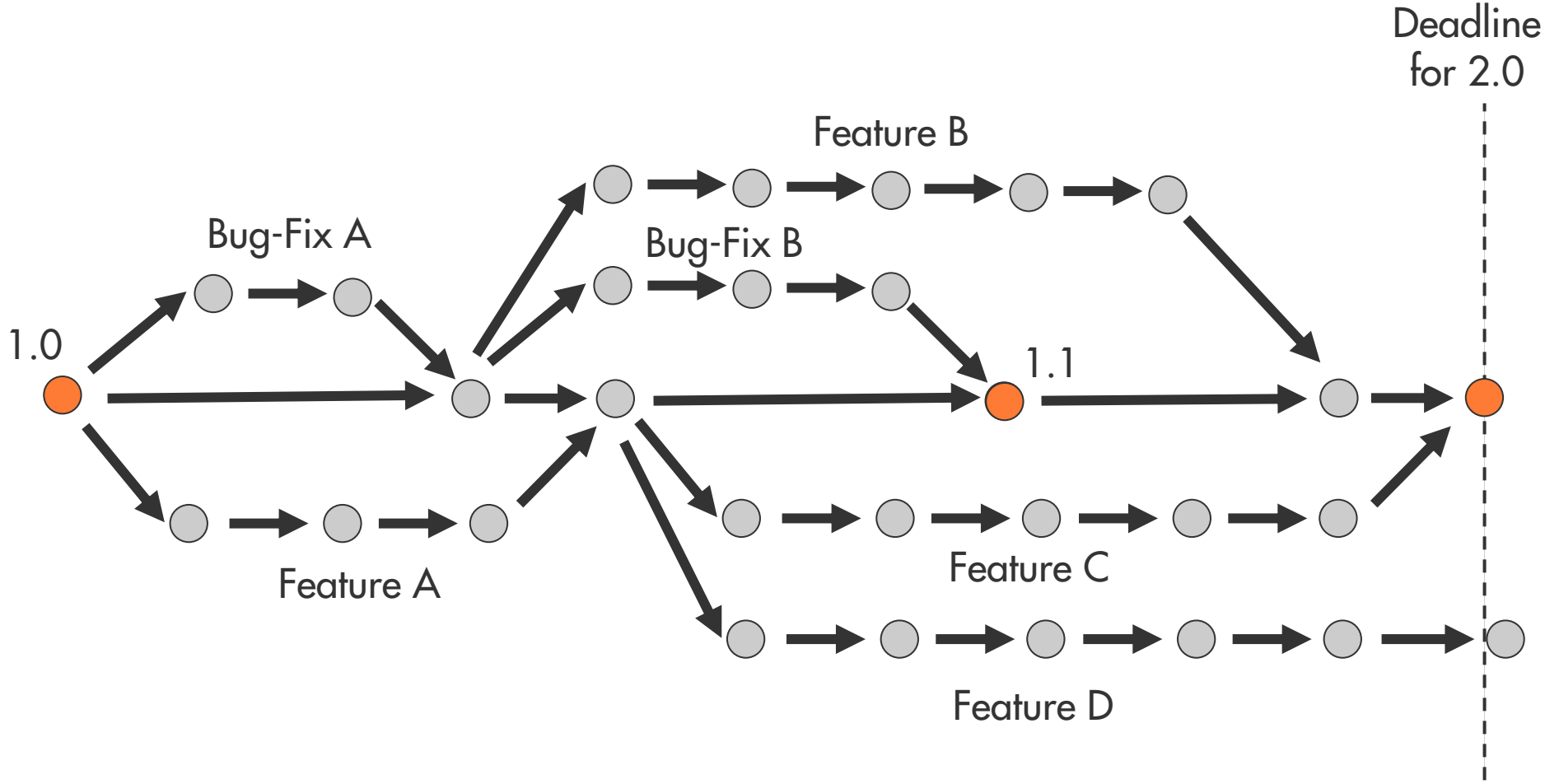


Commits

- ~50 commits per day
- ~70 committers in 6 weeks

Releases

- 1 major release every 6 weeks (patch releases in between)
- ~290 tickets per major release
- ~2500 tickets in the last year


Deadline
for 2.0
Feature B
Bug-Fix A
Bug-Fix B
1.0
1.1
Feature A
Feature C
Feature D

Teamscale / TS-18855

Deep nesting in RepositoryContentUpdaterBase

Edit | Comment | Assign | More | Back to New | Export

Details

Type:	Maintenance	Status:	DONE (View Workflow)
Priority:	Normal	Resolution:	Green
Component/s:	Backend	Fix Version/s:	Teamscale 5.1
Labels:	code-retrospective		
Affected Version:	master		
Merge Request:	https://git.cqse.eu/cqse/teamscale/merge_requests/5382		
PDash Task:	#4888		
Change Log:	-		
Maturity Level:	Development		

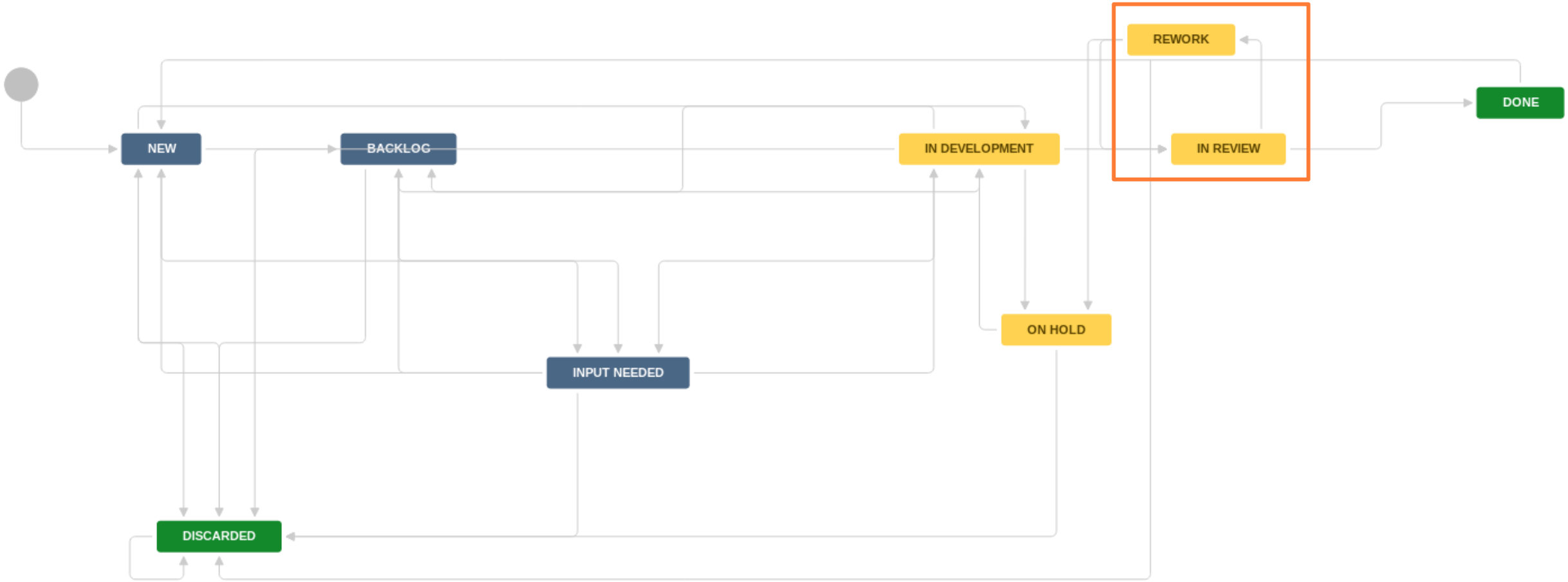
Description

People

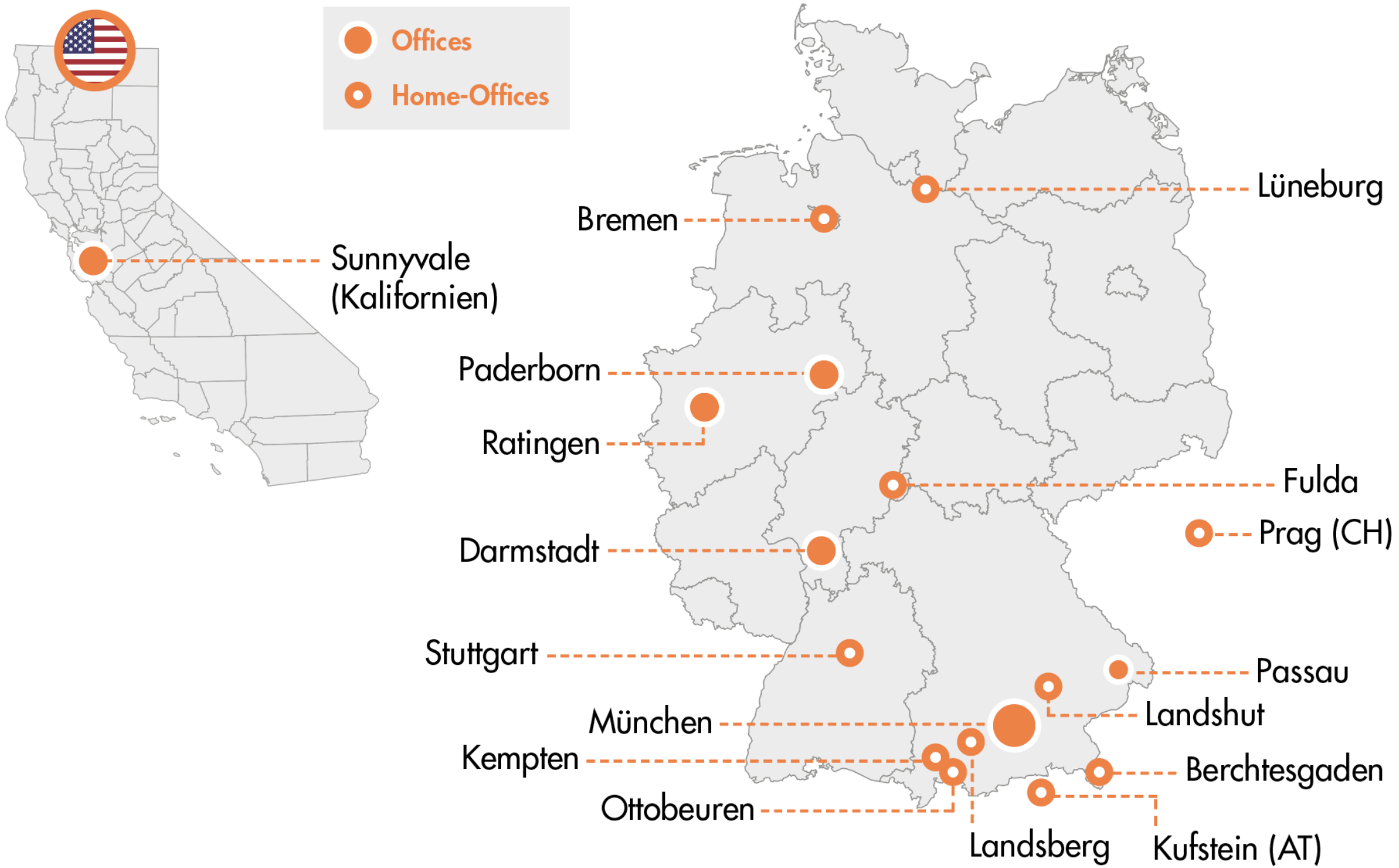
Assignee:	Lars Heinemann
	Assign to me
Reporter:	Lars Heinemann
QA-Contact:	Andreas Sewe
Votes:	0 Vote for this issue
Watchers:	1 Start watching this issue

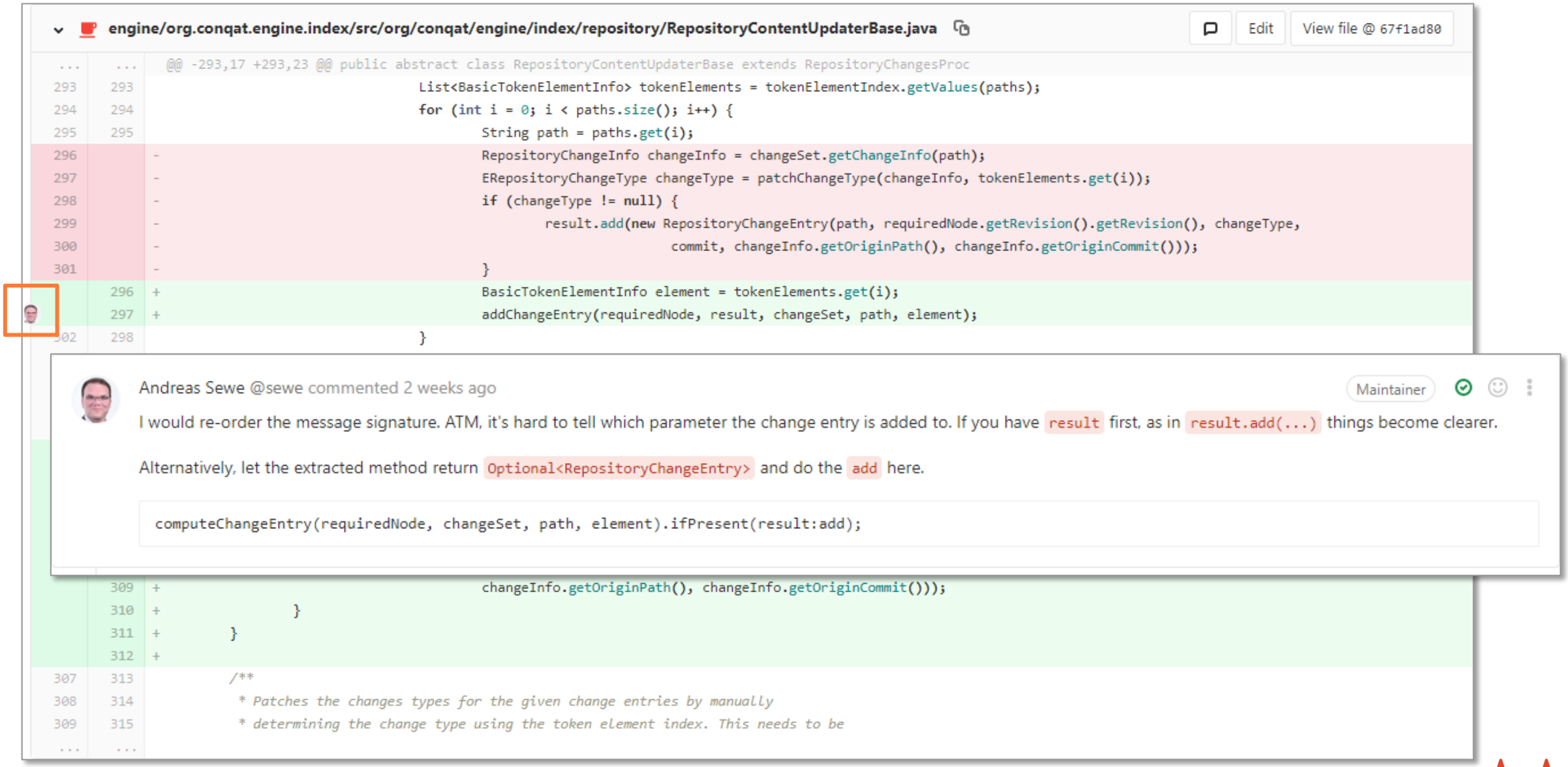
Dates

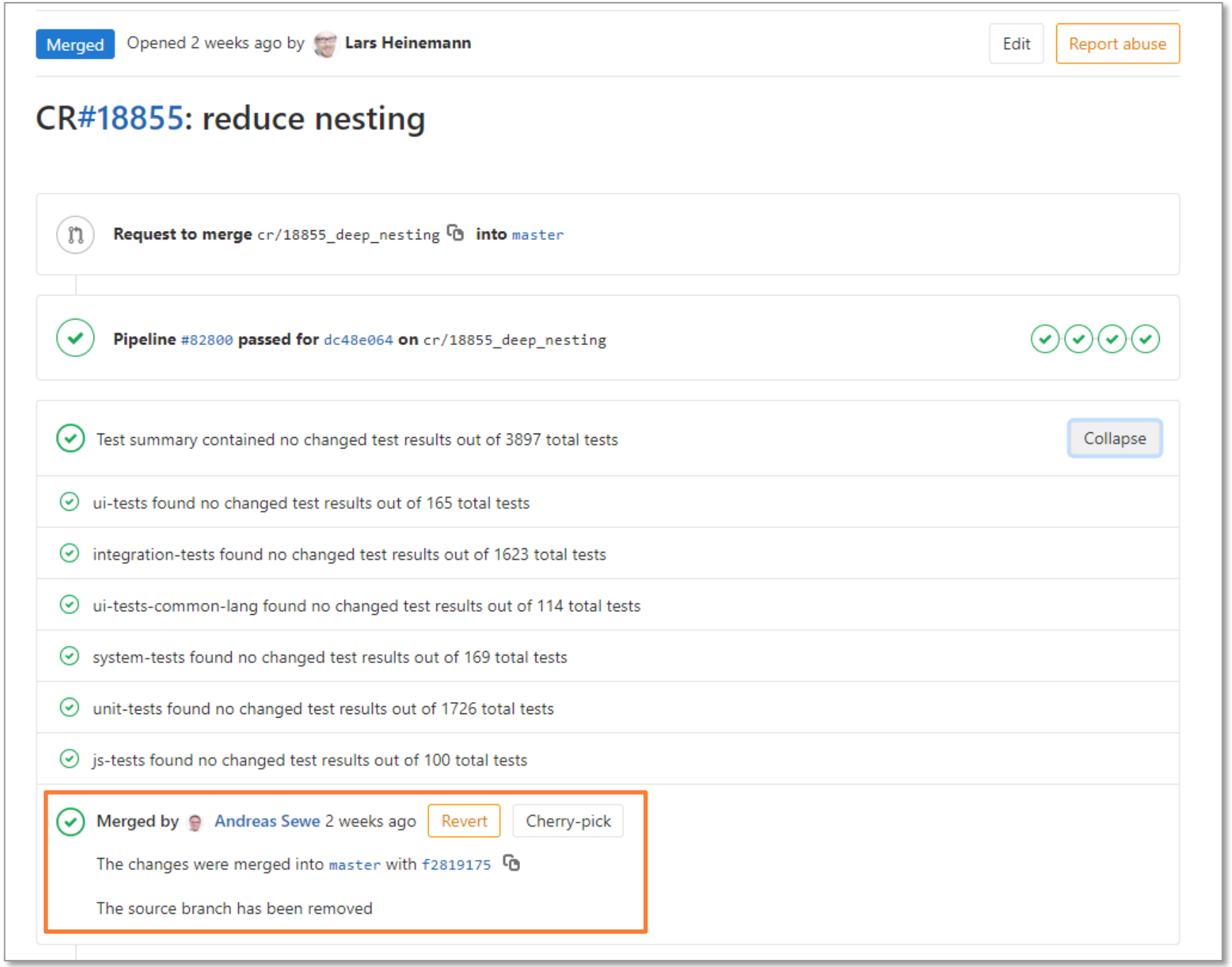
Created:	25/Apr/19 1:06 PM
Updated:	02/May/19 8:37 AM
Resolved:	26/Apr/19 10:58 AM

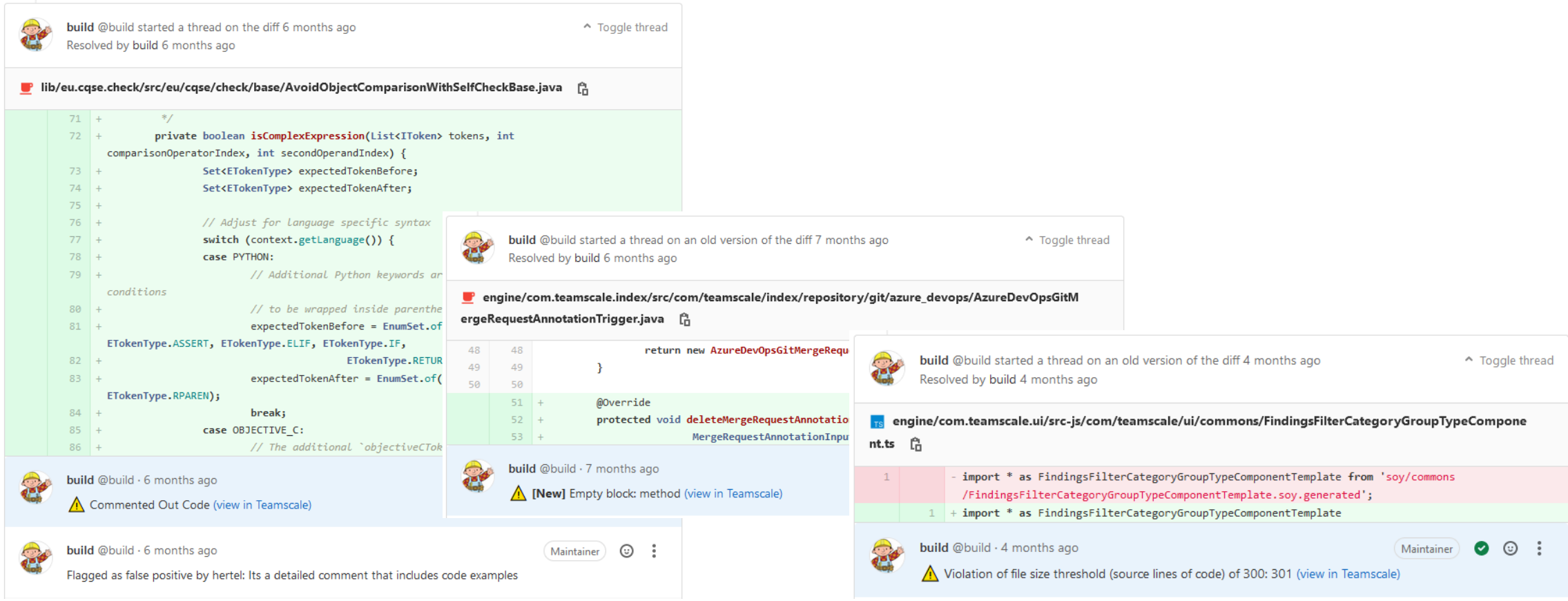

REWORK
DONE
NEW
BACKLOG
IN DEVELOPMENT
IN REVIEW
ON HOLD
INPUT NEEDED
DISCARDED

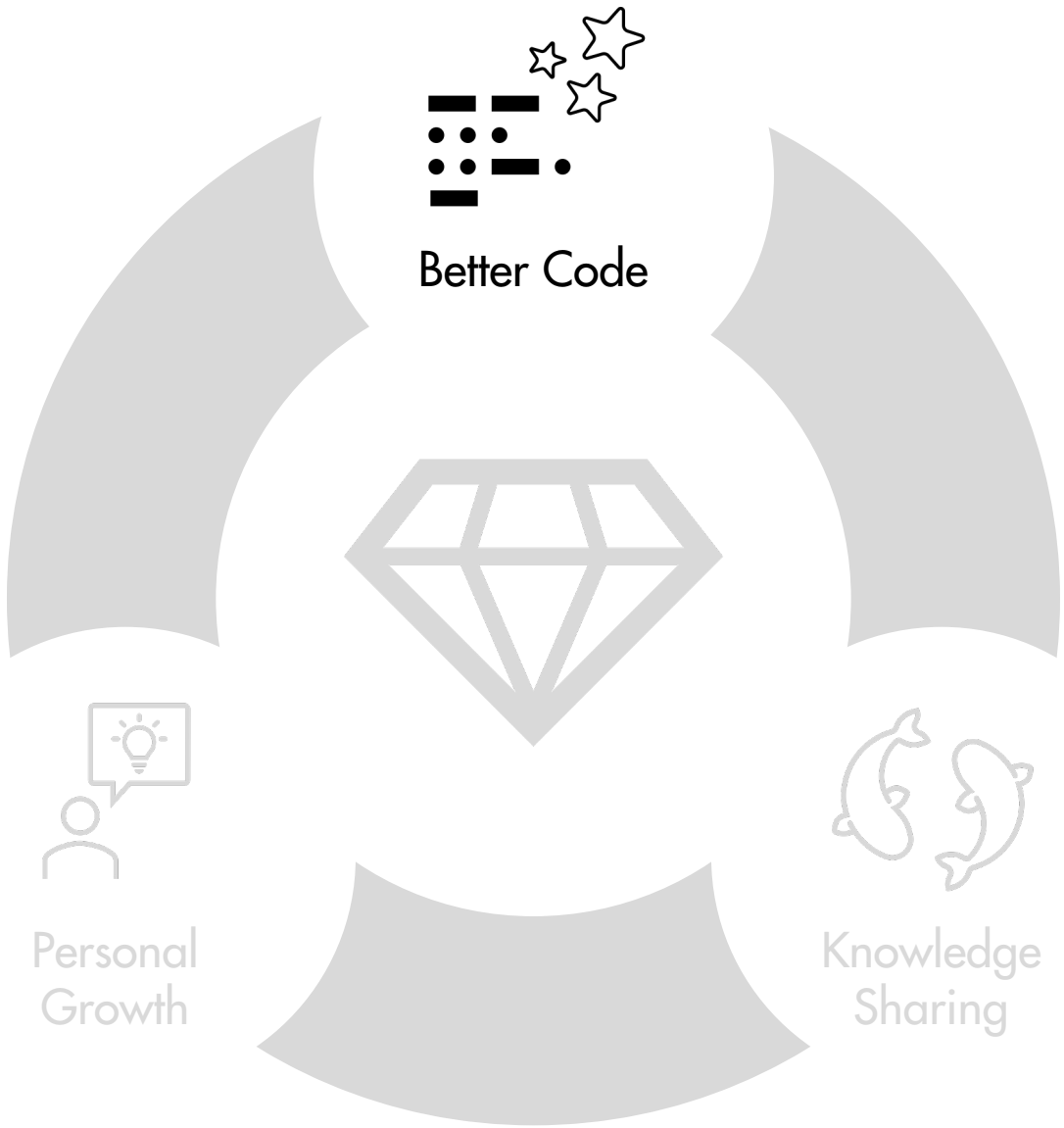
Transition	Time In Source Status
Lars Heinemann made transition - 25/Apr/19 1:07 PM NEW → BACKLOG	7s
Lars Heinemann made transition - 25/Apr/19 3:27 PM BACKLOG → IN DEVELOPMENT	2h 20m
Lars Heinemann made transition - 25/Apr/19 3:27 PM IN DEVELOPMENT → IN REVIEW	22s
Andreas Sewe made transition - 26/Apr/19 9:44 AM IN REVIEW → REWORK	18h 16m
Lars Heinemann made transition - 26/Apr/19 10:39 AM REWORK → IN REVIEW	55m 7s
Andreas Sewe made transition - 26/Apr/19 10:58 AM IN REVIEW → DONE	18m 57s


Offices
Home-Offices
Sunnyvale
(Kalifornien)
Bremen
Lüneburg
Paderborn
Ratingen
Fulda
Prag (CH)
Darmstadt
Stuttgart
Passau
Landshut
München
Kempten
Berchtesgaden
Ottobeuren
Landsberg
Kufstein (AT)

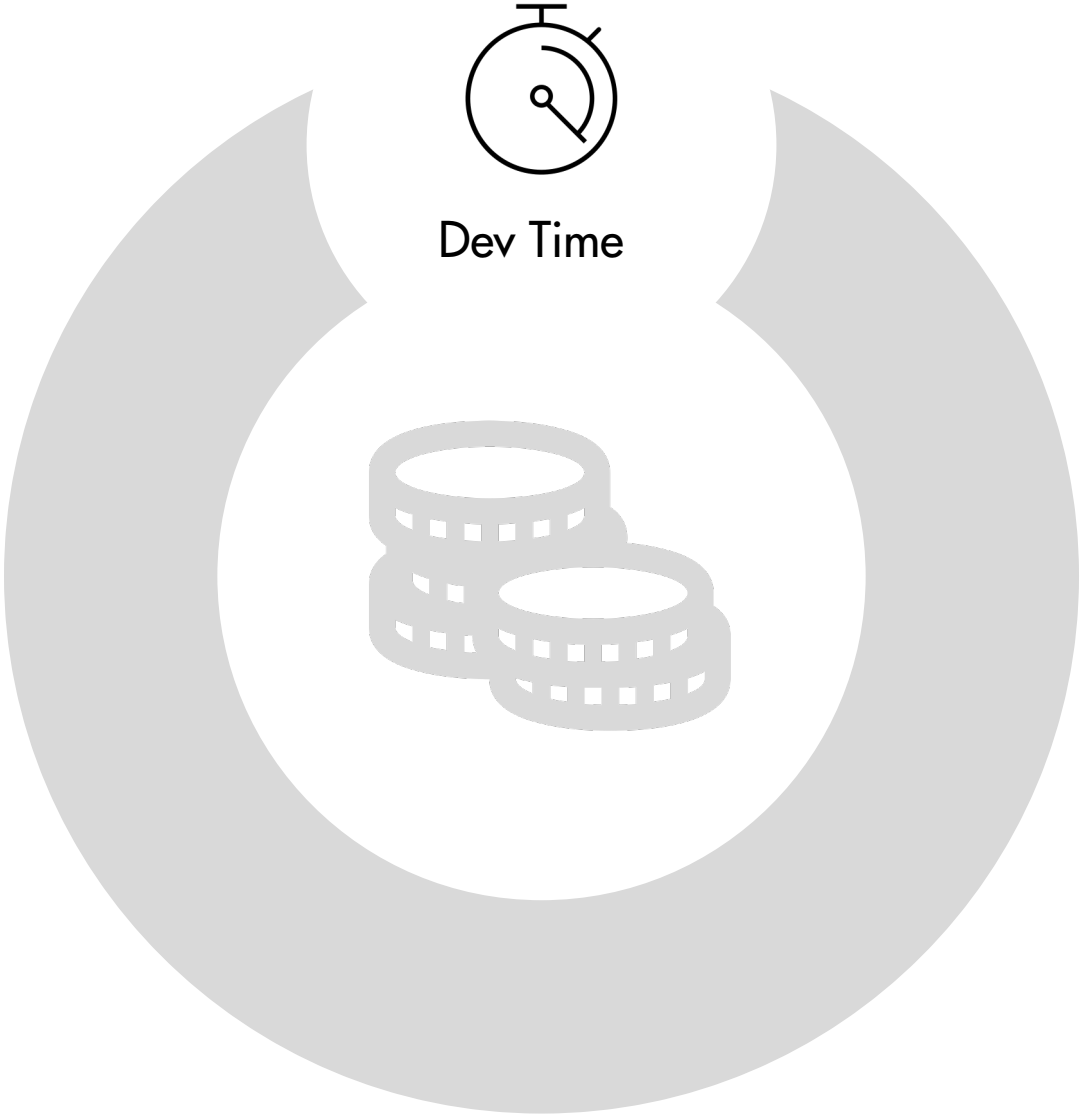

engine/org.conqat.engine.index/src/org/conqat/engine/index/repository/RepositoryContentUpdaterBase.java
Edit
View file @ 67f1ad80
@@ -293,17 +293,23 @@ public abstract class RepositoryContentUpdaterBase extends RepositoryChangesProc
List<BasicTokenElementInfo> tokenElements = tokenElementIndex.getValues(paths);
for (int i = 0; i < paths.size(); i++) {
String path = paths.get(i);
- RepositoryChangeInfo changeInfo = changeSet.getChangeInfo(path);
- ERepositoryChangeType changeType = patchChangeType(changeInfo, tokenElements.get(i));
- if (changeType != null) {
- result.add(new RepositoryChangeEntry(path, requiredNode.getRevision().getRevision(), changeType,
- commit, changeInfo.getOriginPath(), changeInfo.getOriginCommit()));
- }
+ BasicTokenElementInfo element = tokenElements.get(i);
+ addChangeEntry(requiredNode, result, changeSet, path, element);
}
Andreas Sewe @sewe commented 2 weeks ago
Maintainer
I would re-order the message signature. ATM, it's hard to tell which parameter the change entry is added to. If you have result first, as in result.add(...) things become clearer.
Alternatively, let the extracted method return Optional<RepositoryChangeEntry> and do the add here.
computeChangeEntry(requiredNode, changeSet, path, element).ifPresent(result::add);
changeInfo.getOriginPath(), changeInfo.getOriginCommit()));
}
}
/**
* Patches the changes types for the given change entries by manually
* determining the change type using the token element index. This needs to be


Merged
Opened 2 weeks ago by Lars Heinemann
Edit
Report abuse
CR#18855: reduce nesting
Request to merge cr/18855_deep_nesting into master
Pipeline #82800 passed for dc48e064 on cr/18855_deep_nesting
Test summary contained no changed test results out of 3897 total tests
Collapse
ui-tests found no changed test results out of 165 total tests
integration-tests found no changed test results out of 1623 total tests
ui-tests-common-lang found no changed test results out of 114 total tests
system-tests found no changed test results out of 169 total tests
unit-tests found no changed test results out of 1726 total tests
js-tests found no changed test results out of 100 total tests
Merged by Andreas Sewe 2 weeks ago
Revert
Cherry-pick
The changes were merged into master with f2819175
The source branch has been removed

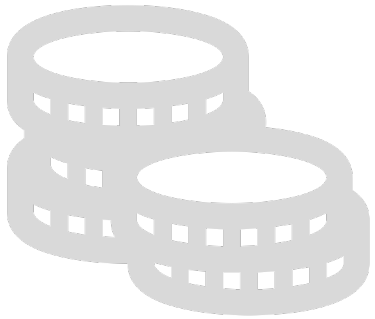

build @build started a thread on the diff 6 months ago
Resolved by build 6 months ago
Toggle thread
lib/eu.cqse.check/src/eu/cqse/check/base/AvoidObjectComparisonWithSelfCheckBase.java
71 + */
72 + private boolean isComplexExpression(List<IToken> tokens, int comparisonOperatorIndex, int secondOperandIndex) {
73 + Set<ETokenType> expectedTokenBefore;
74 + Set<ETokenType> expectedTokenAfter;
75 +
76 + // Adjust for language specific syntax
77 + switch (context.getLanguage()) {
78 + case PYTHON:
79 + // Additional Python keywords ar conditions
80 + // to be wrapped inside parenthe
81 + expectedTokenBefore = EnumSet.of ETokenType.ASSERT, ETokenType.ELIF, ETokenType.IF,
82 + ETokenType.RETUR
83 + expectedTokenAfter = EnumSet.of(ETokenType.RPAREN);
84 + break;
85 + case OBJECTIVE_C:
86 + // The additional `objectiveCTok
build @build · 6 months ago
Commented Out Code (view in Teamscale)
build @build · 6 months ago
Maintainer
Flagged as false positive by hertel: Its a detailed comment that includes code examples
build @build started a thread on an old version of the diff 7 months ago
Resolved by build 6 months ago
Toggle thread
engine/com.teamscale.index/src/com/teamscale/index/repository/git/azure_devops/AzureDevOpsGitMergeRequestAnnotationTrigger.java
48 48 return new AzureDevOpsGitMergeRequ
49 49 }
50 50
51 + @Override
52 + protected void deleteMergeRequestAnnotatio
53 + MergeRequestAnnotationInpu
build @build · 7 months ago
[New] Empty block: method (view in Teamscale)
build @build started a thread on an old version of the diff 4 months ago
Resolved by build 4 months ago
Toggle thread
engine/com.teamscale.ui/src-js/com/teamscale/ui/commons/FindingsFilterCategoryGroupTypeComponent.ts
1 - import * as FindingsFilterCategoryGroupTypeComponentTemplate from 'soy/commons/FindingsFilterCategoryGroupTypeComponentTemplate.soy.generated';
1 + import * as FindingsFilterCategoryGroupTypeComponentTemplate
build @build · 4 months ago
Maintainer
Violation of file size threshold (source lines of code) of 300: 301 (view in Teamscale)



How (much) do code reviews improve our code?

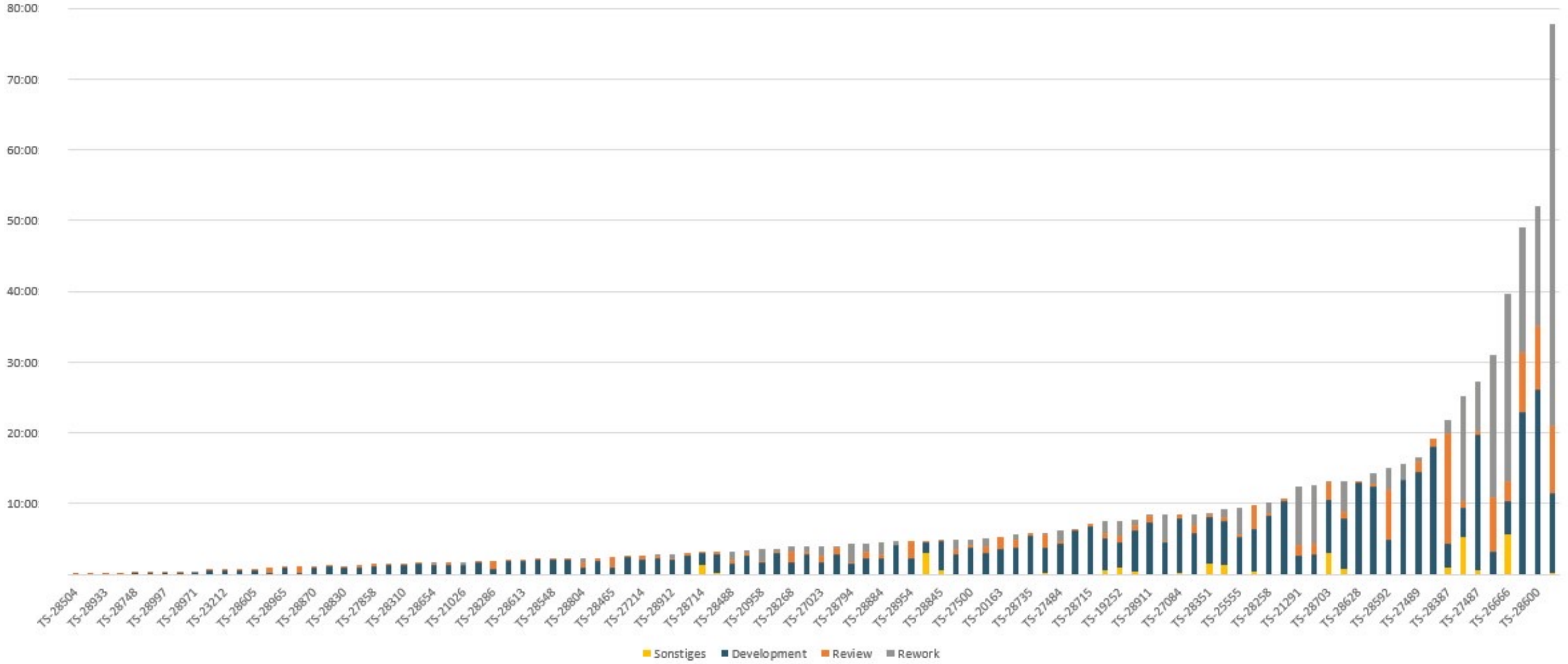


How much time do we invest into code reviews?

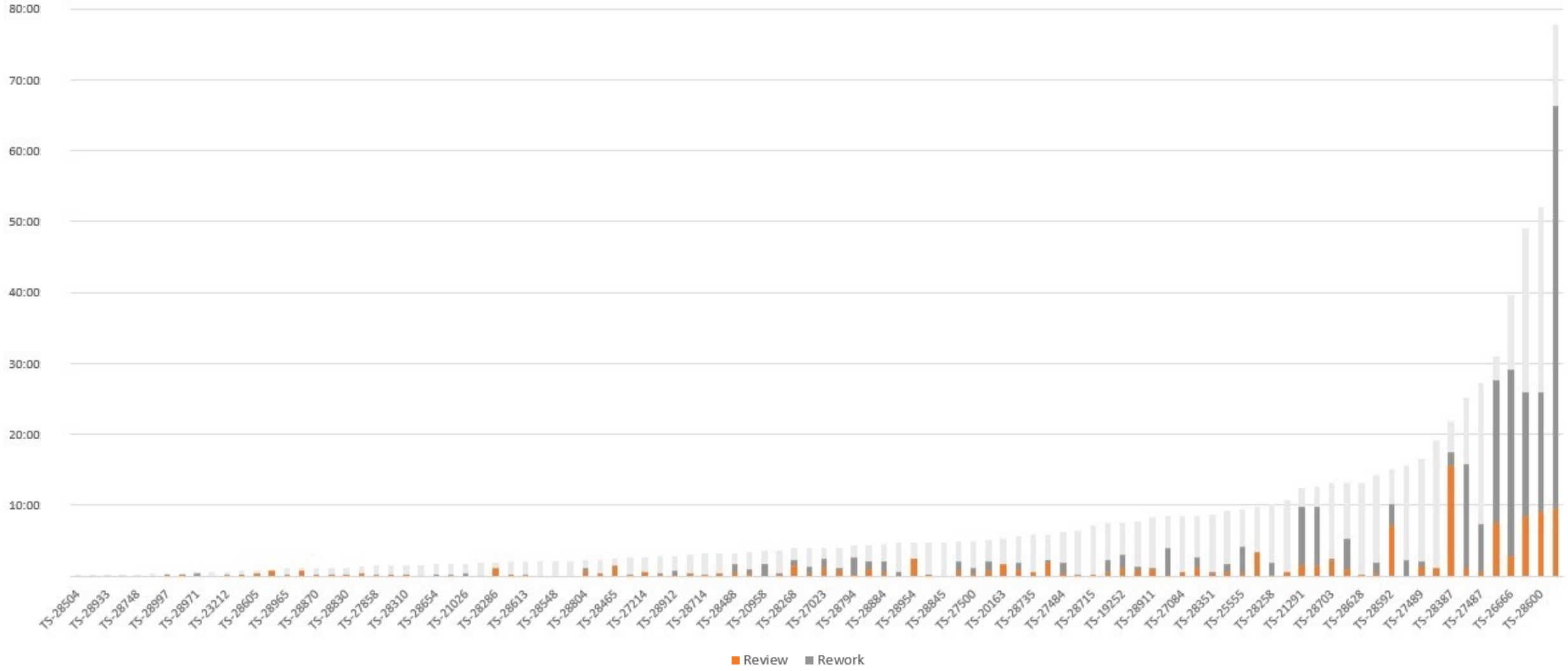


How much time do we invest into code reviews?

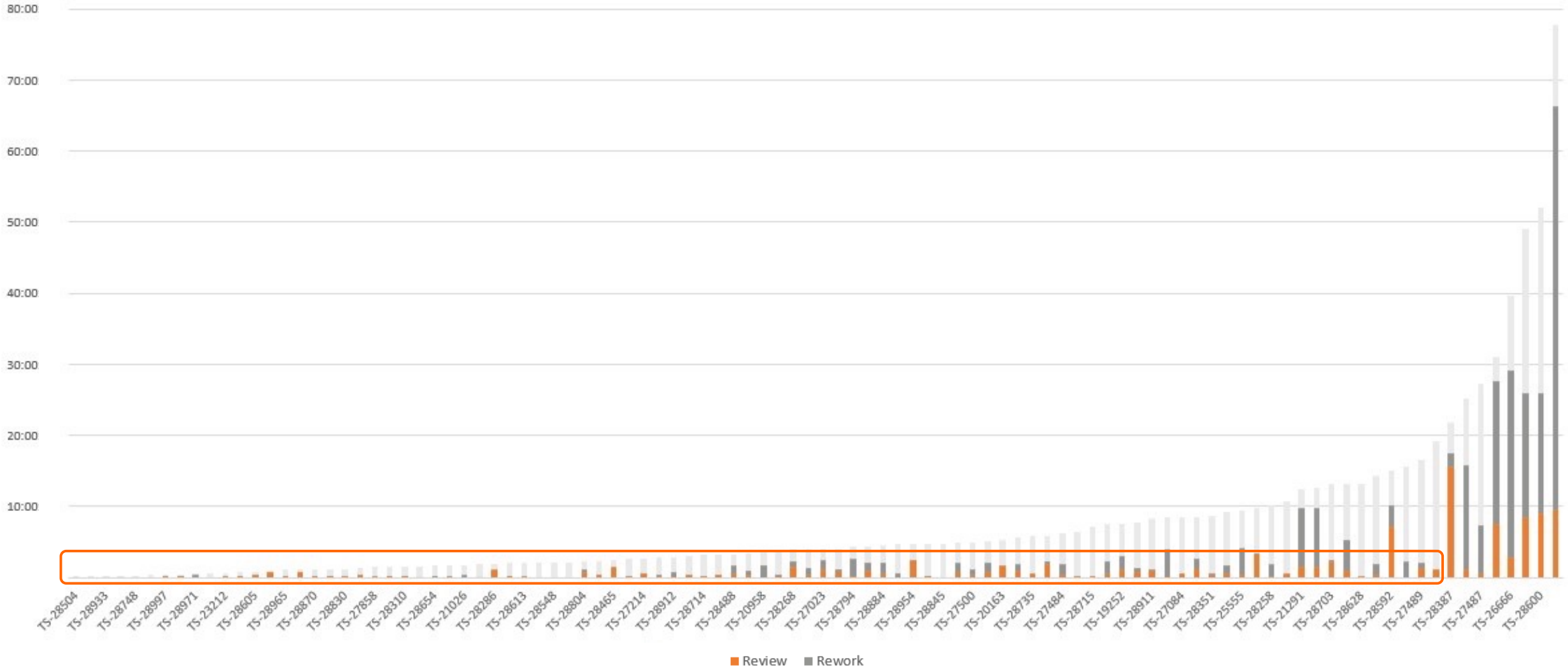
Time Usage per Ticket



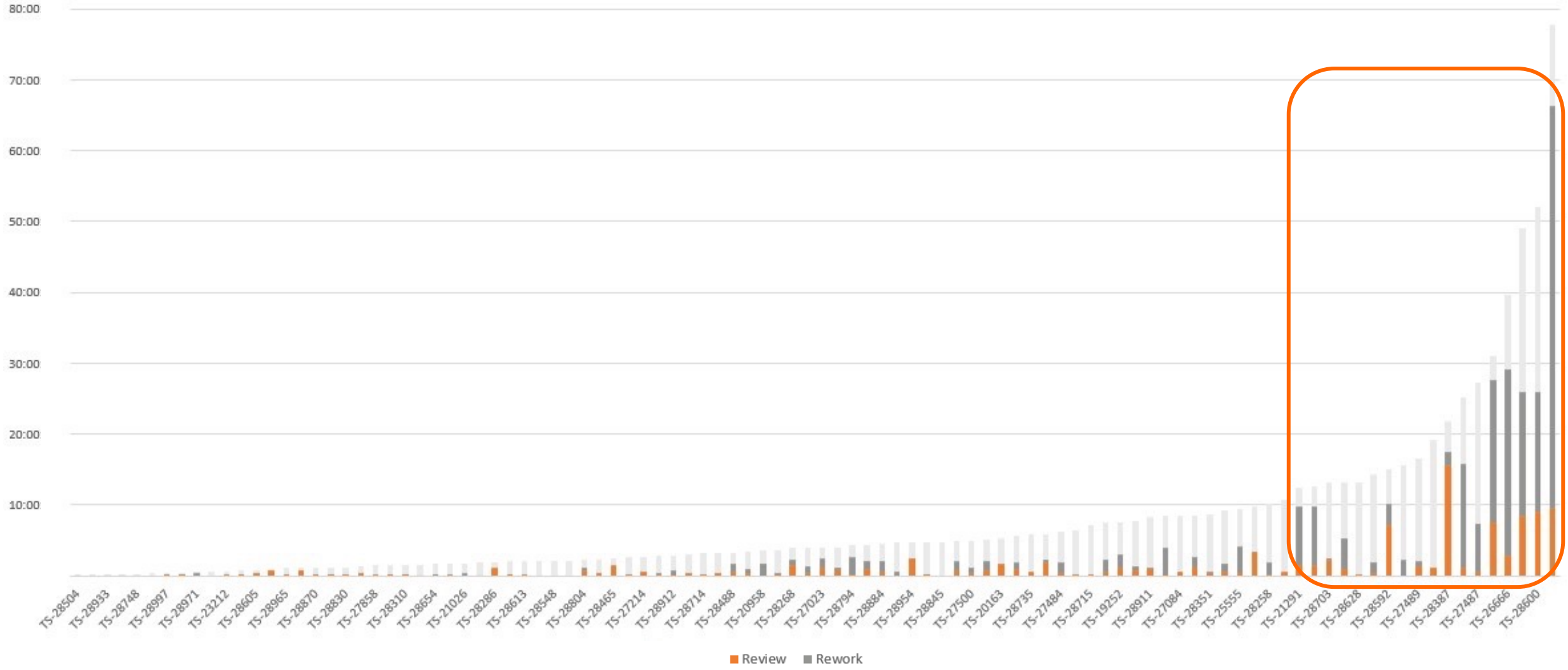
Review and Rework Time per Ticket



Review and Rework Time per Ticket



Review and Rework Time per Ticket



TS-26659

Teamscale / TS-26659

Intellij Plugin: License expiry message not displayed as reason for stalled Pre-Commit analysis

Edit | Comment | Assign | Back to New

Type:	Bug	Status:	DONE (View workflow)
Priority:	High	Resolution:	Green
Components:	IntelliJ Plugin	Fix versions:	Teamscale 7.6
Labels:	students		
Affected Version:	6.6.4		
Customer:			
PDash Task:	#4886		
Change Log:	Fix: IDE plugins constantly request unavailable pre-commit results due to expired license		

Teamscale: Teamscale Info

Project name	Findings balance	Test Gap
CQSE All	31 removed	100.00%

Dev Squad:	Denali
Freshdesk URL:	https://cqse.freshdesk.com/a/tickets/19594
Merge Request:	https://git.cqse.eu/cqse/teamscale/-/merge_requests/11582
Maturity Level:	Customer Visible

Description

People

Assignee:	Korbinian Weidinger
	Assign to me
Reporter:	
QA-Contact:	Andreas Sewe
Votes:	0 Vote for this issue
Watchers:	3 Start watching this issue

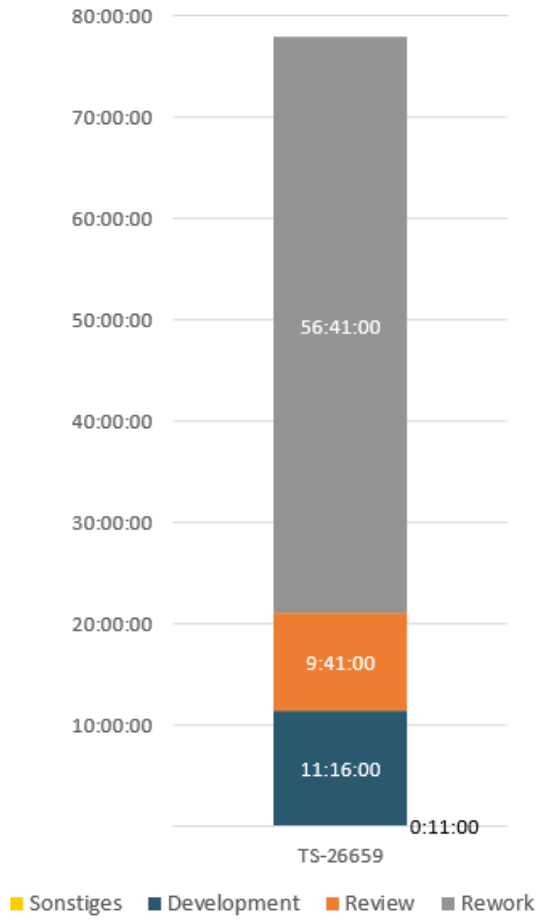
Dates

Created:	31/Mar/21 6:15 PM
Updated:	16/Feb/22 3:01 PM
Resolved:	25/Nov/21 5:55 PM

Automation

Time Tracking

Agile



TS-26659

Teamscale / TS-26659

Intellij Plugin: License expiry message not displayed as reason for stalled Pre-Commit analysis

Edit | Comment | Assign | Back to New

Type:	Bug	Status:	DONE (View workflow)
Priority:	High	Resolution:	Green
Components:	IntelliJ Plugin	Fix versions:	Teamscale 7.6
Labels:	students		
Affected Version:	6.6.4		
Customer:			
PDash Task:	#4886		
Change Log:	Fix: IDE plugins constantly request unavailable pre-commit results due to expired license		

Teamscale: Teamscale Info

Project name	Findings balance	Test Gap
CQSE All	31 removed	100.00%

Dev Squad:	Denali
Freshdesk URL:	https://cqse.freshdesk.com/a/tickets/19594
Merge Request:	https://git.cqse.eu/cqse/teamscale/-/merge_requests/11582
Maturity Level:	Customer Visible

Description

People

Assignee:	Korbinian Weidinger
	Assign to me
Reporter:	
QA-Contact:	Andreas Sewe
Votes:	0 Vote for this issue
Watchers:	3 Start watching this issue

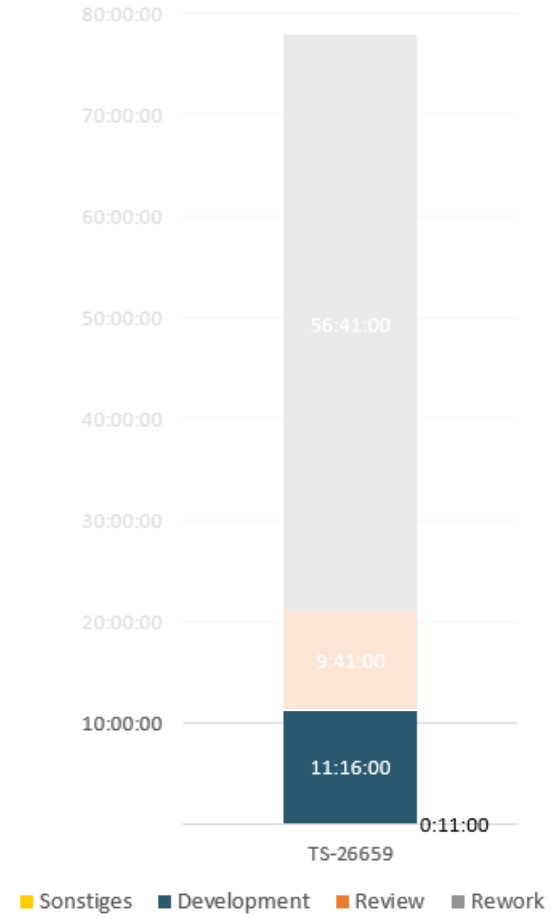
Dates

Created:	31/Mar/21 6:15 PM
Updated:	16/Feb/22 3:01 PM
Resolved:	25/Nov/21 5:55 PM

Automation

Time Tracking

Agile



TS-26659

Teamscale / TS-26659

Intellij Plugin: License expiry message not displayed as reason for stalled Pre-Commit analysis

Edit | Comment | Assign | Back to New

Type:	Bug	Status:	DONE (View workflow)
Priority:	High	Resolution:	Green
Components:	IntelliJ Plugin	Fix versions:	Teamscale 7.6
Labels:	students		
Affected Version:	6.6.4		
Customer:			
PDash Task:	#4886		
Change Log:	Fix: IDE plugins constantly request unavailable pre-commit results due to expired license		

Teamscale: Teamscale Info

Project name	Findings balance	Test Gap
CQSE All	31 removed	100.00%

Dev Squad:	Denali
Freshdesk URL:	https://cqse.freshdesk.com/a/tickets/19594
Merge Request:	https://git.cqse.eu/cqse/teamscale/-/merge_requests/11582
Maturity Level:	Customer Visible

Description

People

Assignee:	Korbinian Weidinger
	Assign to me
Reporter:	
QA-Contact:	Andreas Sewe
Votes:	0 Vote for this issue
Watchers:	3 Start watching this issue

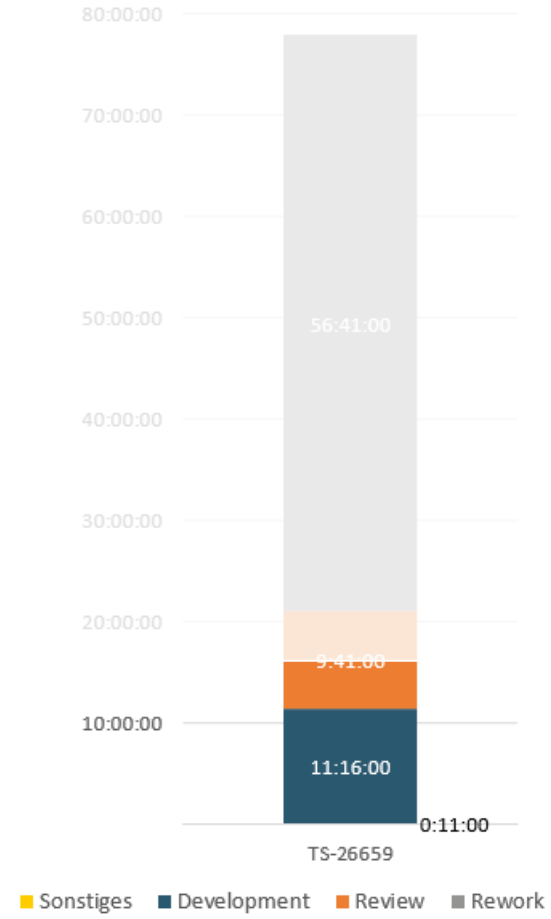
Dates

Created:	31/Mar/21 6:15 PM
Updated:	16/Feb/22 3:01 PM
Resolved:	25/Nov/21 5:55 PM

Automation

Time Tracking

Agile



TS-26659

Teamscale / TS-26659

Intellij Plugin: License expiry message not displayed as reason for stalled Pre-Commit analysis

Edit | Comment | Assign | Back to New

Type:	Bug	Status:	DONE (View workflow)
Priority:	High	Resolution:	Green
Components:	IntelliJ Plugin	Fix versions:	Teamscale 7.6
Labels:	students		
Affected Version:	6.6.4		
Customer:			
PDash Task:	#4886		
Change Log:	Fix: IDE plugins constantly request unavailable pre-commit results due to expired license		

Teamscale: Teamscale Info

Project name	Findings balance	Test Gap
CQSE All	31 removed	100.00%

Dev Squad:	Denali
Freshdesk URL:	https://cqse.freshdesk.com/a/tickets/19594
Merge Request:	https://git.cqse.eu/cqse/teamscale/-/merge_requests/11582
Maturity Level:	Customer Visible

Description

People

Assignee:	Korbinian Weidinger
	Assign to me
Reporter:	
QA-Contact:	Andreas Sewe
Votes:	0 Vote for this issue
Watchers:	3 Start watching this issue

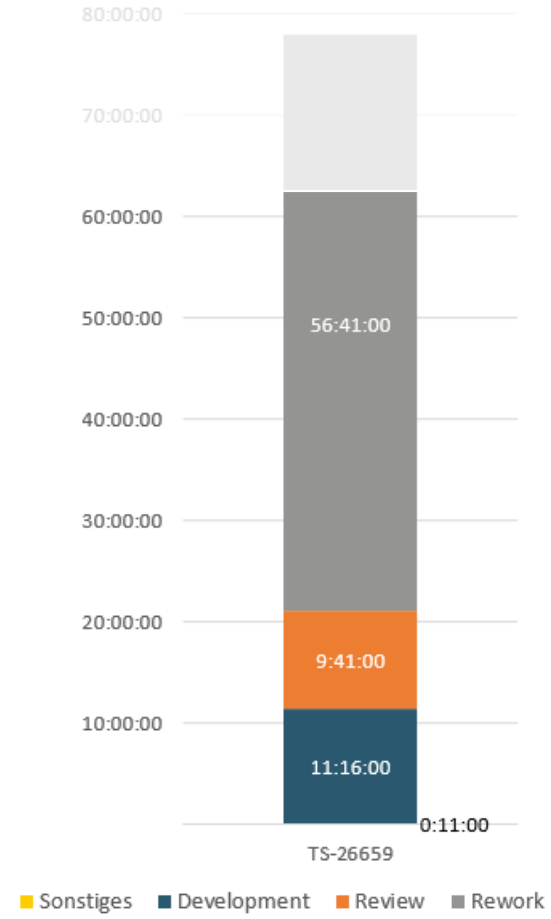
Dates

Created:	31/Mar/21 6:15 PM
Updated:	16/Feb/22 3:01 PM
Resolved:	25/Nov/21 5:55 PM

Automation

Time Tracking

Agile



TS-26659

Teamscale / TS-26659

Intellij Plugin: License expiry message not displayed as reason for stalled Pre-Commit analysis

Edit | Comment | Assign | Back to New

Type:	Bug	Status:	DONE (View workflow)
Priority:	High	Resolution:	Green
Components:	IntelliJ Plugin	Fix versions:	Teamscale 7.6
Labels:	students		
Affected Version:	6.6.4		
Customer:			
PDash Task:	#4886		
Change Log:	Fix: IDE plugins constantly request unavailable pre-commit results due to expired license		

Teamscale: Teamscale Info

Project name	Findings balance	Test Gap
CQSE All	31 removed	100.00%

Dev Squad:	Denali
Freshdesk URL:	https://cqse.freshdesk.com/a/tickets/19594
Merge Request:	https://git.cqse.eu/cqse/teamscale/-/merge_requests/11582
Maturity Level:	Customer Visible

Description

People

Assignee:	Korbinian Weidinger
	Assign to me
Reporter:	
QA-Contact:	Andreas Sewe
Votes:	0 Vote for this issue
Watchers:	3 Start watching this issue

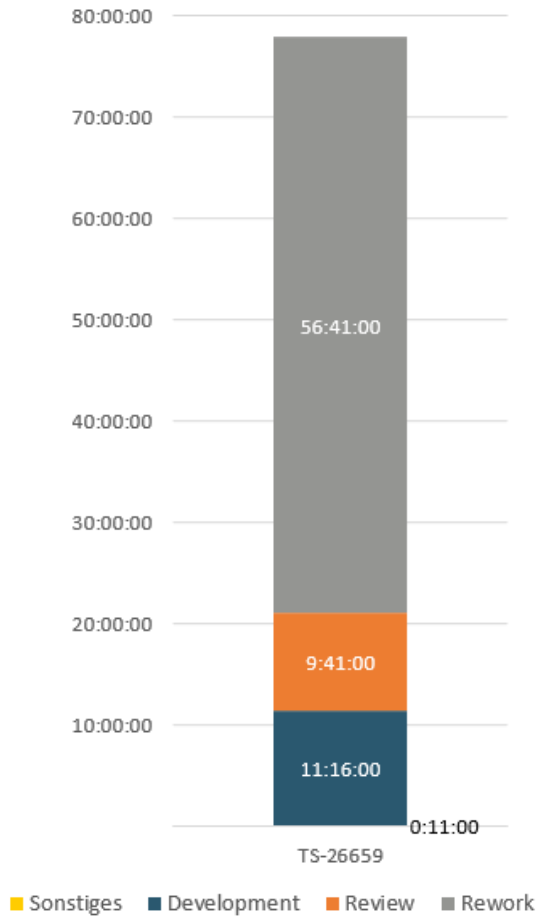
Dates

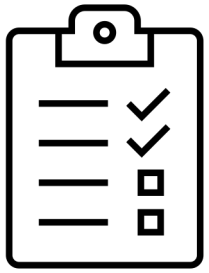
Created:	31/Mar/21 6:15 PM
Updated:	16/Feb/22 3:01 PM
Resolved:	25/Nov/21 5:55 PM

Automation

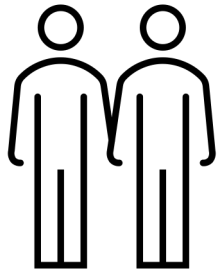
Time Tracking

Agile



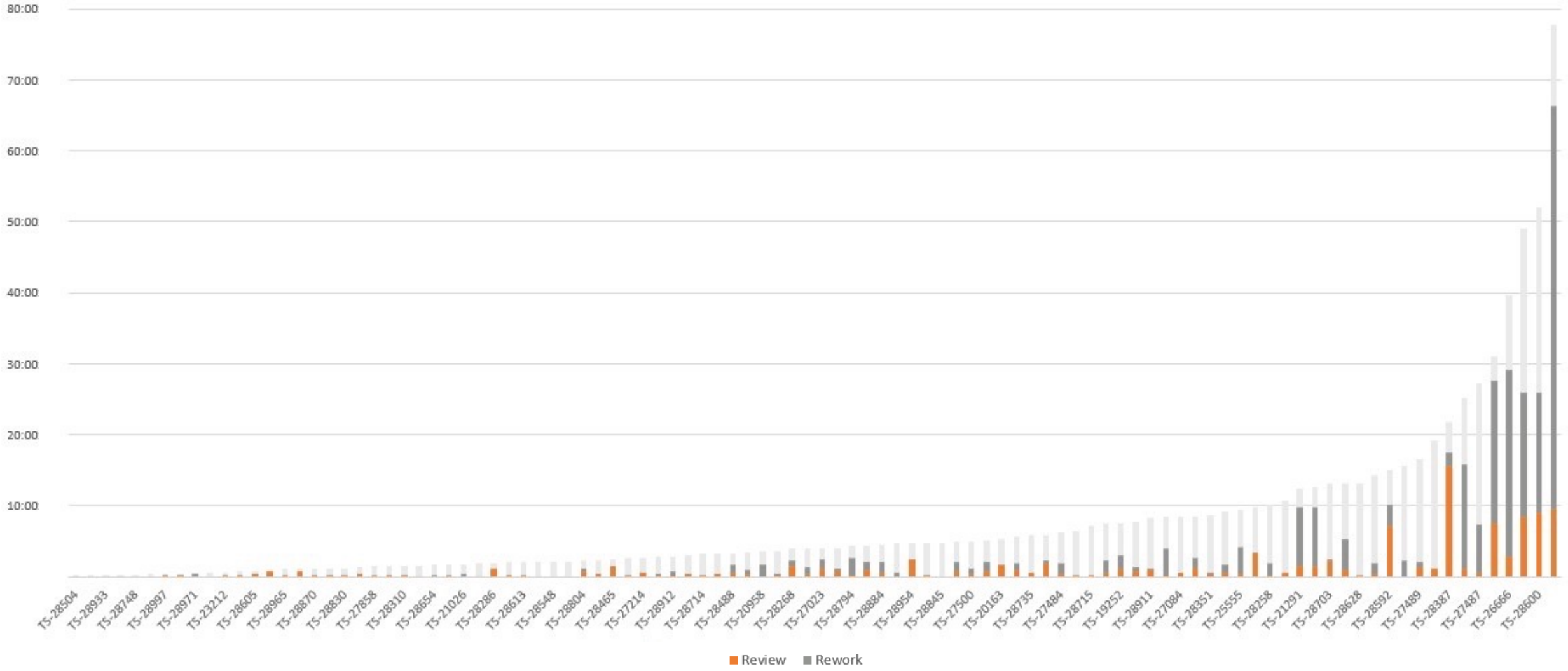


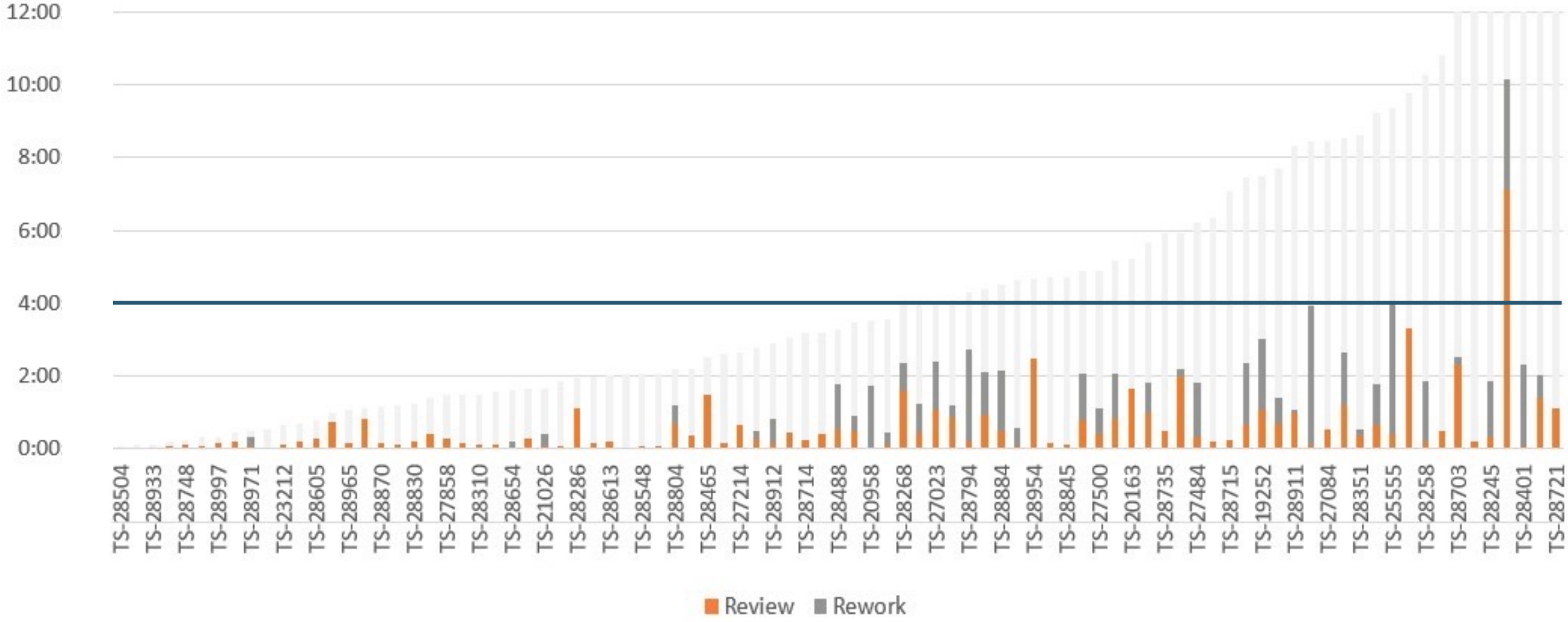
Reviewer
Changed
Requirements

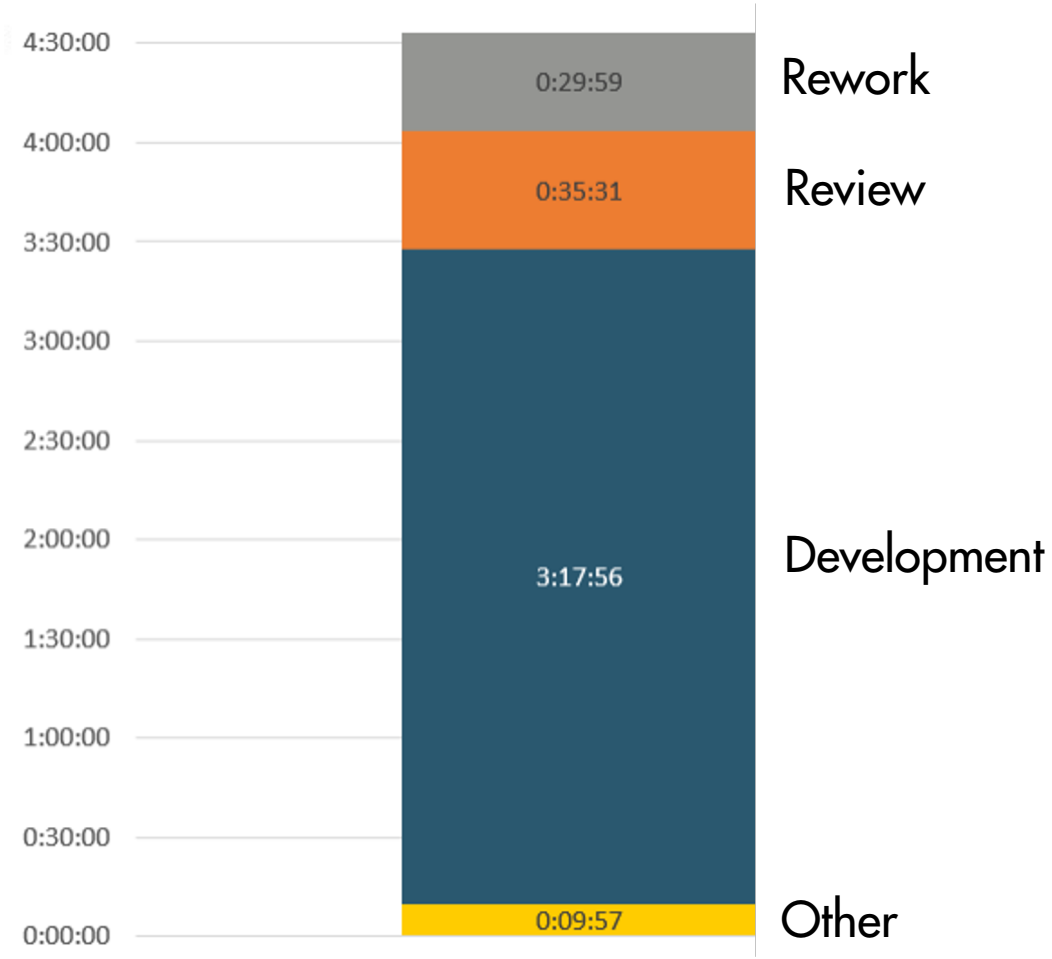


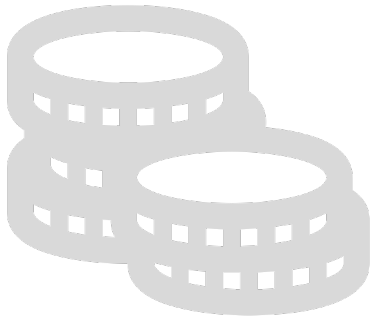
Developing
Together
(early feedback)

Time Usage per Ticket



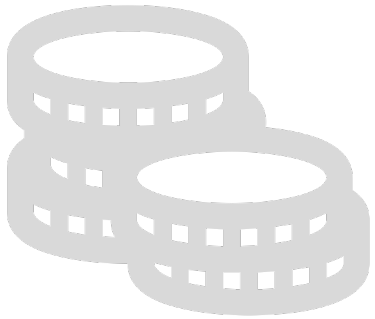

12:00
10:00
8:00
6:00
4:00
2:00
0:00
TS-28504
TS-28933
TS-28748
TS-28997
TS-28971
TS-23212
TS-28605
TS-28965
TS-28870
TS-28830
TS-27858
TS-28310
TS-28654
TS-21026
TS-28286
TS-28613
TS-28548
TS-28804
TS-28465
TS-27214
TS-28912
TS-28714
TS-28488
TS-20958
TS-28268
TS-27023
TS-28794
TS-28884
TS-28954
TS-28845
TS-27500
TS-20163
TS-28735
TS-27484
TS-28715
TS-19252
TS-28911
TS-27084
TS-28351
TS-25555
TS-28258
TS-28703
TS-28245
TS-28401
TS-28721
Review
Rework


4:30:00
4:00:00
3:30:00
3:00:00
2:30:00
2:00:00
1:30:00
1:00:00
0:30:00
0:00:00
0:29:59
0:35:31
3:17:56
0:09:57
Rework
Review
Development
Other



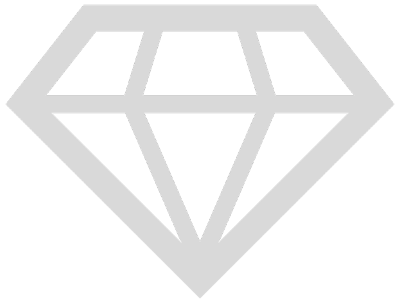
How much time do we invest into code reviews?

25% of the ticket time



How much time do we invest into code reviews?

33% additional time



How (much) do code reviews improve our code?

engine/org.conqat.engine.index/src/org/conqat/engine/index/repository/RepositoryContentUpdaterBase.java

Edit | View file @ 67f1ad80

```
@@ -293,17 +293,23 @@ public abstract class RepositoryContentUpdaterBase extends RepositoryChangesProc
                                List<BasicTokenElementInfo> tokenElements = tokenElementIndex.getValues(paths);
                                for (int i = 0; i < paths.size(); i++) {
                                        String path = paths.get(i);
-                                       RepositoryChangeInfo changeInfo = changeSet.getChangeInfo(path);
-                                       ERepositoryChangeType changeType = patchChangeType(changeInfo, tokenElements.get(i));
-                                       if (changeType != null) {
-                                               result.add(new RepositoryChangeEntry(path, requiredNode.getRevision().getRevision(), changeType,
-                                                               commit, changeInfo.getOriginPath(), changeInfo.getOriginCommit()));
-                                       }
+                                       BasicTokenElementInfo element = tokenElements.get(i);
+                                       addChangeEntry(requiredNode, result, changeSet, path, element);
                                }
                        }
+
+       private void addChangeEntry(RequiredNode requiredNode, List<RepositoryChangeEntry> result, ChangeSet changeSet,
+                       String path, BasicTokenElementInfo element) {
+               RepositoryChangeInfo changeInfo = changeSet.getChangeInfo(path);
+               ERepositoryChangeType changeType = patchChangeType(changeInfo, element);
+               if (changeType != null) {
+                       result.add(new RepositoryChangeEntry(path, requiredNode.getRevision().getRevision(), changeType,
+                                       commit, changeInfo.getOriginPath(), changeInfo.getOriginCommit()));
+               }
+       }
+
        /**
         * Patches the changes types for the given change entries by manually
         * determining the change type using the token element index. This needs to be
```

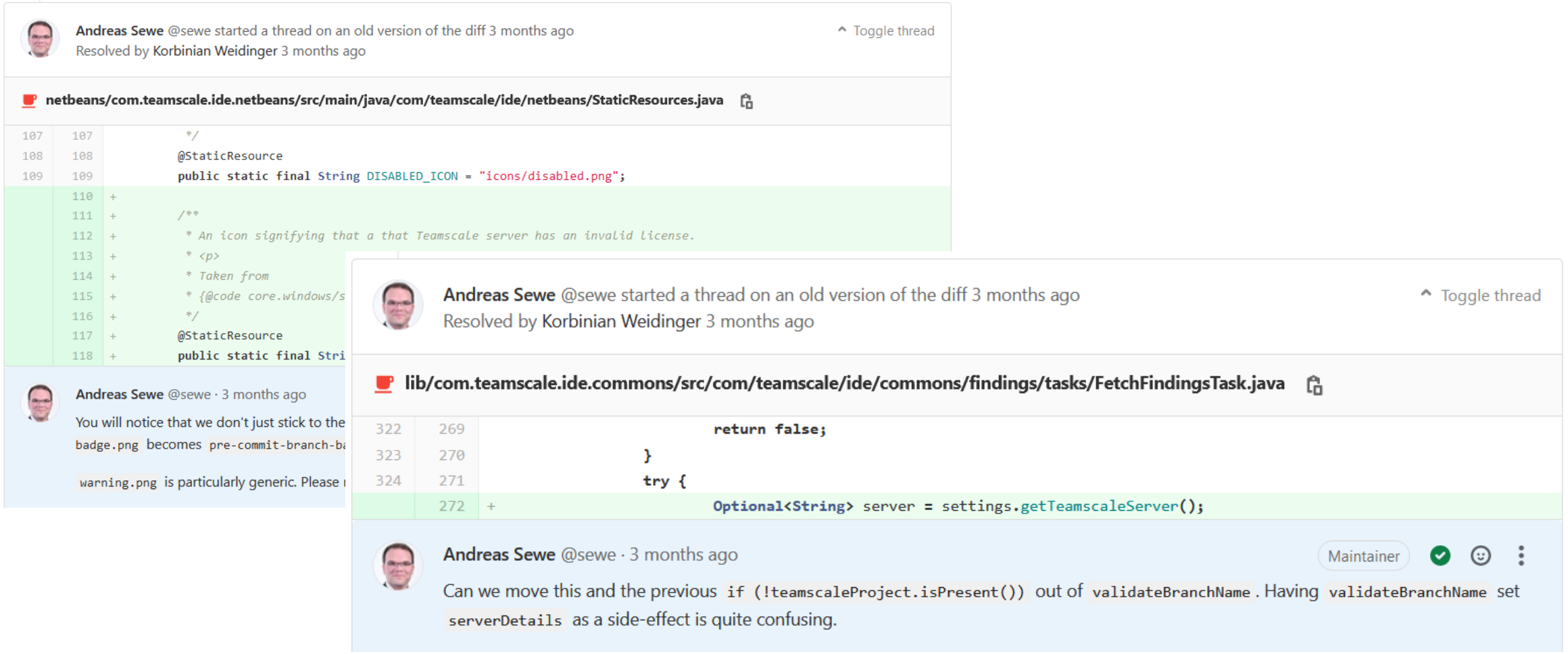
Andreas Sewe @sewe commented 2 weeks ago — Maintainer

I would re-order the message signature. ATM, it's hard to tell which parameter the change entry is added to. If you have `result` first, as in `result.add(...)` things become clearer.

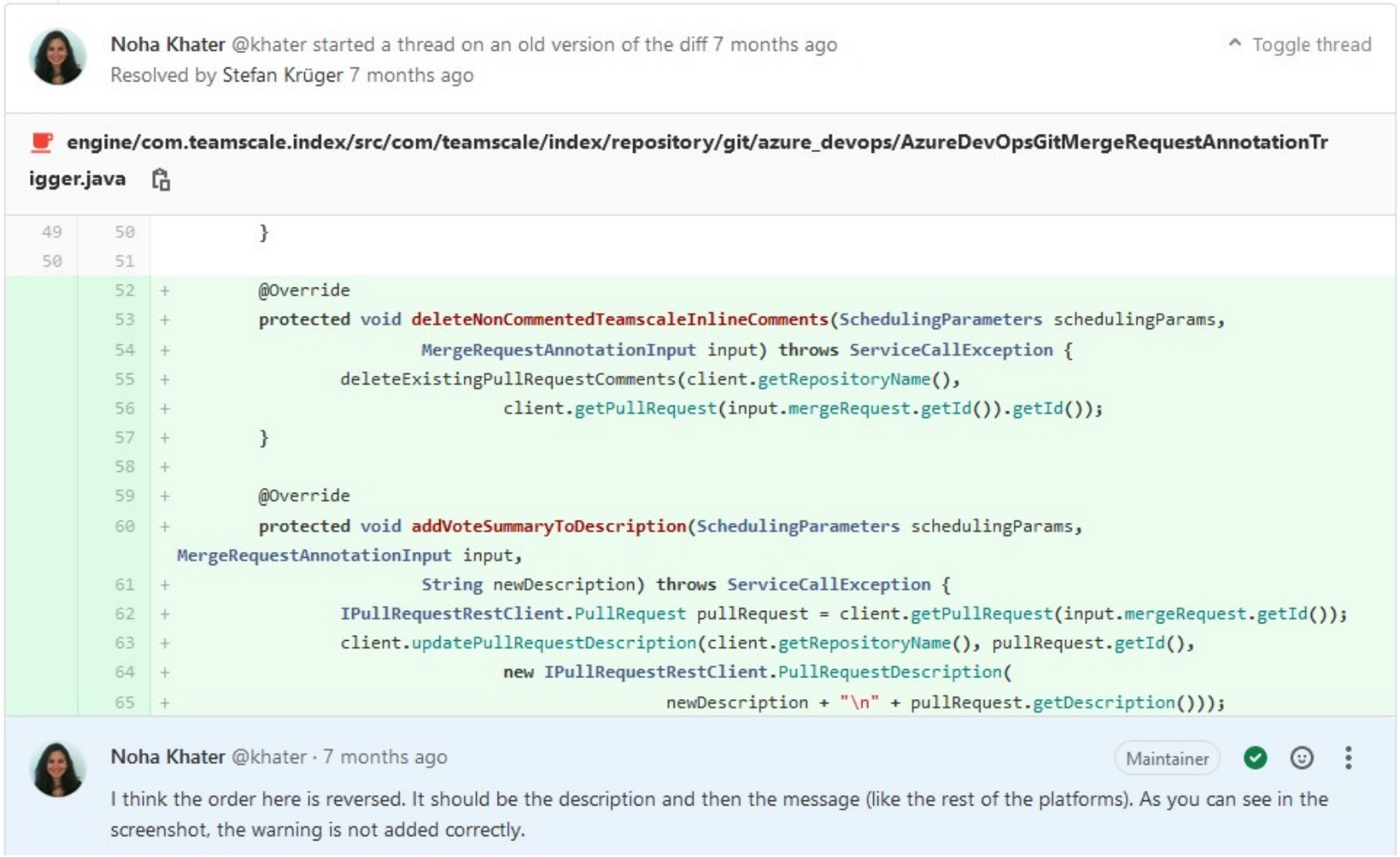
Alternatively, let the extracted method return `Optional<RepositoryChangeEntry>` and do the `add` here.

```
computeChangeEntry(requiredNode, changeSet, path, element).ifPresent(result:add);
```

Maintainability Improvement



Functional Correctness



Noha Khater @khater started a thread on an old version of the diff 7 months ago
Resolved by Stefan Krüger 7 months ago

Toggle thread

engine/com.teamscale.index/src/com/teamscale/index/repository/git/azure_devops/AzureDevOpsGitMergeRequestAnnotationTrigger.java

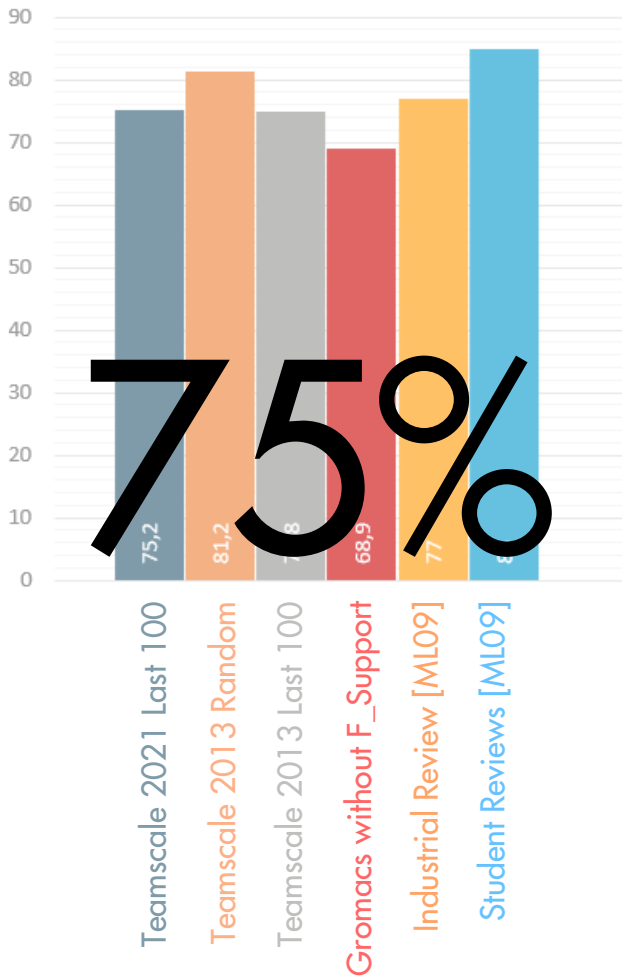
```
49  50          }
50  51
    52  +       @Override
    53  +       protected void deleteNonCommentedTeamscaleInlineComments(SchedulingParameters schedulingParams,
    54  +                       MergeRequestAnnotationInput input) throws ServiceCallException {
    55  +               deleteExistingPullRequestComments(client.getRepositoryName(),
    56  +                               client.getPullRequest(input.mergeRequest.getId()).getId());
    57  +       }
    58  +
    59  +       @Override
    60  +       protected void addVoteSummaryToDescription(SchedulingParameters schedulingParams,
                MergeRequestAnnotationInput input,
    61  +                       String newDescription) throws ServiceCallException {
    62  +               IPullRequestRestClient.PullRequest pullRequest = client.getPullRequest(input.mergeRequest.getId());
    63  +               client.updatePullRequestDescription(client.getRepositoryName(), pullRequest.getId(),
    64  +                               new IPullRequestRestClient.PullRequestDescription(
    65  +                                               newDescription + "\n" + pullRequest.getDescription()));
```

Noha Khater @khater · 7 months ago

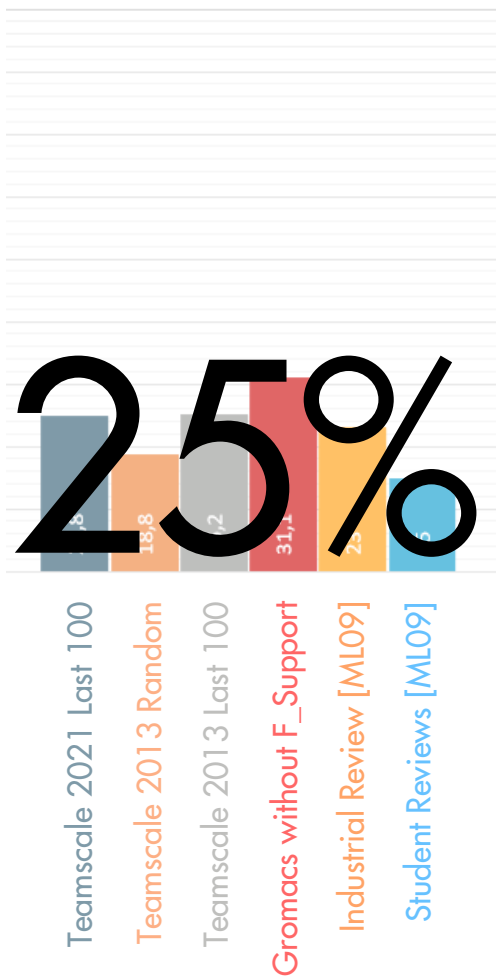
Maintainer

I think the order here is reversed. It should be the description and then the message (like the rest of the platforms). As you can see in the screenshot, the warning is not added correctly.

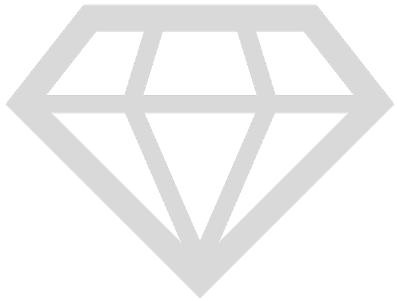
Maintainability



Correctness



Modern Code Reviews in Open-Source Projects: Which Problems Do They Find/Fix?, Beller, Juergens, MSR 2014

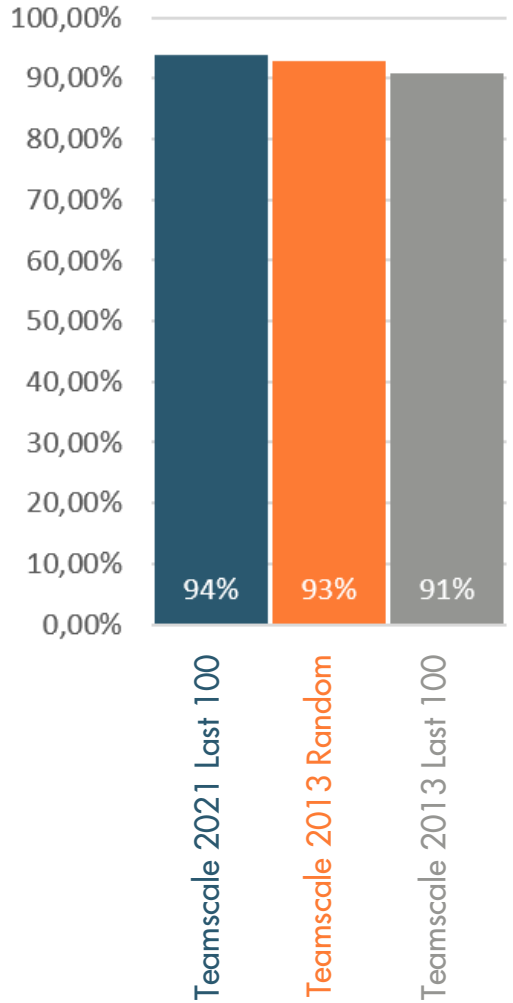


How do code reviews improve our code?

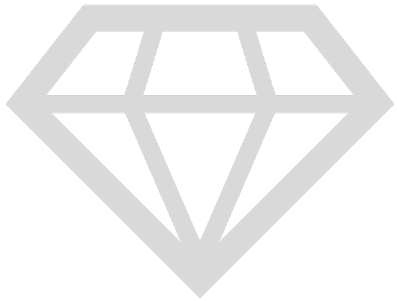
Maintainability 3:1 Correctness

7

Implemented Improvements

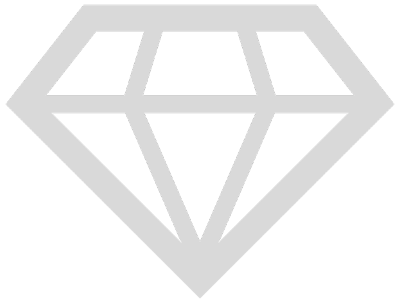


Modern Code Reviews in Open-Source Projects: Which Problems Do They Find/Fix?, Beller, Juergens, MSR 2014



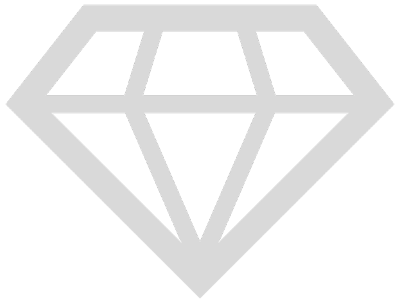
How much do code reviews improve our code?

6.6 improvements per ticket
(1.6 fixes, 5 maintainability)



How much do code reviews improve our code?

1.65 improvements per hour
(0.4 fixes, 1.25 maintainability)

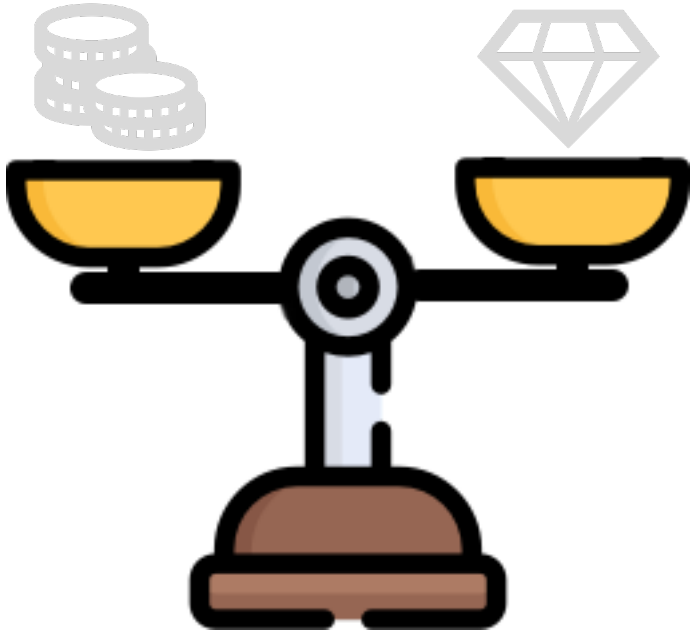


How much do code reviews improve our code?

1.5 improvements per hour
(0.5 fixes, 1 maintainability)

Are Code Reviews worth it?

33%
additional
time invest



½ fix per hour
+ 1 maintainability improvement per hour
+ personal growth
+ knowledge sharing

Let's discuss! ☺

Sven Amann
amann@cqse.eu
+49 172 186 00 63

CQSE GmbH c/o HUB31
Hilpertstr. 31
64295 Darmstadt
www.cqse.eu

Chat with me

tmscl.me/wearedev-sven

CQSE