Meine Erfahrungen aus 10 Jahren Entwicklung von Qualitätsanalysen

Dr. Elmar Juergens

CQSE

jenkins/test/src/main/java/org/jvnet/hudson/test/HudsonTestCase.java   (revision 12d96a56...)

jenkins/test/src/main/java/org/jvnet/hudson/test/JenkinsRule.java   (revision 3909f5ac...)

```java
if (lhs==null && rhs==null)      return;
if (lhs==null)       fail("lhs is null while rhs="+rhs);
if (rhs==null)       fail("rhs is null while lhs="+lhs);

Constructor<?> lc = findDataBoundConstructor(lhs.getClass());
Constructor<?> rc = findDataBoundConstructor(rhs.getClass());
assertEquals("Data bound constructor mismatch. Different type?",lc,rc);

List<String> primitiveProperties = new ArrayList<String>();

String[] names = ClassDescriptor.loadParameterNames(lc);
Class<?>[] types = lc.getParameterTypes();
assertEquals(names.length,types.length);
for (int i=0; i<types.length; i++) {
    Object lv = ReflectionUtils.getPublicProperty(lhs, names[i]);
    Object rv = ReflectionUtils.getPublicProperty(rhs, names[i]);

    if (Iterable.class.isAssignableFrom(types[i])) {
        Iterable lcol = (Iterable) lv;
        Iterable rcol = (Iterable) rv;
        Iterator ltr,rtr;
        for (ltr=lcol.iterator(), rtr=rcol.iterator(); ltr.hasNext() && rtr.hasNext();)
            Object litem = ltr.next();
            Object ritem = rtr.next();

            if (findDataBoundConstructor(litem.getClass())!=null) {
                assertEqualDataBoundBeans(litem,ritem);
            } else {
                assertEquals(litem,ritem);
            }
        }
        assertFalse("collection size mismatch between "+lhs+" and "+rhs, ltr.hasNext() ^
    } else
    if (findDataBoundConstructor(types[i])!=null || (lv!=null && findDataBoundConstructo
        // recurse into nested databound objects
        assertEqualDataBoundBeans(lv,rv);
    } else {
        primitiveProperties.add(names[i]);
    }
}

// compare shallow primitive properties
if (!primitiveProperties.isEmpty())
    assertEqualBeans(lhs,rhs,Util.join(primitiveProperties,","));

*
 Makes sure that two collections are identical via {@link #assertEqualDataBoundBeans(Objec
/
blic void assertEqualDataBoundBeans(List<?> lhs, List<?> rhs) throws Exception {
  assertEquals(lhs.size(), rhs.size());
```

```java
if (lhs==null && rhs==null)      return;
if (lhs==null)       fail("lhs is null while rhs="+rhs);
if (rhs==null)       fail("rhs is null while lhs="+lhs);

Constructor<?> lc = findDataBoundConstructor(lhs.getClass());
Constructor<?> rc = findDataBoundConstructor(rhs.getClass());
assertThat("Data bound constructor mismatch. Different type?", (Constructor)rc, is((Cons

List<String> primitiveProperties = new ArrayList<String>();

String[] names = ClassDescriptor.loadParameterNames(lc);
Class<?>[] types = lc.getParameterTypes();
assertThat(types.length, is(names.length));
for (int i=0; i<types.length; i++) {
    Object lv = ReflectionUtils.getPublicProperty(lhs, names[i]);
    Object rv = ReflectionUtils.getPublicProperty(rhs, names[i]);

    if (lv != null && rv != null && Iterable.class.isAssignableFrom(types[i])) {
        Iterable lcol = (Iterable) lv;
        Iterable rcol = (Iterable) rv;
        Iterator ltr,rtr;
        for (ltr=lcol.iterator(), rtr=rcol.iterator(); ltr.hasNext() && rtr.hasNext();)
            Object litem = ltr.next();
            Object ritem = rtr.next();

            if (findDataBoundConstructor(litem.getClass())!=null) {
                assertEqualDataBoundBeans(litem,ritem);
            } else {
                assertThat(ritem, is(litem));
            }
        }
        assertThat("collection size mismatch between " + lhs + " and " + rhs, ltr.hasNex
                is(false));
    } else
    if (findDataBoundConstructor(types[i])!=null || (lv!=null && findDataBoundConstructo
        // recurse into nested databound objects
        assertEqualDataBoundBeans(lv,rv);
    } else {
        primitiveProperties.add(names[i]);
    }
}

// compare shallow primitive properties
if (!primitiveProperties.isEmpty())
    assertEqualBeans(lhs,rhs,Util.join(primitiveProperties,","));

*
 Makes sure that two collections are identical via {@link #assertEqualDataBoundBeans(Objec
/
blic void assertEqualDataBoundBeans(List<?> lhs, List<?> rhs) throws Exception {
```

- Dashboard
- Activity
- **Findings**
- Metrics
- Tests
- Issues
- Tasks
- Architecture
- Delta
- Projects
- System
- Admin

The variable 'r' may contain a null value at this point and is dereferenced

Code Anomalies > **Null pointer dereference** > Dataflow/Null pointer dereference

The variable that is dereferenced in this statement may contain a null value. This may lead to an exception being thrown at runtime. There are two reasons for this finding: Either the variable has explicitly been assigned a null value at some point before this statement and is now being dereferenced, or the variable is checked for a null value at some point in the

read more

Details

introduced Jan 02 2008 19:09
no diff (introduced with added or copied file)
removed Aug 07 2013 08:32
removal diff   Diff to...

Associated Tasks

None   Add to Task

Workflow

Add to Tolerations

Flag as False Positive

in **core/src/main/java/hudson/util/Service.java:110**

```
 85        public static <T> void load(Class<T> spi, ClassLoader cl, Collection<Class<? extends T>> result) {
 86            try {
 87                Enumeration<URL> e = cl.getResources("META-INF/services/" + spi.getName());
 88                while(e.hasMoreElements()) {
 89                    BufferedReader r = null;
 90                    URL url = e.nextElement();
 91                    try {
 92                        r = new BufferedReader(new InputStreamReader(url.openStream(),"UTF-8"));
 93                        String line;
 94                        while((line=r.readLine())!=null) {
 95                            if(line.startsWith("#"))
 96                                continue;   // comment line
 97                            line = line.trim();
 98                            if(line.length()==0)
 99                                continue;   // empty line. ignore.
100
101                            try {
102                                result.add(cl.loadClass(line).asSubclass(spi));
103                            } catch (ClassNotFoundException x) {
104                                LOGGER.log(Level.WARNING, "Failed to load "+line, x);
105                            }
106                        }
107                    } catch (IOException x) {
108                        LOGGER.log(Level.WARNING, "Failed to load "+url, x);
109                    } finally {
110                        r.close();
111                    }
112                }
113            } catch (IOException x) {
114                LOGGER.log(Level.WARNING, "Failed to look up service providers for "+spi, x);
115            }
116        }
117
```
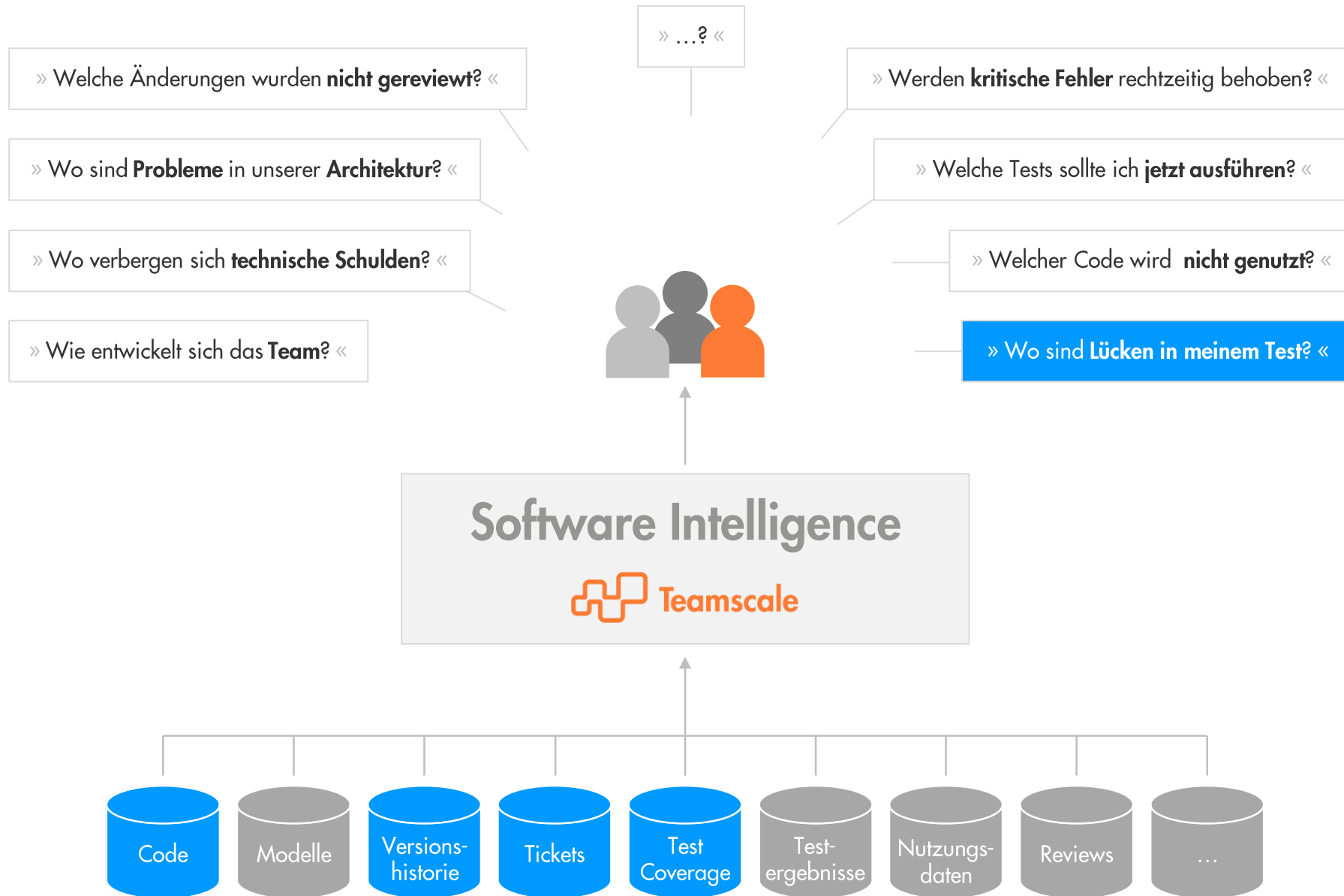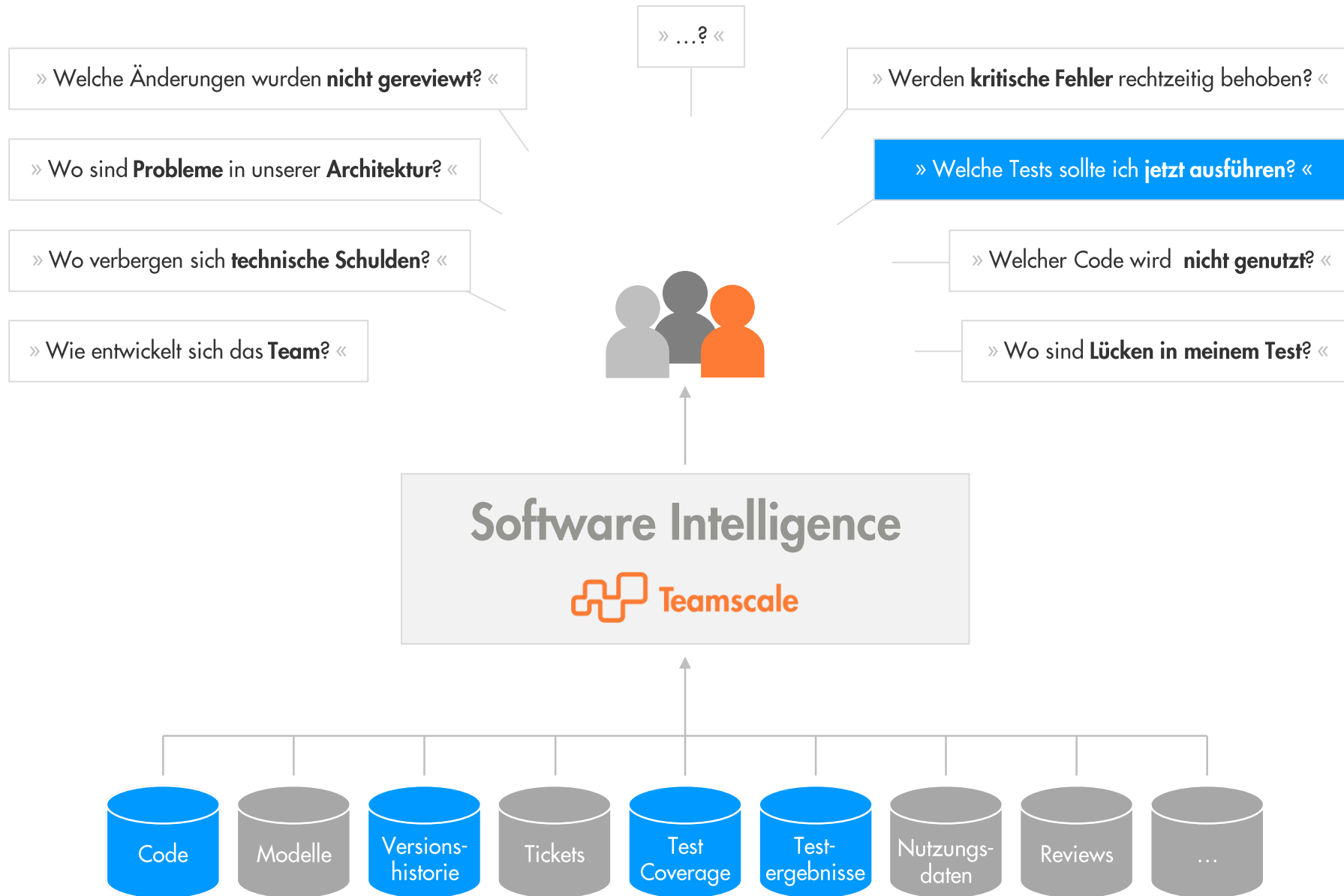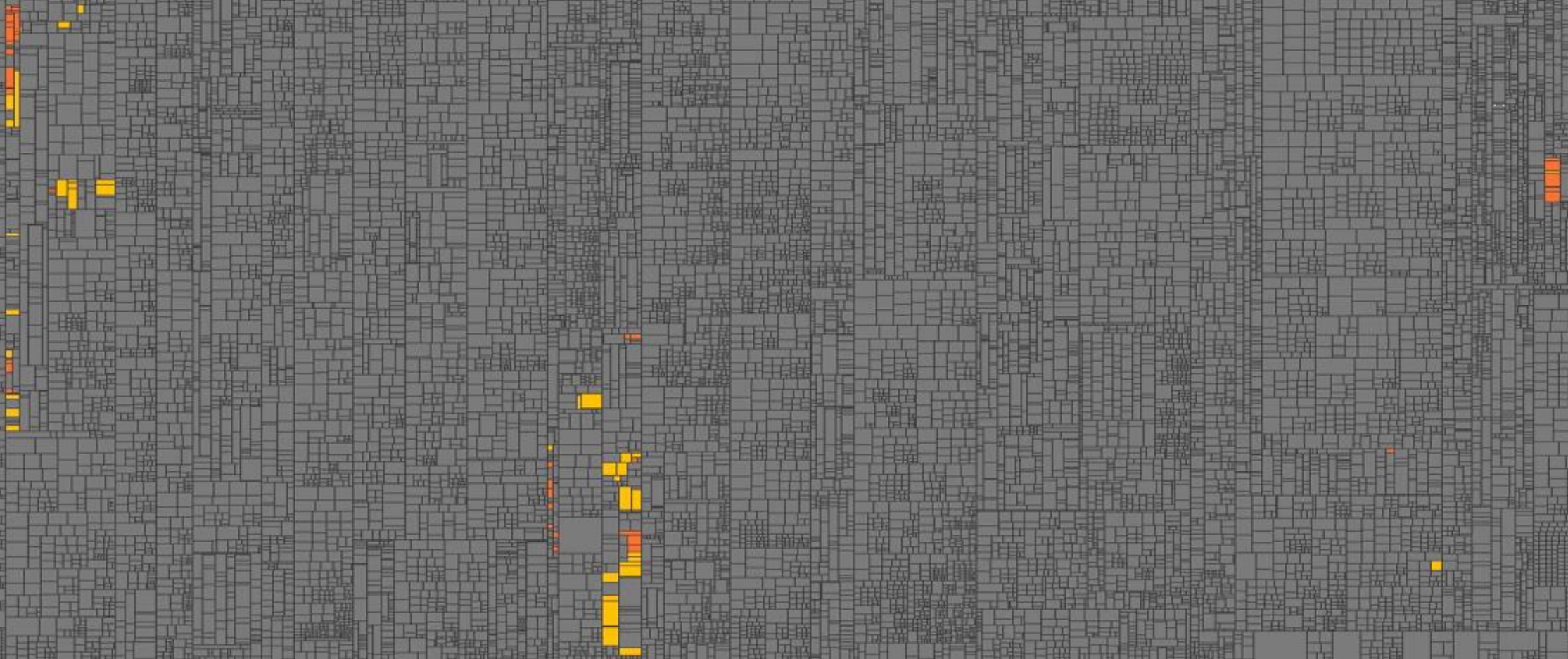
● = Modifiziert & ungetestet
● = Neu & ungetestet
● = Unverändert
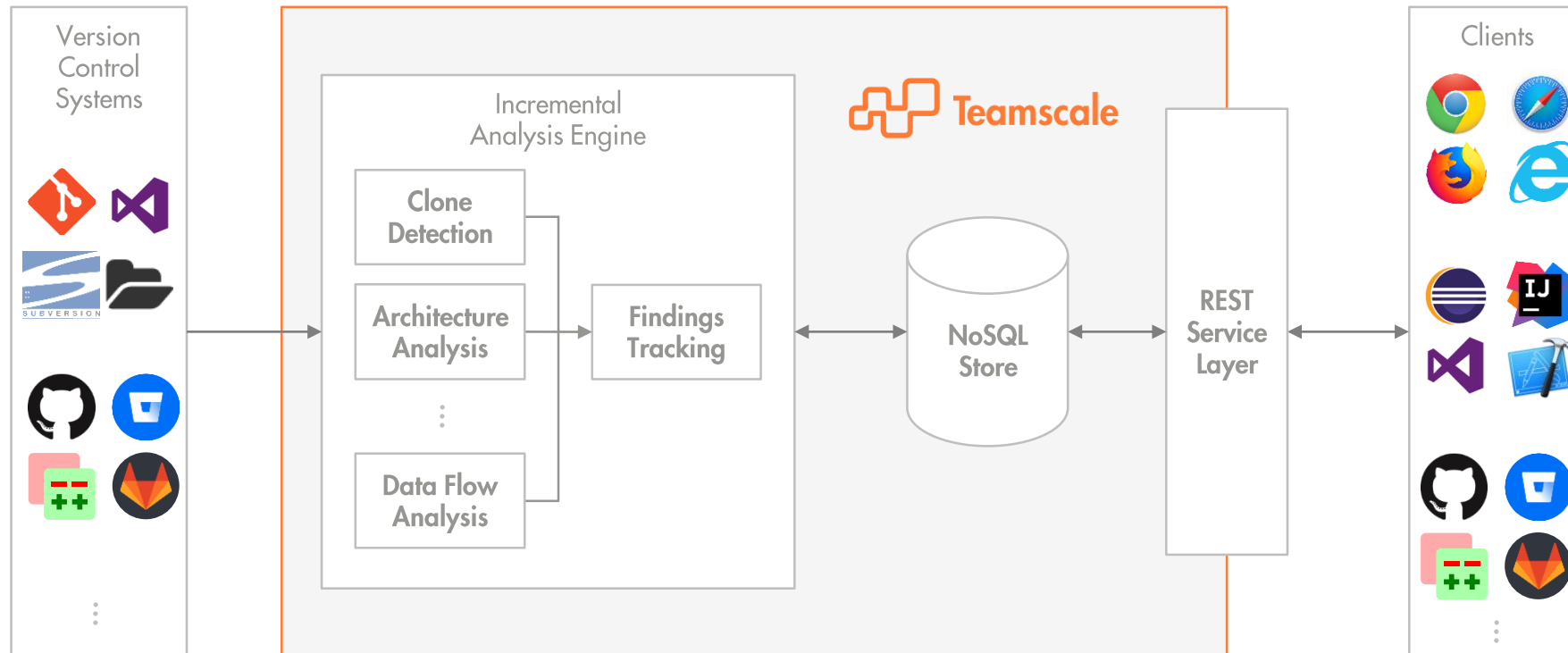● = Geändert & ausgeführt im Test
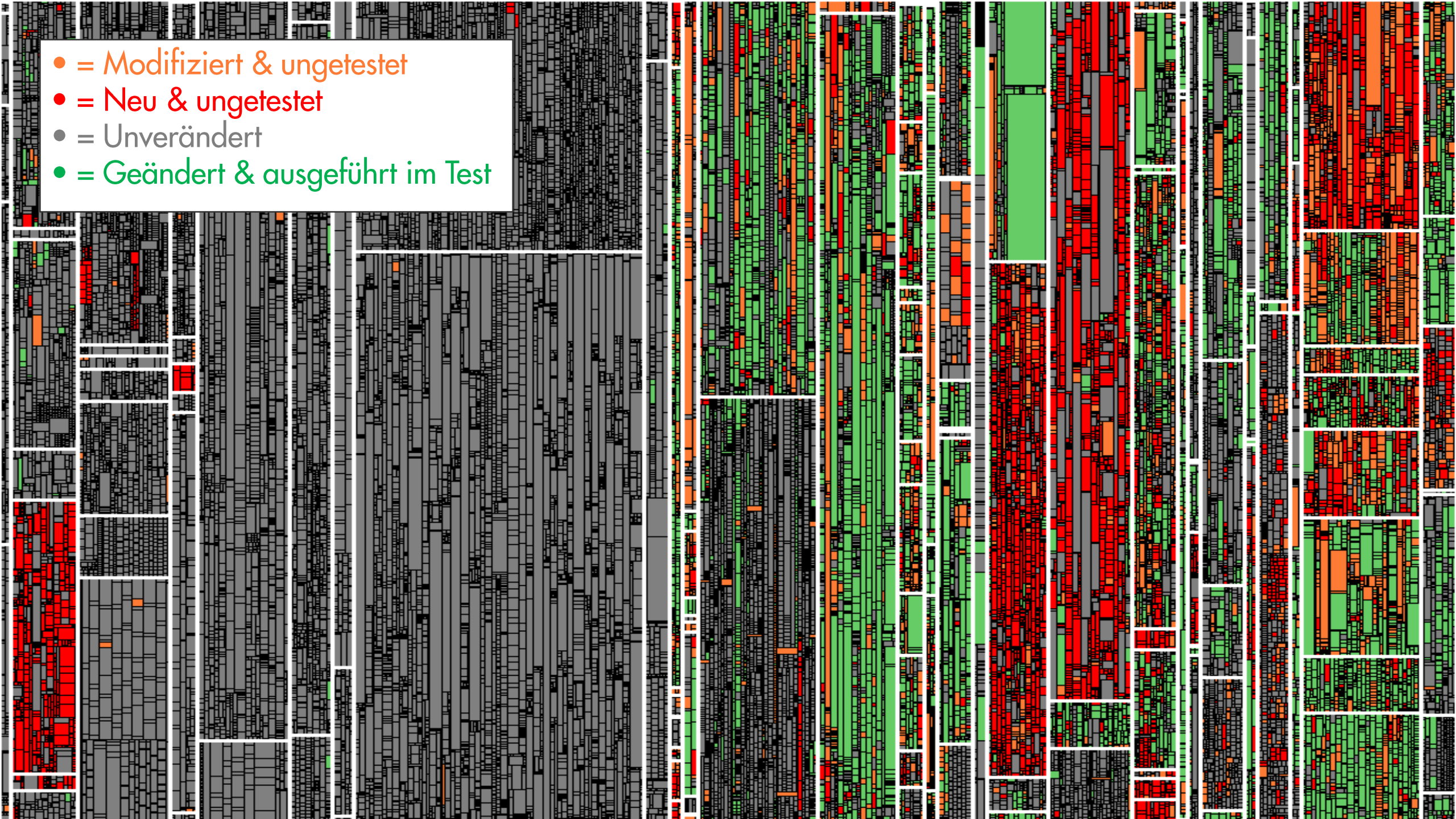
Issue # ⌄	Subject			Test Gap
↗ TS-10549	Undo/Redo for web-based architecture editor	Done		0%
↗ TS-10784	Fix long method finding in TaintAnalysisRunner	Done		0%
↗ TS-10923	Implement metric 'Nesting Depth' for Simulink	Done		29%
↗ TS-11364	External findings are not registered during first upload	Done		14%
↗ TS-11942	Manual test coverage upload during development	Done		43%
↗ TS-12050	Tool for transferring findings blacklists and tasks	Done		50%
↗ TS-12262	Cannot set or alter alias without reanalysis	Done		0%
↗ TS-13151	Fetch parent relationship of TFS work items	Done		0%

| Version Control Systems | Incremental Analysis Engine / Teamscale | NoSQL Store | REST Service Layer | Clients |

ABAP	Groovy	Objective-C	Simulink/StateFlow
Ada	Gosu	Open CL	SQLScript
C#	IEC 61131-3 ST	OScript	Swift
C/C++	Java	PHP	TypeScript
Cobol	JavaScript	PL/SQL	Visual Basic .NET
Delphi	Kotlin	Python	Xtend
Fortran	Matlab	Rust	…
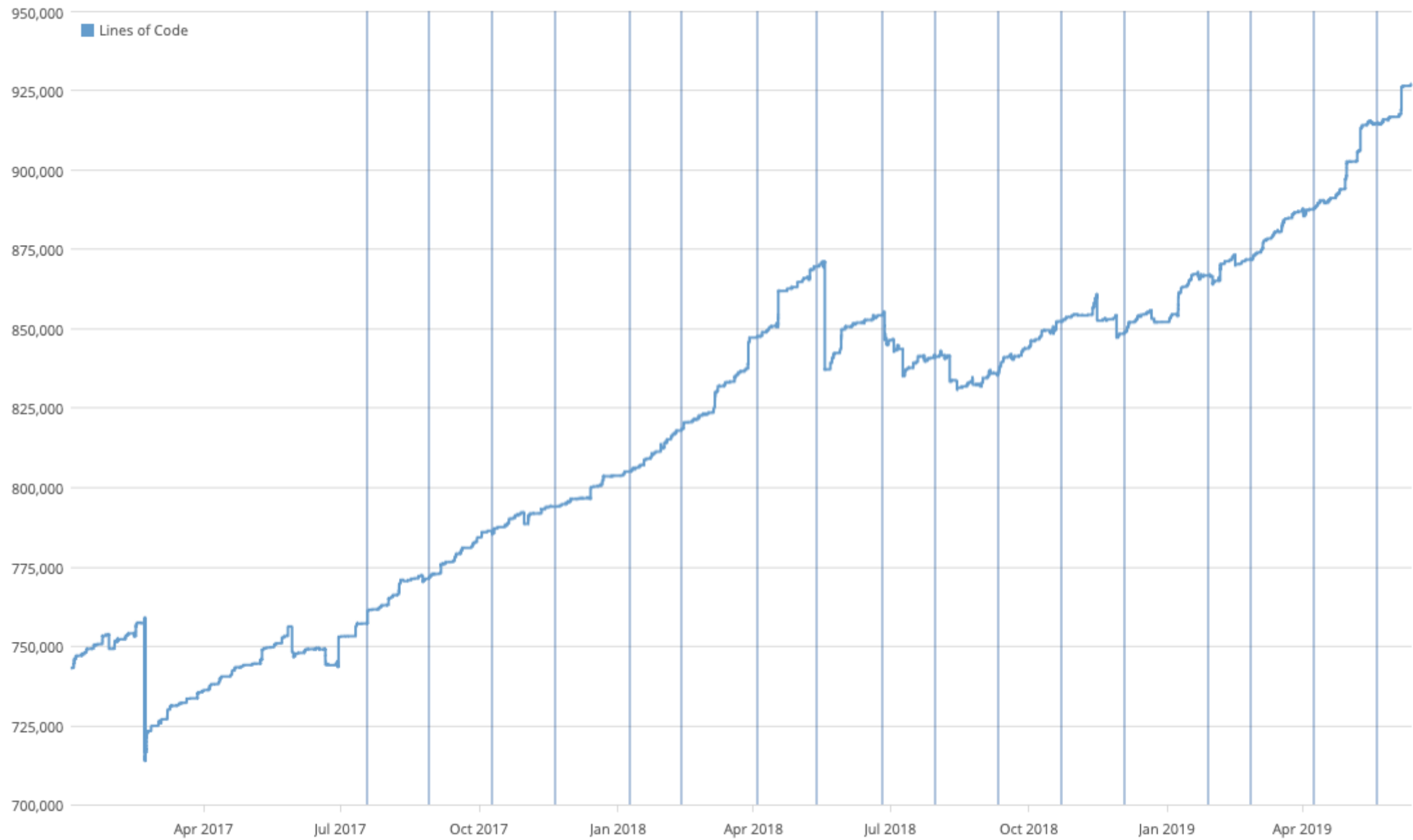
Releases

- 6-wöchentliche Major-Releases
- Wöchentlich 1-2 Patch-Releases
- 42 Releases in 2019 (Ende Mai)

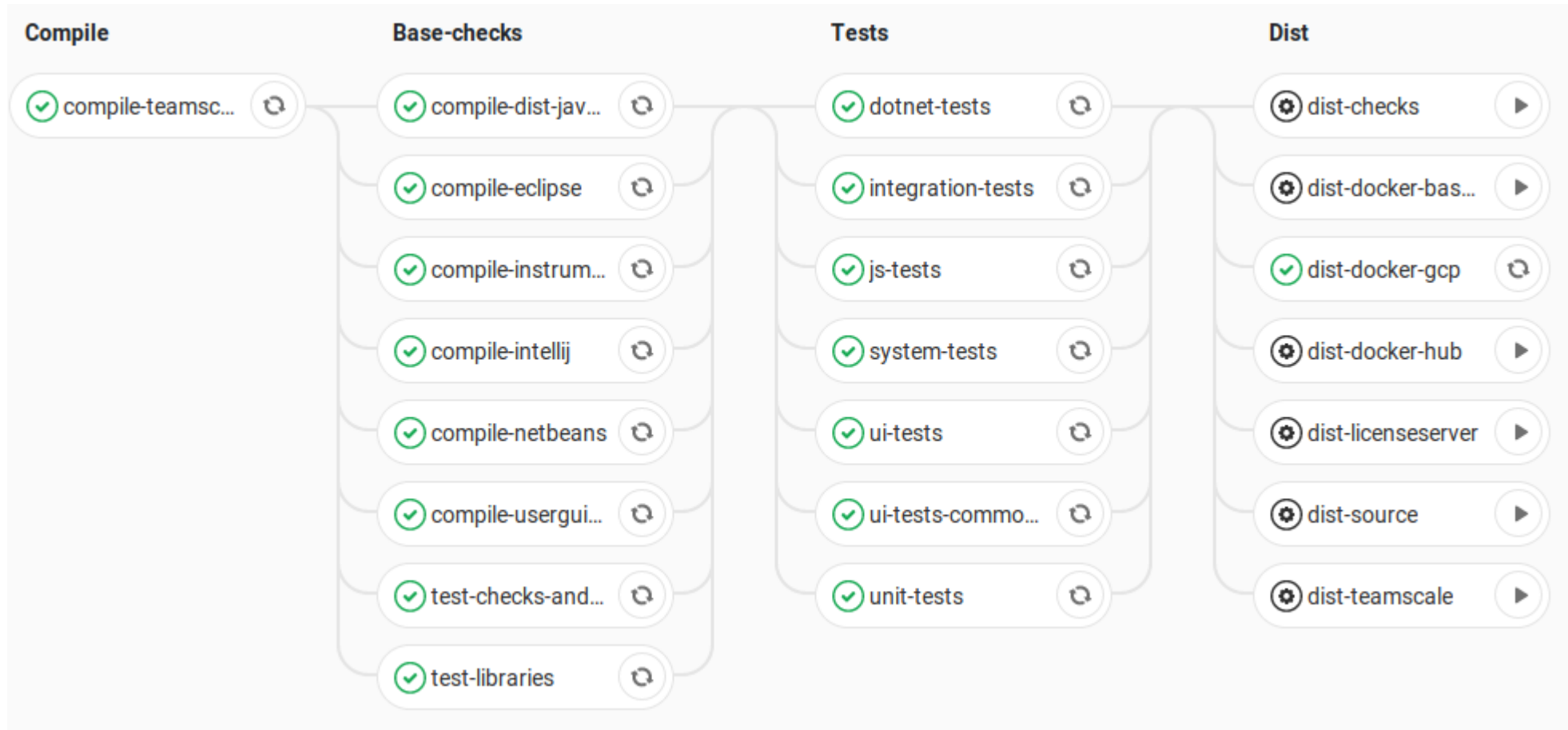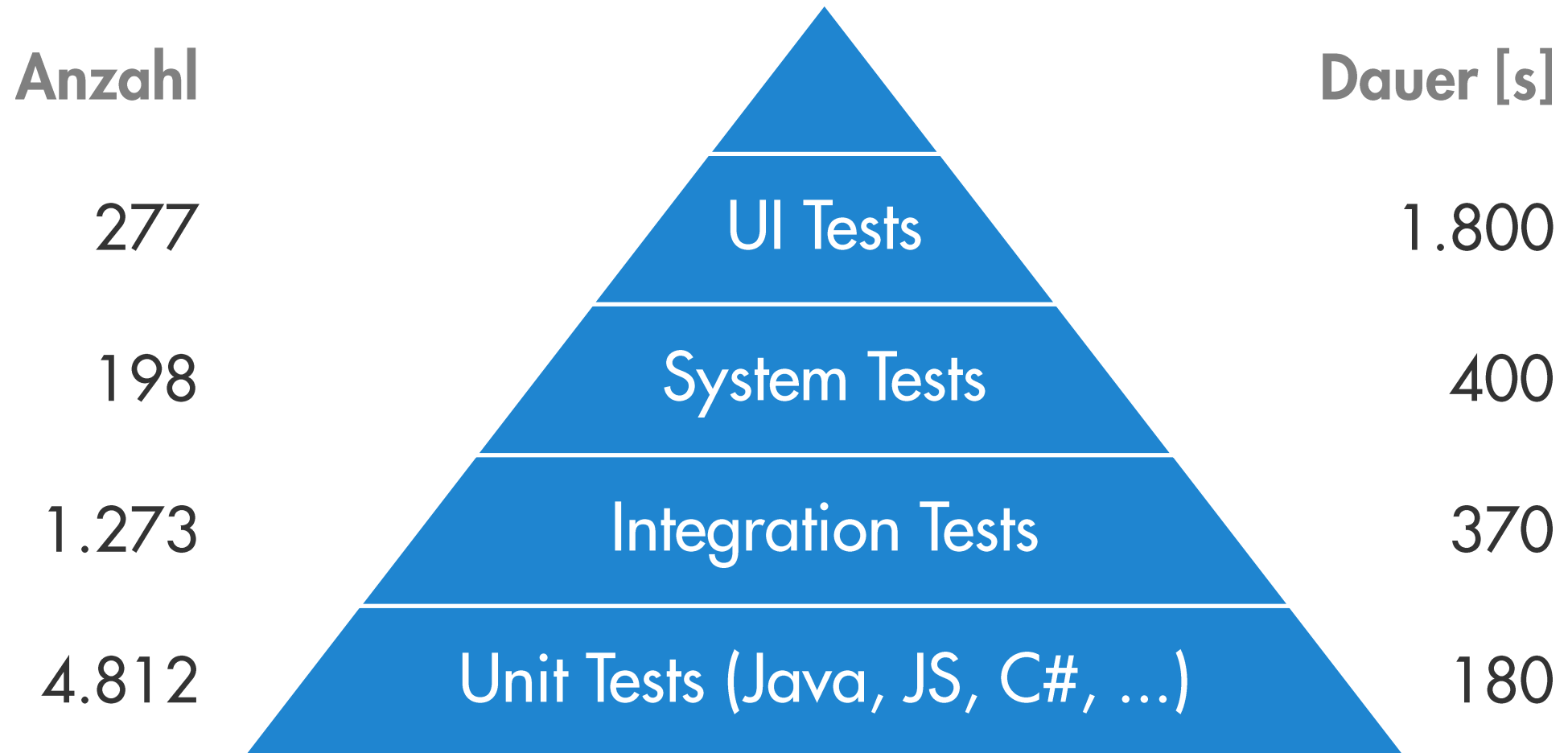| Compile	Base-checks	Tests	Dist
compile-teamsc...	compile-dist-jav...	dotnet-tests	dist-checks
	compile-eclipse	integration-tests	dist-docker-bas...
	compile-instrum...	js-tests	dist-docker-gcp
	compile-intellij	system-tests	dist-docker-hub
	compile-netbeans	ui-tests	dist-licenseserver
	compile-usergui...	ui-tests-commo...	dist-source
	test-checks-and...	unit-tests	dist-teamscale
	test-libraries		

~ 2.000 Pipelines in 6 Wochen
~ 40.000 Jobs (inkl. skipped)

Bis zu 70 parallele Maschinen

Im Mai:
- 19.336 CPU-Stunden
- 72.512 GB-Stunden RAM

~ 500 Euro nur für Build-Ausführung

Wie wir Teamscale testen

Anzahl		Dauer [s]
277	UI Tests	1.800
198	System Tests	400
1.273	Integration Tests	370
4.812	Unit Tests (Java, JS, C#, …)	180

Done **Issue TS-15286** - Allow upload of backups to S3 storage

Creator:	Andreas Sewe (on May 23 2018 16:00)			Last update: May 29 2018 15:54	
Assignee:	Andreas Sewe			Parent: Issue TS-14733	

project	Type	Priority	Resolution	Fix Version	Component
TS	Feature	Normal	Green	Teamscale 4.3	Backend
Affected Version	**Customer**	**Customer Issue**	**Epic Name**	**Freshdesk URL**	

Merge Request		**PDash Task**	**QA-Contact**	
https://git.cqse.eu/cqse/teamscale/merge_requests/2963/			kinnen	

Description

This is part of TS-14733. Initial goal for 4.3 (as the feature freeze is soon) is to support special {{s3:}} URI as *Path to backup.zip*.
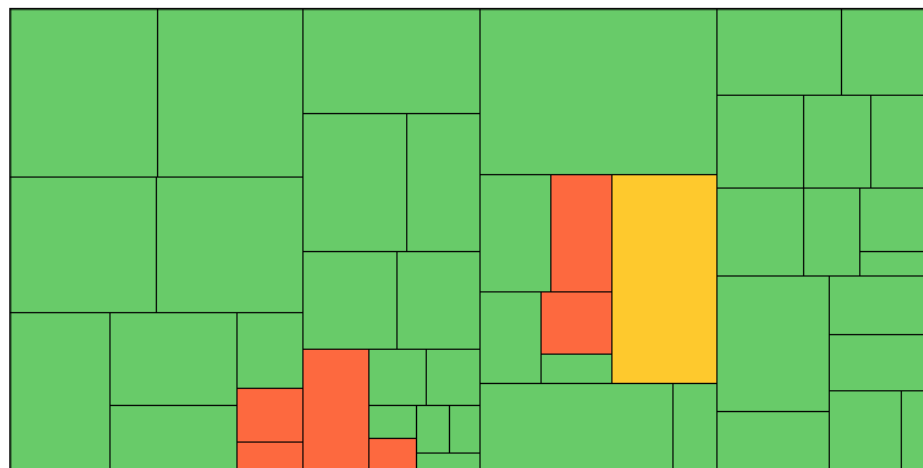
Credentials can be stored either in the URI or (if possible) under *Admin > Settings > System Accounts*.

⌄ Affected files (15)

⌃ Test Gap Treemap

May 28 2018 13:40–now | Test Gap: 14.89%   Coverage sources: **All**

Associated repository changes (7)

Merge branch 'cr/15286_allow_backups_to_s3' into 'master'   May 29 2018
CR#15286 Allow upl...   15:54
by Thomas Kinnen in revision 23d333c8 📋 in branch master (Gitlab)
Merged from cr/15286_allow_backups_to_s3 0 2
Files: 7 added, 3 changed, 1 deleted
Findings: 0 4

Merged   Opened 2 weeks ago by   Andreas Sewe 8 of 8 tasks completed
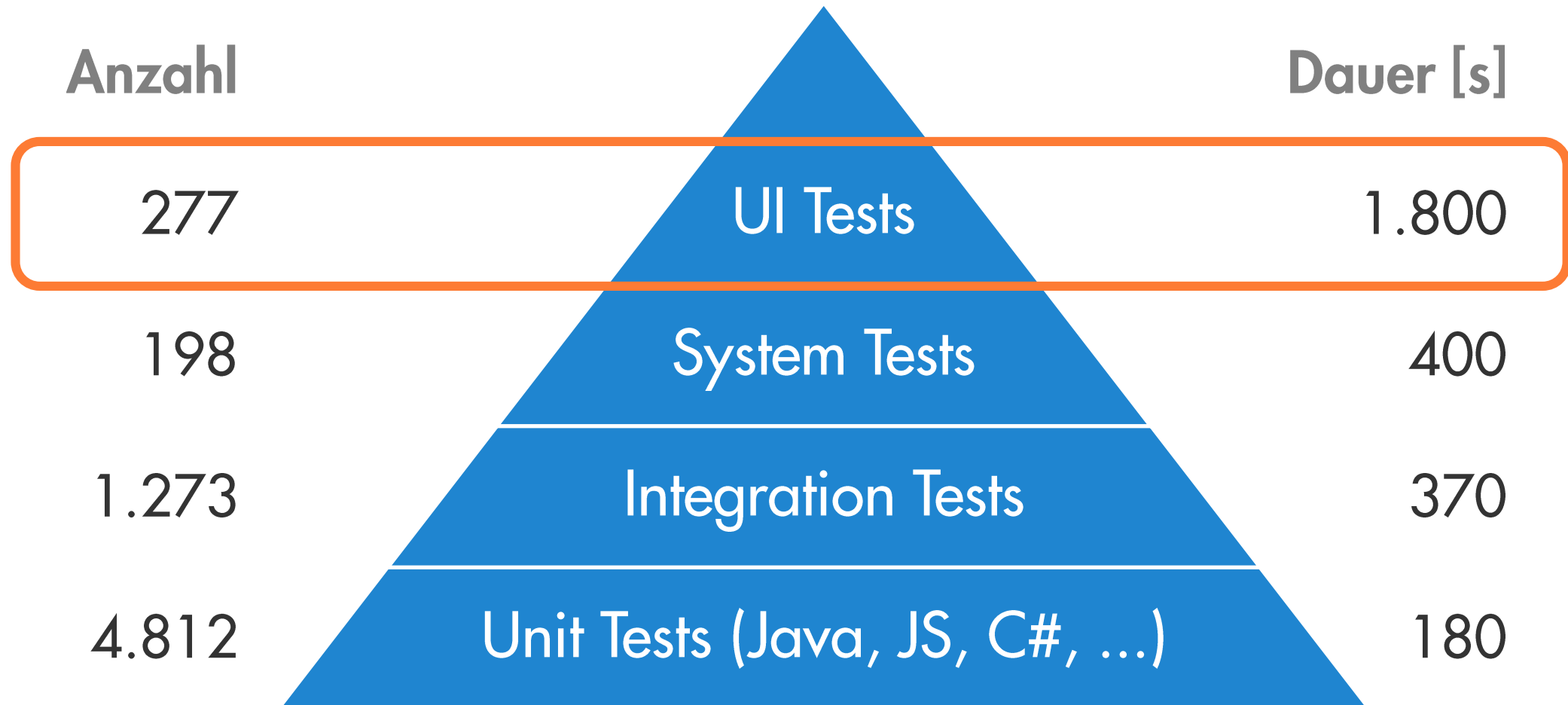
Edit   Close merge request

CR#15286 Allow upload of backups to S3

- ☑ Code adheres to our Coding Guidelines and produces no errors or warnings in the IDE
- ☑ The change is verifiably tested (no non-trivial test gaps)
- ☑ No new Teamscale findings reported for the code changes (old findings in changed code are addressed as negotiated with reviewer)
- ☑ In case of user-visible changes or new features, the Teamscale product documentation (Admin Guide, User Guide, Tutorials) is updated accordingly
- ☑ CHANGELOG (at the root of the project) is updated in case of prominent features or important bug fixes
- ☑ Migrations written for all changes to data in backed up indices and migration tested at least once
- ☑ Migrations written for all changes to dashboard widgets/parameters and migration tested at least once
- ☑ The Feature Specific Review Checklists have been considered

⚠ 0   ✎ 2   ☑ 455

Edited 2 weeks ago by Andreas Sewe

Wie wir Teamscale testen

Anzahl		Dauer [s]
277	UI Tests	1.800
198	System Tests	400
1.273	Integration Tests	370
4.812	Unit Tests (Java, JS, C#, …)	180

Pipeline	Id ▾	Name	Commit	Status	Runtime	Tests executed	Time saved	TIA enabled	Runtime distribution
83265	694608	ui-tests-common-lang	master (May 8, 2019 10:54:36 AM)	SUCCESS	20m 38s	100% -> master always runs all tests	00m 30s	Yes	
83265	694607	ui-tests	master (May 8, 2019 10:54:36 AM)	SUCCESS	26m 09s	100% -> master always runs all tests	01m 58s	Yes	

Pipeline	Id ▾	Name	Commit	Status	Runtime	Tests executed	Time saved	TIA enabled	Runtime distribution
83239	694073	ui-tests-common-lang	cr/18778_add_tasks_slide (May 7, 2019 7:03:06 PM)	SUCCESS	06m 27s	16%	14m 41s	Yes	
83239	694072	ui-tests	cr/18778_add_tasks_slide (May 7, 2019 7:03:06 PM)	SUCCESS	05m 42s	08%	22m 25s	Yes	
83237	694027	ui-tests-common-lang	cr/18985_apply_patterns_when_toggle_on (May 7, 2019 6:37:16 PM)	SUCCESS	12m 44s	47%	08m 24s	Yes	
83237	694026	ui-tests	cr/18985_apply_patterns_when_toggle_on (May 7, 2019 6:37:16 PM)	SUCCESS	16m 14s	44%	11m 53s	Yes	
83234	693958	ui-tests-common-lang	cr/18089_initial_reports_implementation (May 7, 2019 5:15:30 PM)	SUCCESS	05m 01s	01%	16m 07s	Yes	
83234	693957	ui-tests	cr/18089_initial_reports_implementation (May 7, 2019 5:15:30 PM)	SUCCESS	05m 01s	00%	23m 06s	Yes	

83059	690294	ui-tests-common-lang	cr/18089_initial_reports_implementation (May 2, 2019 10:00:07 PM)	FAILED	10m 06s	29%	11m 02s	Yes	

2009

2019

CQSE History

2009   2010   2011   2012   2013   2014   2015   2016   2017   2018   2019

10 MA   20 MA   30 MA
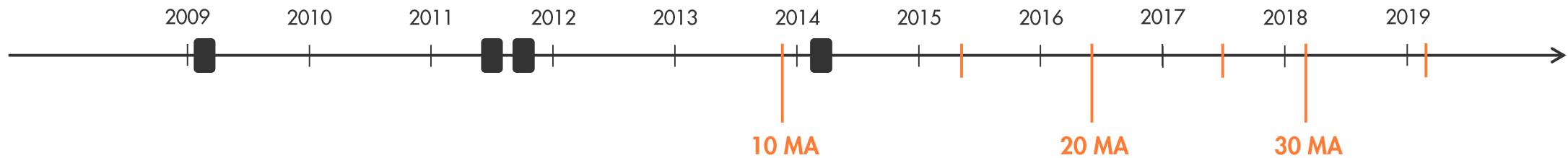
- Jan 2009: Gründung
- Feb 2009: Erste Beratungskunden

- Mid 2011: Promotionen abgeschlossen, Fokus auf CQSE
- Sept 2011: Erstes Büro, erster Mitarbeiter

- Jan 2014: Release Teamscale 1.0

Offices

Home-offices

Bremen

Ratingen

Fulda

Darmstadt

Landshut

München
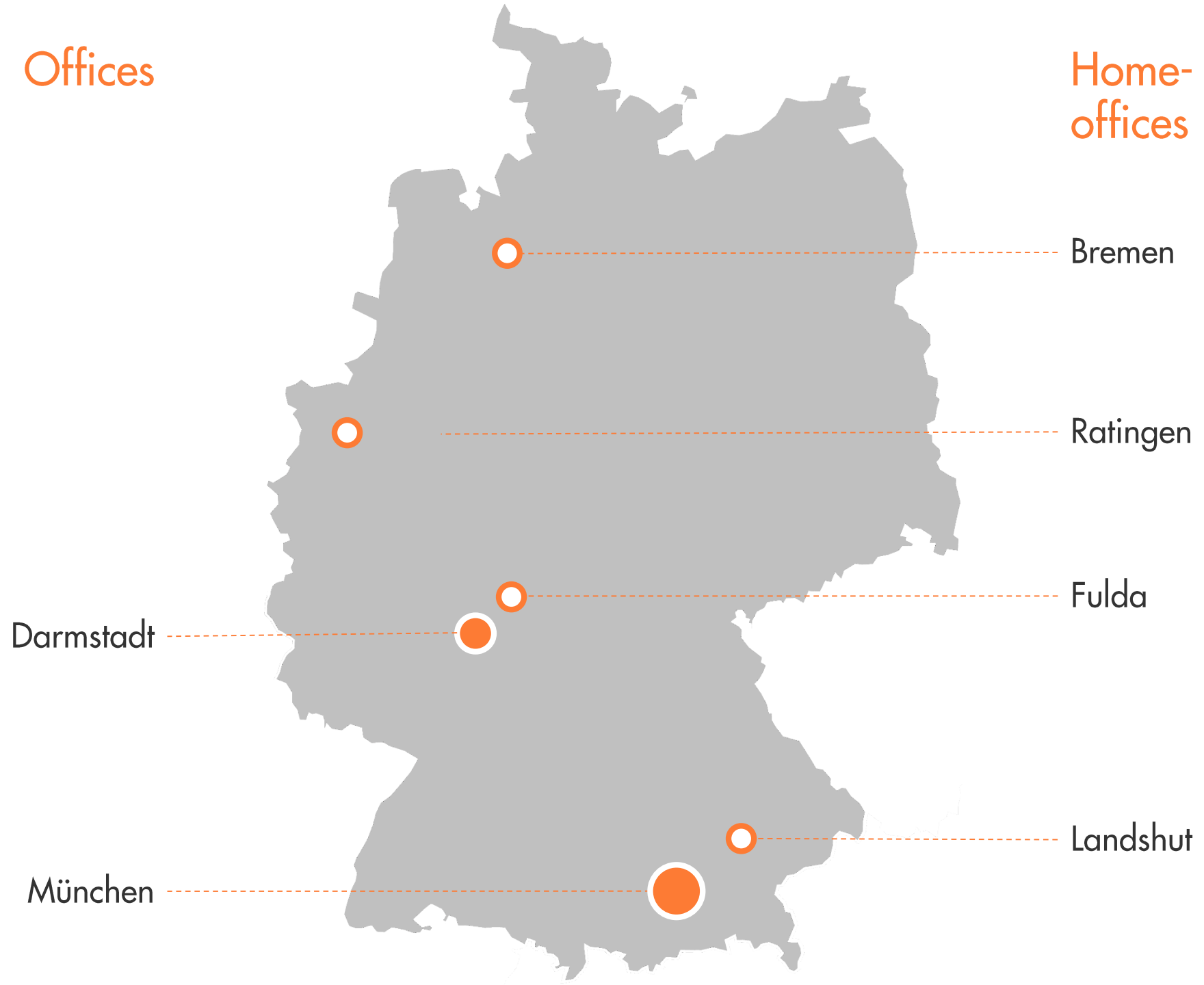
2009

2019

Code Peer Reviews

Release
Version 1.0

Version 1.0

Critical
Bug-Fixing

Version 1.1

Entwicklung

Version 2.0

Clean Code Days
2014

2009   2010   2011   2012   2013   2014   2015   2016   2017   2018   2019

10 MA   20 MA   30 MA

Treemap for Review Rating

Version 1.0

Critical
Bug-Fixing

Version 1.1

Entwicklung

Version 2.0

Deadline for 2.0

Feature B

Bug-Fix A

Bug-Fix B

1.0

1.1

Feature A

Feature C

Feature D
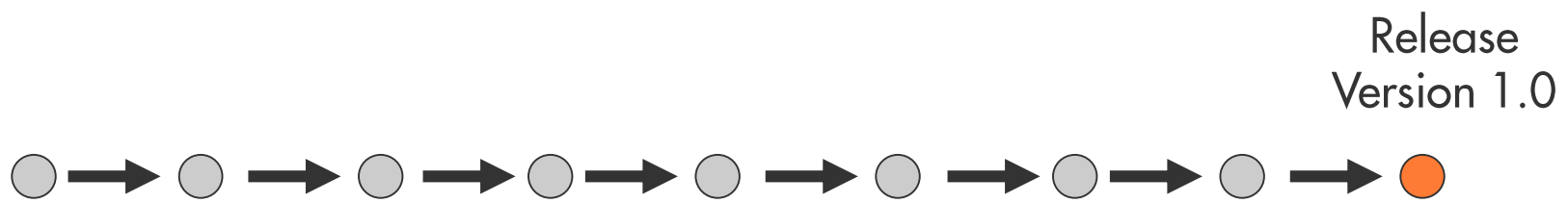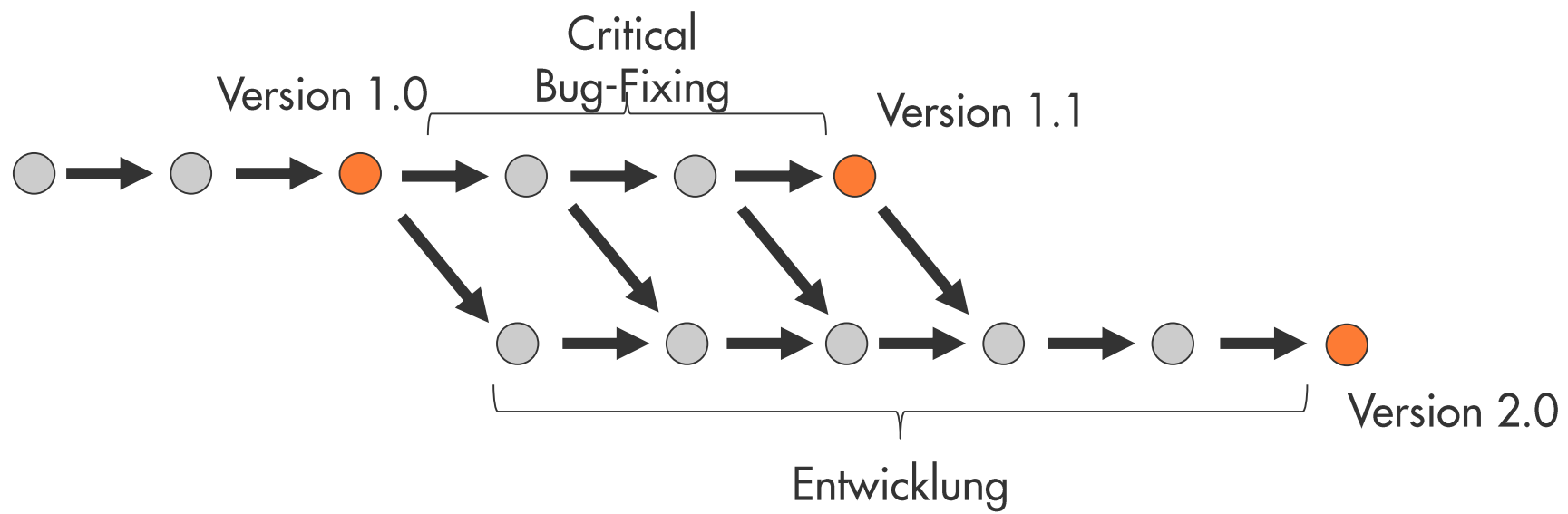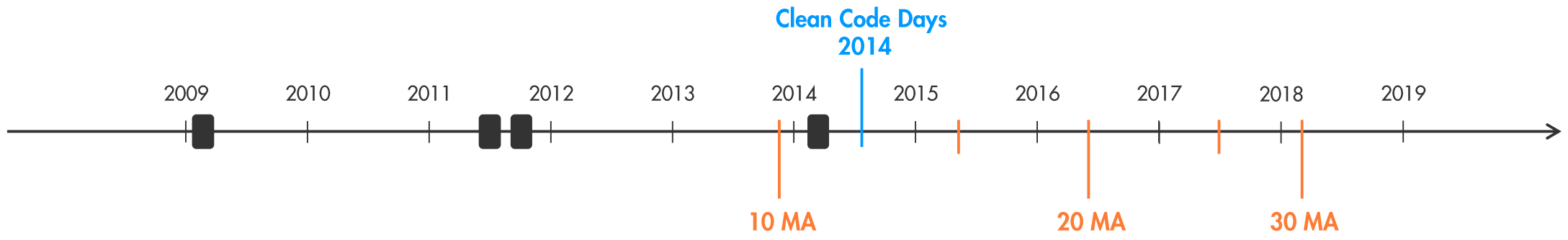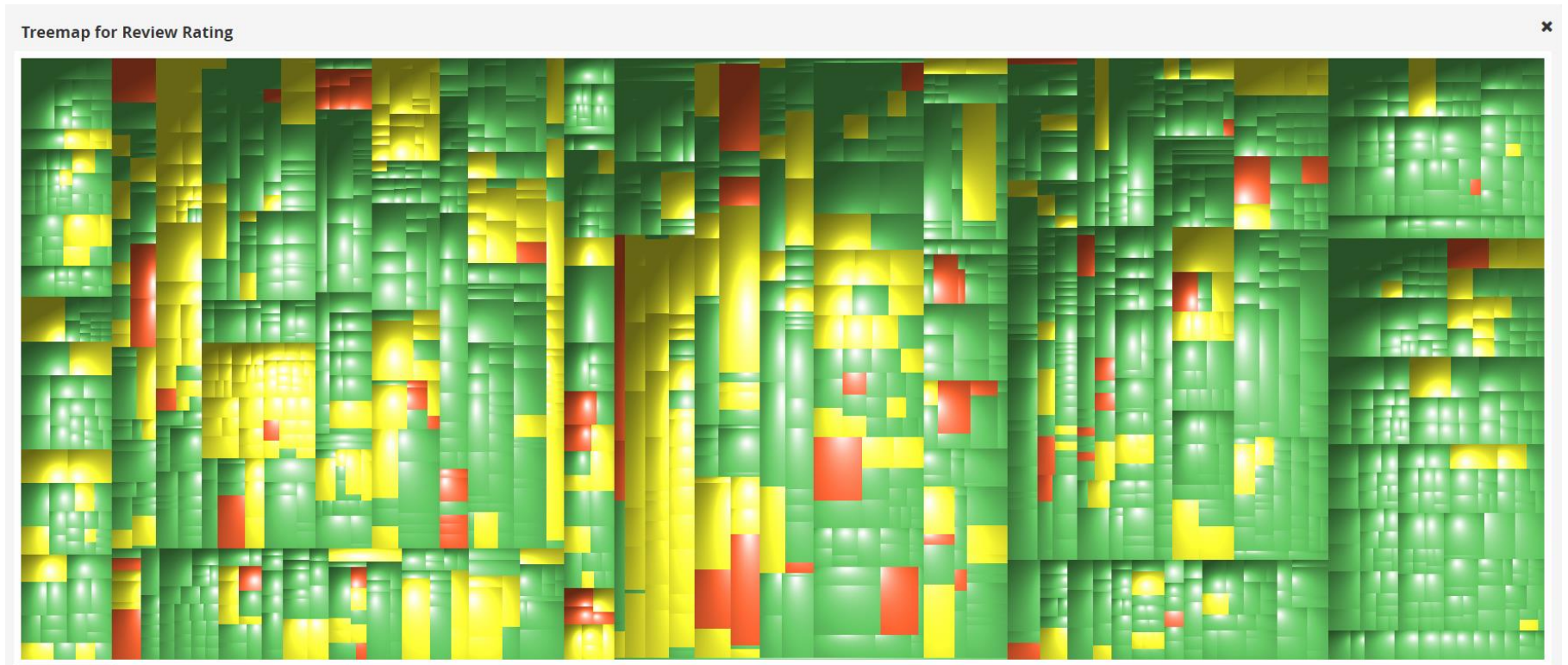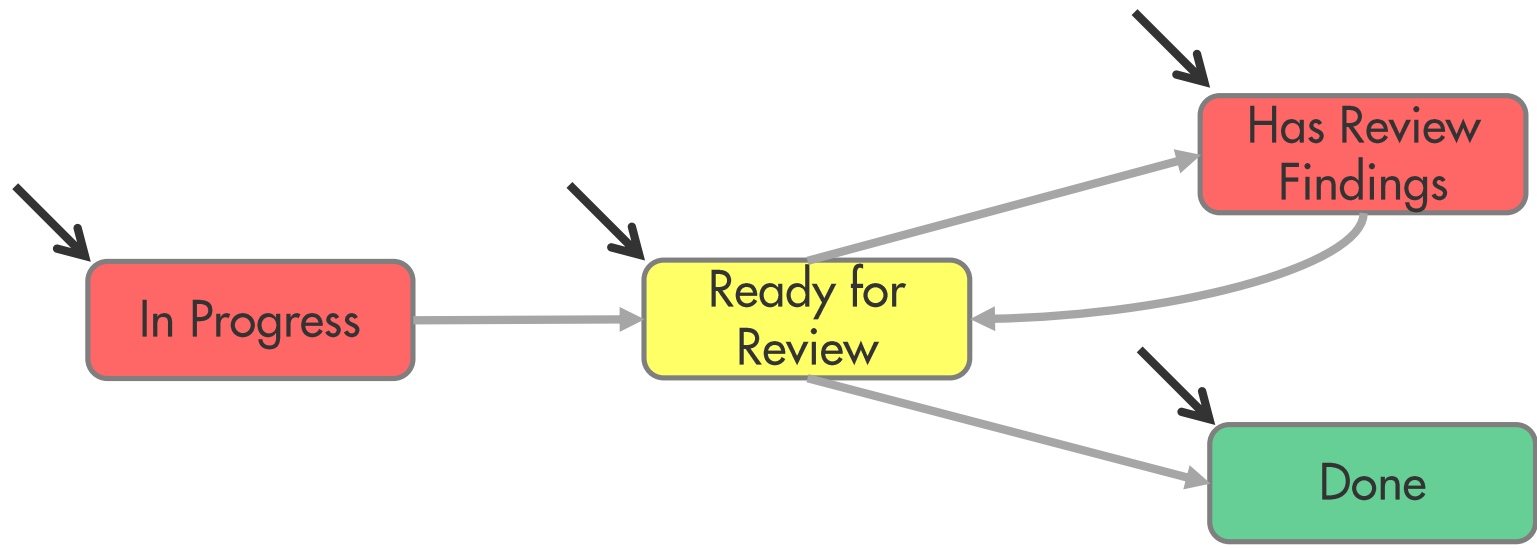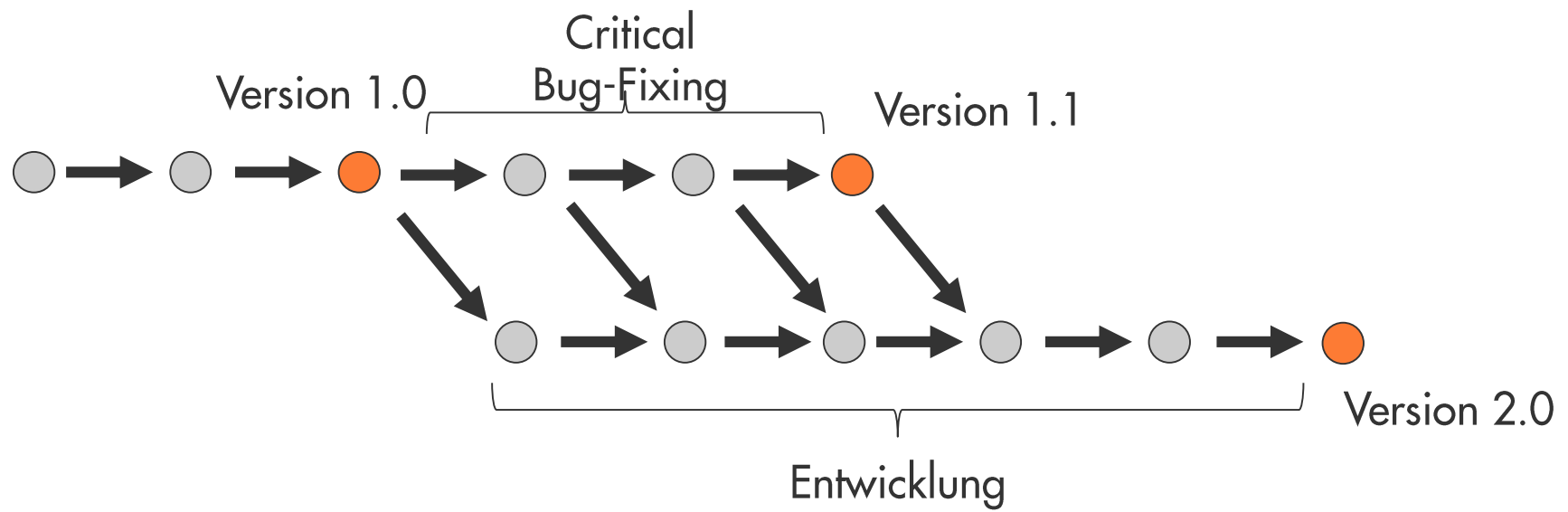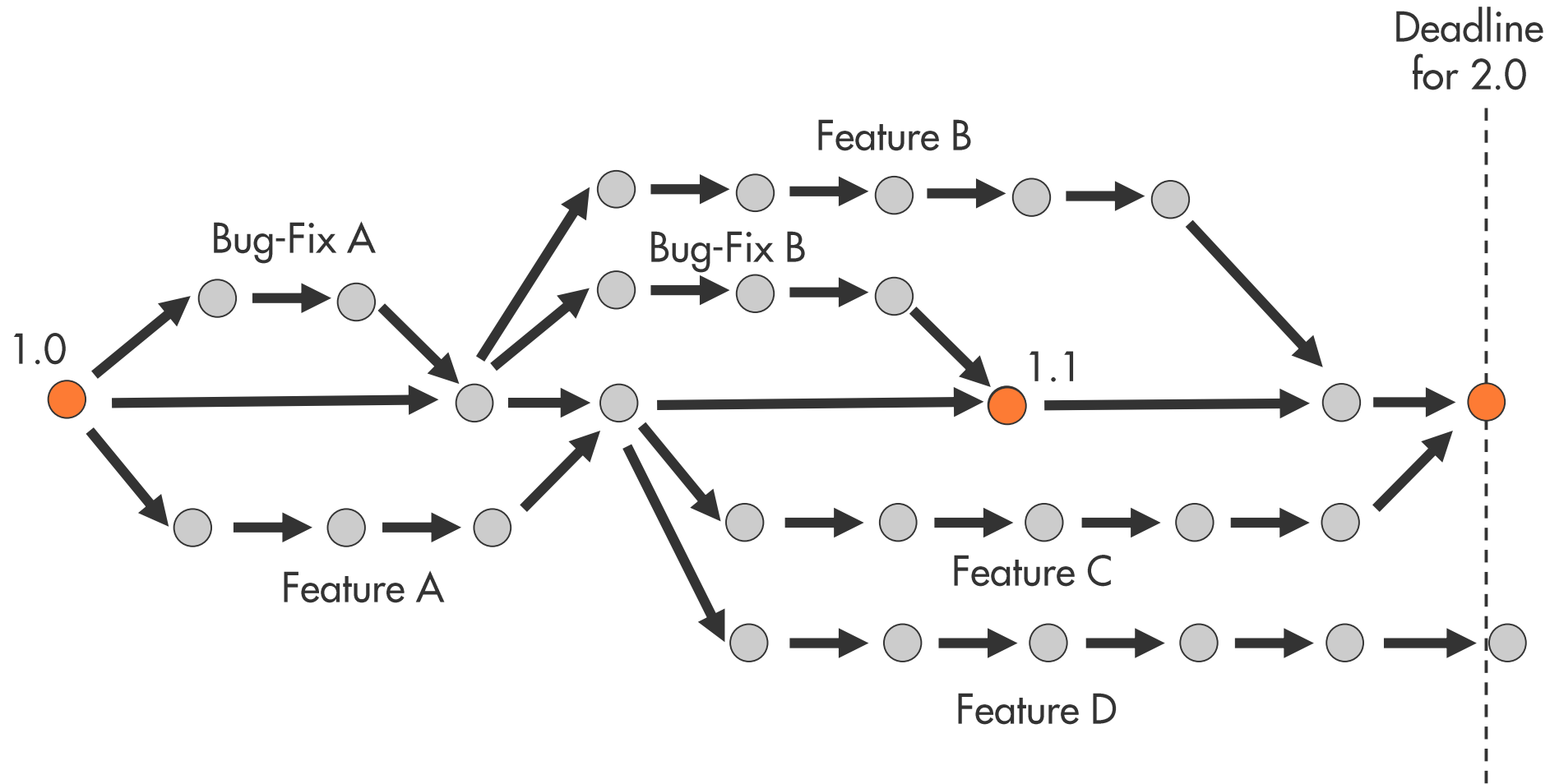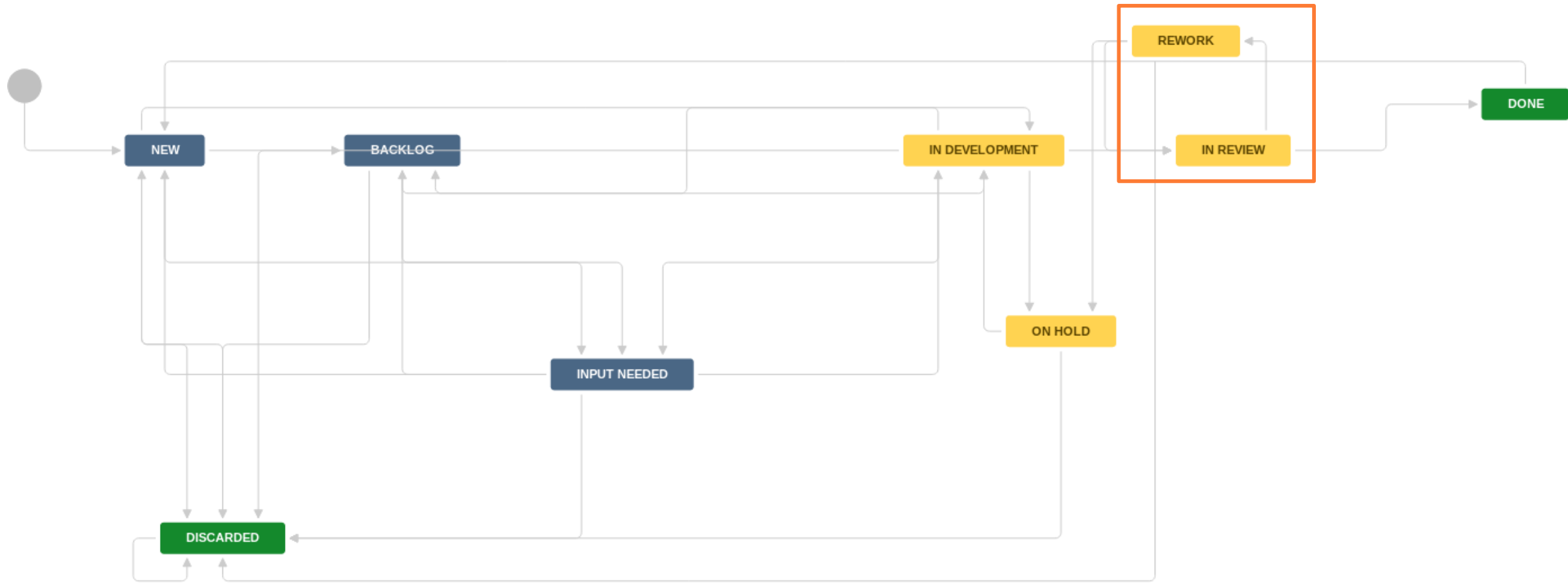
Deep nesting in RepositoryContentUpdaterBase

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			

⌄ 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

› Description

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

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
293  293                        List<BasicTokenElementInfo> tokenElements = tokenElementIndex.getValues(paths);
294  294                        for (int i = 0; i < paths.size(); i++) {
295  295                            String path = paths.get(i);
     296  -                        RepositoryChangeInfo changeInfo = changeSet.getChangeInfo(path);
     297  -                        ERepositoryChangeType changeType = patchChangeType(changeInfo, tokenElements.get(i));
     298  -                        if (changeType != null) {
     299  -                            result.add(new RepositoryChangeEntry(path, requiredNode.getRevision().getRevision(), changeType,
     300  -                                    commit, changeInfo.getOriginPath(), changeInfo.getOriginCommit()));
     301  -                        }
     296  +                        BasicTokenElementInfo element = tokenElements.get(i);
     297  +                        addChangeEntry(requiredNode, result, changeSet, path, element);
302  298                        }
```

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);
```

```
     309  +                            changeInfo.getOriginPath(), changeInfo.getOriginCommit()));
     310  +                    }
     311  +                }
     312  +
307  313          /**
308  314           * Patches the changes types for the given change entries by manually
309  315           * 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

Done **Issue TS-18855** - **Deep**

Creator: Lars Heinemann (on

Assignee: Lars Heinemann

project	Type	Pr
TS	Maintenance	No

Labels	Affected Version	
code-retrospective	master	

Merge Request

https://git.cqse.eu/cqse/teamscale/merg

Description

Code Retro 5.0

⯆ Affected files **1**

⯆ Test Gaps

⯆ Findings balance ☑ 1

⯆ Issue commits **3**

⯅ Issue commits **3**

Merge branch 'cr/18855_deep_nesting' into 'master'
by Andreas Sewe as revision f2819175 📋 in ⑂ master
Files: 1 changed

Apr 26 2019
10:58

CR#18855: rework
by Lars Heinemann as revision dc48e064 📋 in ⑂ cr/18855_deep_nesting
Files: 1 changed

Apr 26 2019
10:39

CR#18855: reduce nesting
by Lars Heinemann as revision 67f1ad80 📋 in ⑂ cr/18855_deep_nesting
Files: 1 changed
Findings: ☑ 1

Apr 25 2019
15:19
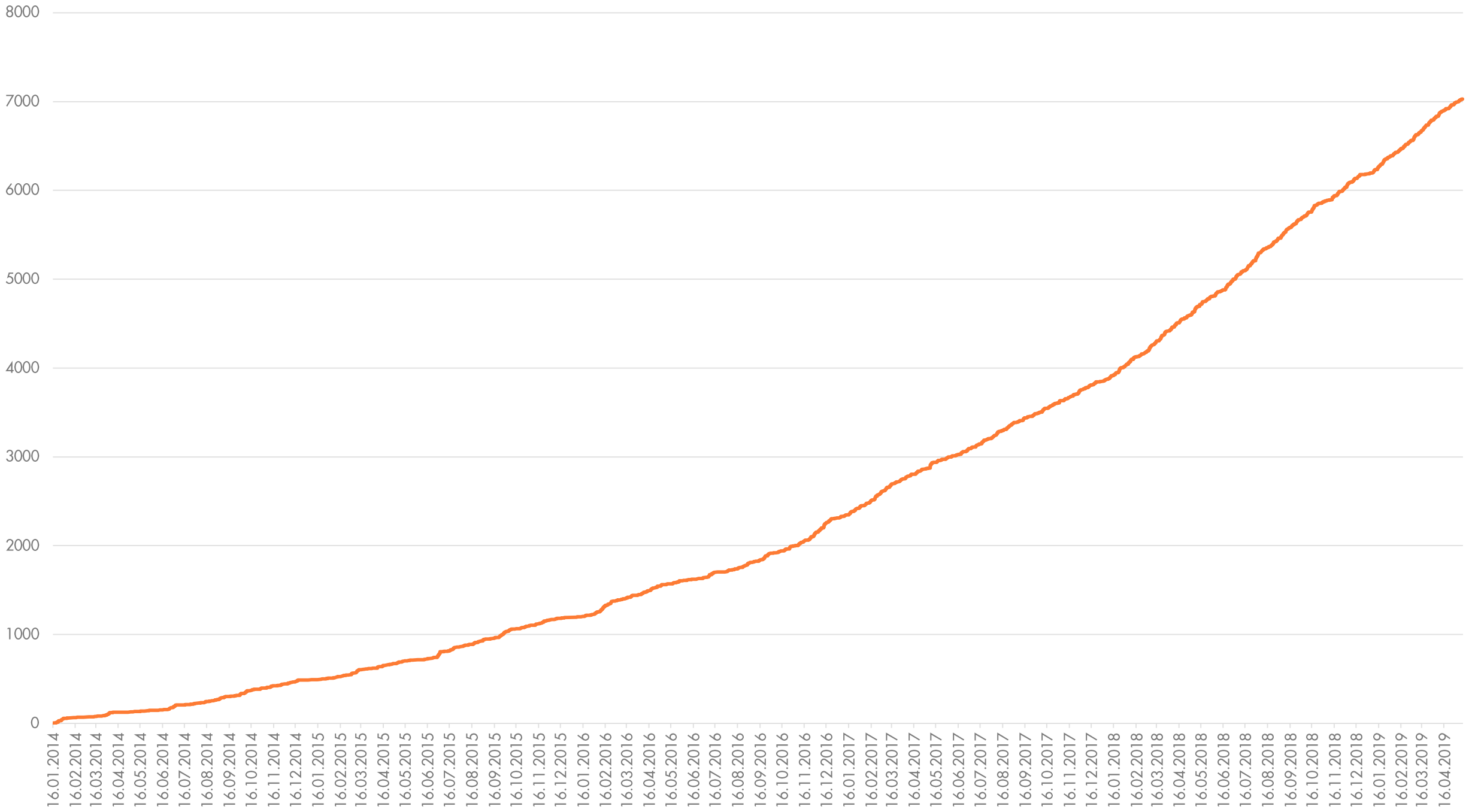
„Wachstumsschmerzen": Reviews > 2 Wochen

Query: closed = false && inState(status=Yellow) > 14d

Rework > 5d

Issue #	Subject	
TS-7608	Code - Add support to create code region URLs	Rework
TS-12642	Editing a dashboard always refers to the master branch	Rework
TS-15098	Separate permission view for assigning global roles	Rework
TS-15124	Inform user about pre-exsiting role assignments	Rework
TS-15835	External TestExecution data does not delta compress	Rework
TS-17941	Flag files that are part of test sources	Rework
TS-18217	Red error page when user limit is exceeded	Rework
TS-18406	Show Project Id on Log Overview Pages	Rework
TS-18621	Move instance comparison from service level to trigger level	Rework
TS-	Show percentages instead of absolute values in assessment tooltips	

Review > 3d

Issue #	Subject	
TS-9603	Rethink analysis group "General checks (built-in)"	In Review
TS-12911	Analysis profile documentation generator	In Review
TS-17038	Lost drag & drop operations in architecture editor due to slow reassessment	In Review
TS-17522	Project delete service should optionally delete dashboards and permissions	In Review
TS-17607	EditUserViewTest > testChangeUserPassword is flickering	In Review
TS-18969	Unintuitive user feedback for failing coverage uploads	In Review
TS-19155	Show Findings-Churn- and Test-Gap-Badges in Azure DevOps	In Review
TS-19161	Also show user's full name in assigned roles	In Review
TS-19326	False positive in "Implicit call to toString() on array" check	In Review

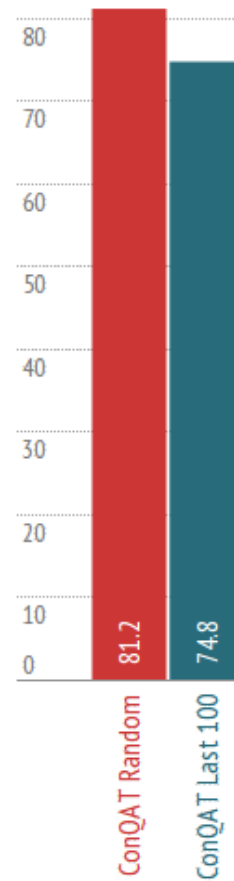# „Wachstumsschmerzen": Reviews > 2 Wochen

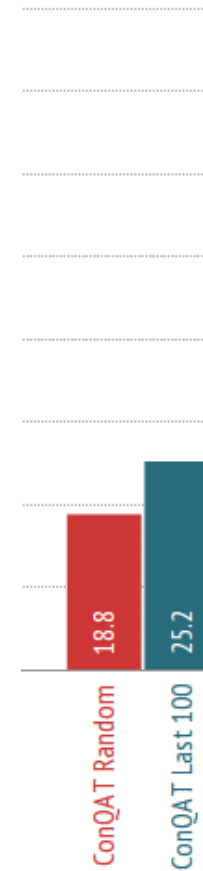**Query**:   closed = false && inState(status=Yellow) > 14d

Nutzen von Code Reviews

Code wird besser

Maintainability

Functional

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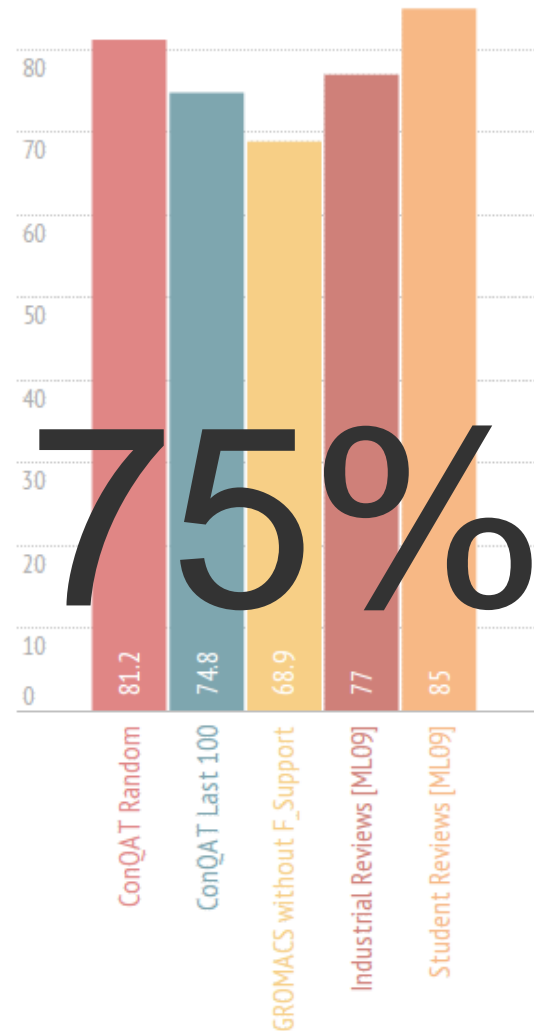
Maintainability

Functional

75%

25%

80	
70	
60	
50	
40	
30	
20	
10	
0	

ConQAT Random **81.2**
ConQAT Last 100 **74.8**
GROMACS without F_Support **68.9**
Industrial Reviews [ML09] **77**
Student Reviews [ML09] **85**

ConQAT Random **18.8**
ConQAT Last 100 **25.2**
GROMACS without F_Support **31.1**
Industrial Reviews [ML09] **23**
Student Reviews [ML09] **15**
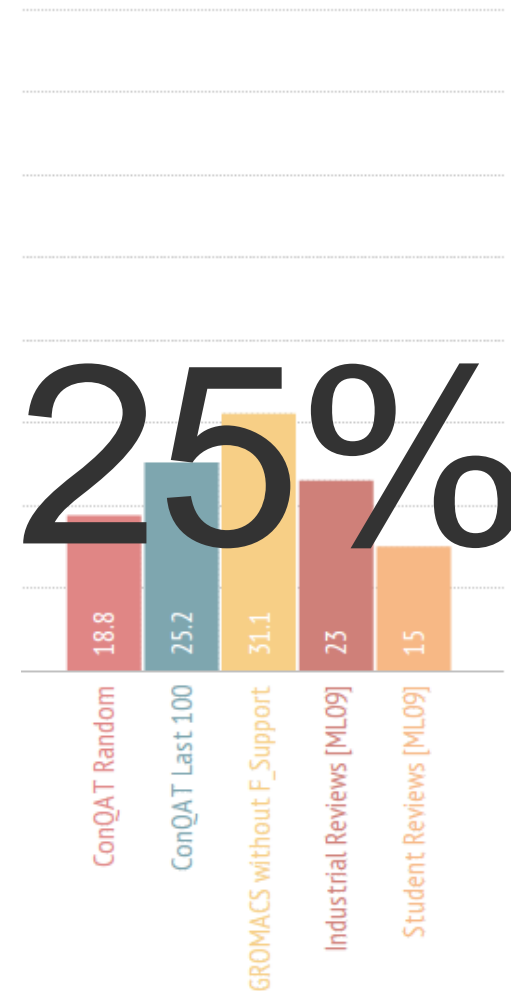
Nutzen von Code Reviews

Code wird besser

Feedback zu eigener Arbeit
- Persönliches Wachstum
- Man ist nie alleine „schuld" und traut sich mehr

Adrian H.	Christoph D.	Ingmar K.	Mehmet K.	Stefan W.
Albert St.	Christoph K.	Jakob R.	Michael A.	Stephan d. C.
Alexander von R.	Christopher S.	Jan S.	Michael K.	Sven A.
Alice R.	Corneliu P.	Jason F.	Michael L.	Thiemo T.
Anatol D.	Daniel B.	Jeremias W.	Michael S.	Thomas D.
Andi S.	Daniel K.	Johannes V.	Moritz B.	Thomas K.
Andre K.	Daniel V.	Jonathan C.	Nadja S.	Thomas P.
Andreas G.	Daniela St.	Katrin K.	Nils G.	Tilman S.
Andreas P.	Dennis P.	Klaus L.	Nils K.	Tim S.
Andreas S.	Dominik B.	Lars H.	Noha K.	Timo B.
Andreas W.	Elijah E.	Marcel B.	Patrick B.	Timo P.
Andreas W.	Elmar J.	Maria D.	Peter G.	Tobias R.
Ann Katrin G.	Emilija M.	Marius M.	Rainer N.	Valentin Z.
Ashutosh B.	Enis C.	Markus H.	Ralf B.	Vasileios K.
Athanasios C.	Eva N.	Martin F.	Raphael N.	Vasyl M.
Bakr A.	Fabian S.	Martin K.	Ricardo S.	Veronika B.
Benedikt H.	Florian D.	Martin P.	Robert B.	Viktoria M.
Benedikt S.	Florian Dr.	Matej S.	Roman H.	Viktoriia B.
Benjamin H.	Florian G.	Mathias B.	Rupert O.	Vincenz D.
Christian A.	Florian Z.	Mathieu P.	Sebastian E.	Witali A.
Christian D.	Fouad S.	Max G.	Sebastian R.	Zardosht H.
Christian P.	Gregor A.	Maximilian J.	Sebastian W.	
Christian Z.	Henning F.	Maximilian S.	Simon A.	

Nutzen von Code Reviews

Code wird besser

Feedback zu eigener Arbeit
- Persönliches Wachstum
- Man ist nie alleine „schuld" und traut sich mehr

Wissenstransfer
- Gemeinsames Produkt
- Jedes Ticket kann von mehreren Personen umgesetzt werden

Schafft Umgebung, in der man Verantwortung übernehmen kann.

~~Code~~ Peer Reviews

Aufgaben, Teams & Entwicklungspfade

Marketing & Vertrieb
- Angebot verhandeln
- Magazin-Artikel schreiben
- Vortrag auf Konferenz halten
- Demo halten

Requirement Engineers
- Neue Analysen ausdenken
- Anforderungen Priorisieren
- Forschung... ...beiten ...neuen ...

Entwicklung
- Code Review machen
- Doku schreiben
- Release erstellen
- Code aufräumen

Test
- Kundenumgebung nachstellen
- Integrationstest schreiben
- ...

Betrieb
- Updates einspiel...
- B...nde LDAP anbinden
- Monitoring aufsetzen
- ...
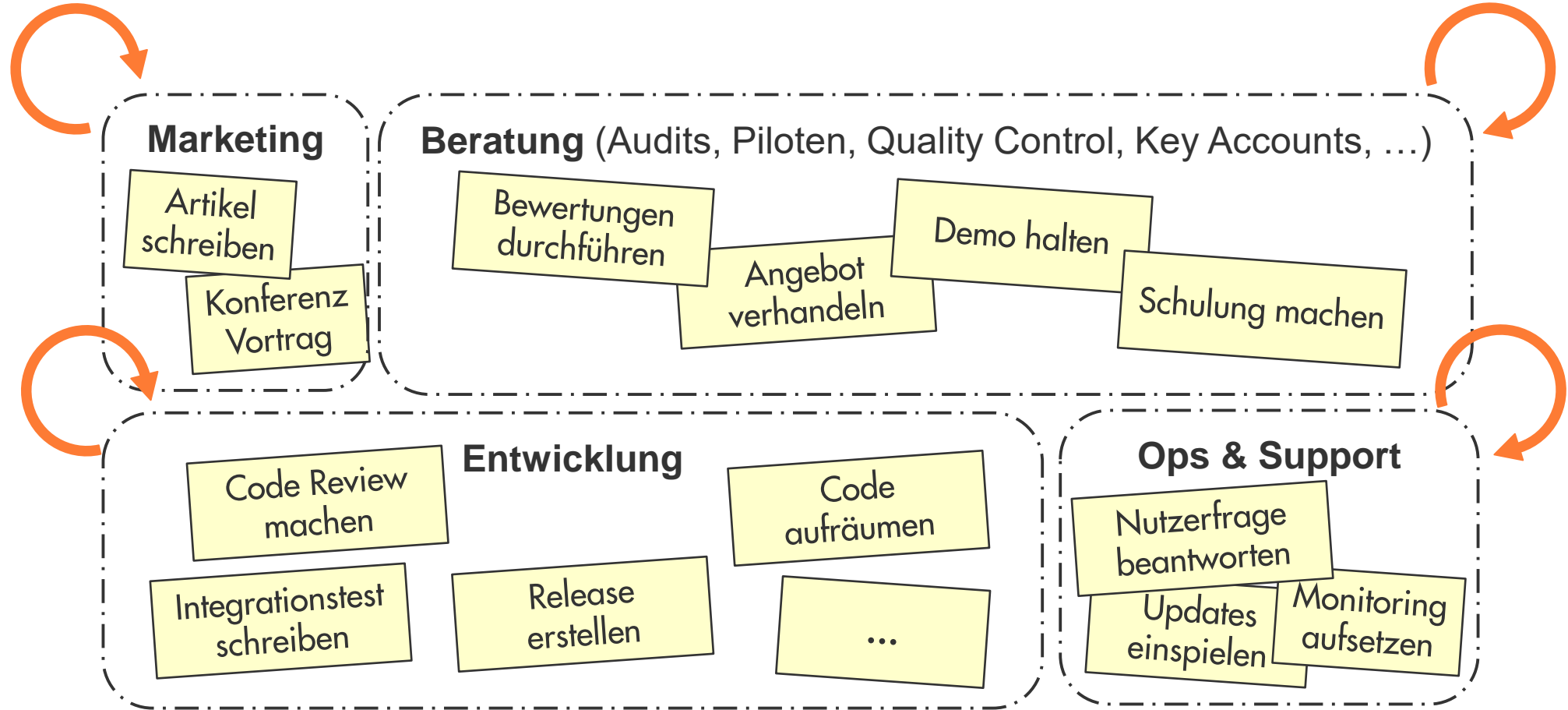
Support
- Nutzerfrage beantworten
- FAQ pflegen
- ...

Marketing

Beratung (Audits, Piloten, Quality Control, Key Accounts, …)

Entwicklung

Ops & Support

Marketing

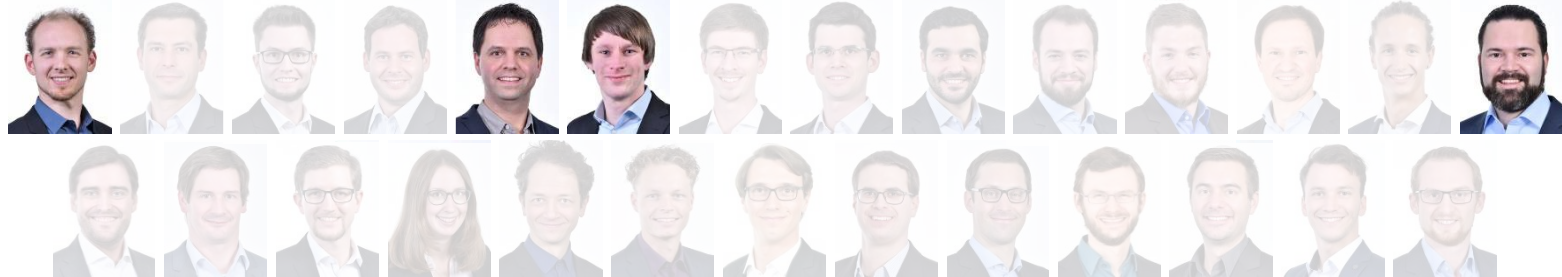**Beratung** (Audits, Piloten, Quality Control, Key Accounts, …)

Entwicklung

Ops & Support
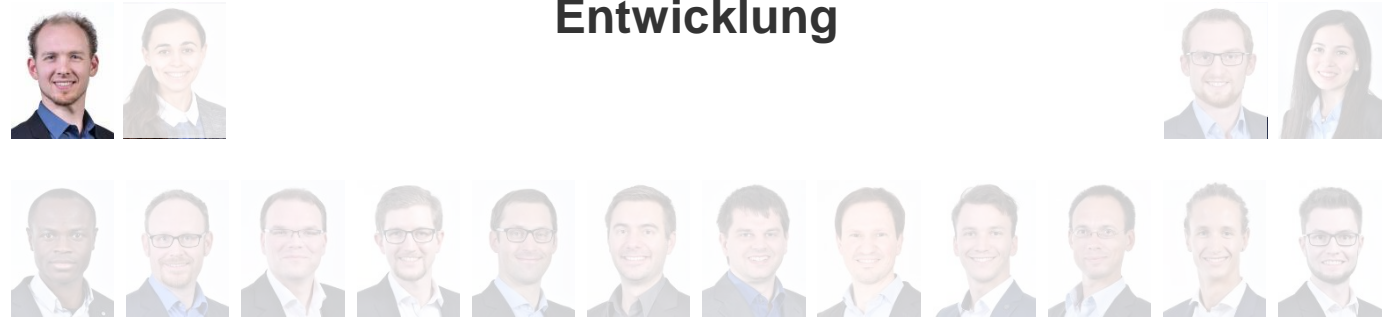
Marketing

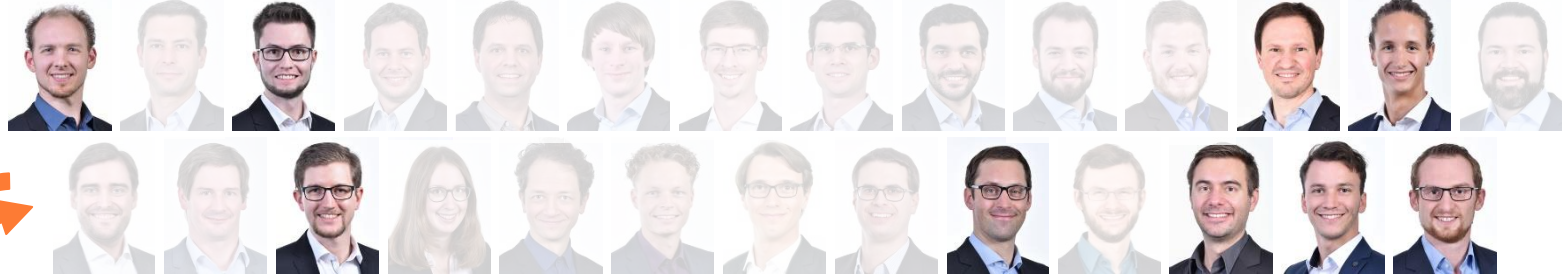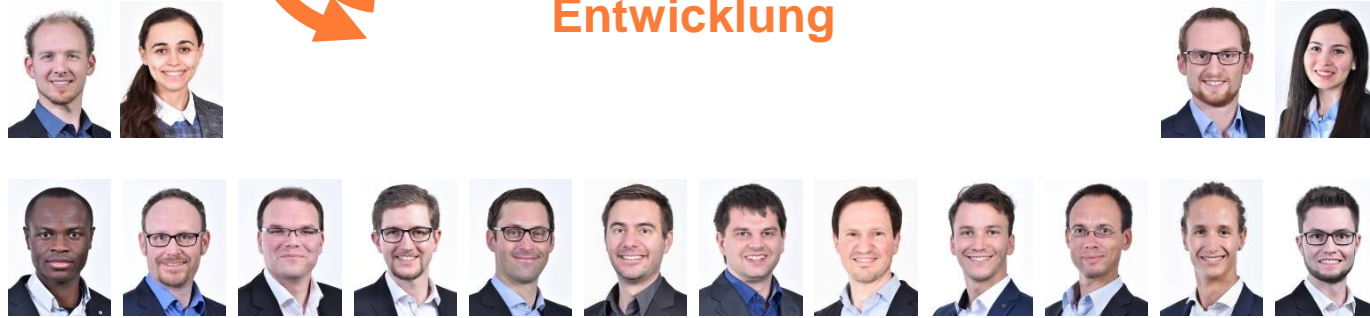**Beratung** (Audits, Piloten, Quality Control, Key Accounts, …)
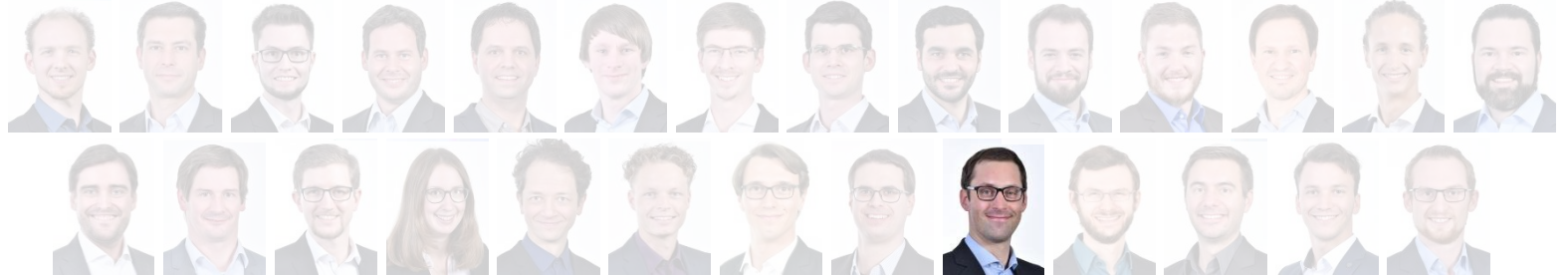
Entwicklung

Ops & Support

Marketing

Beratung (Audits, Piloten, Quality Control, Key Accounts, …)

Entwicklung
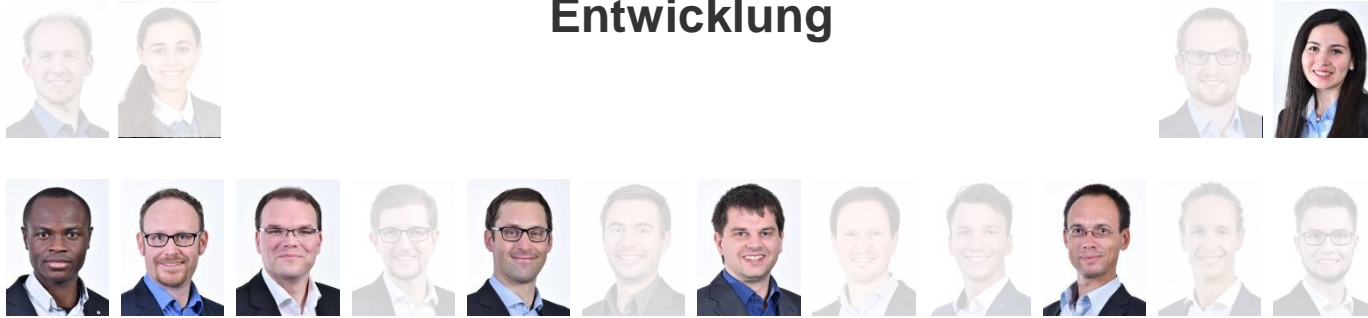
Ops & Support

Lessons Learned

- Bei vielen Aufgaben weiß man vorher nicht, ob es einem Spaß macht. Es muss einfach möglich sein, neue Aufgaben auszuprobieren.

- Es muss ohne Schaden für die eigene Karriere möglich sein, eine Aufgabe wieder abzugeben, falls sie einem nicht liegt.

- Manche möchten viele Aufgaben nicht machen und das ist auch okay.

Das Gehaltsmodell muss das unterstützen.

Fazit

Werkzeuge und Prozesse sind dann gut, wenn sie meine Arbeit besser machen und mir den Rücken frei halten.

Das gilt für Qualitätsanalysen genauso wie für alle anderen Werkzeuge und Prozesse in einer Firma.

Muss ich wirklich schon wieder alles testen?

Vortrag über Forschungsergebnisse und eigene Erfahrungen mit Testpriorisierung und -selektion durch die Test-Impact-Analyse.

- Große automatisierte Test-Suites haben oft eine lange Laufzeit von Stunden oder Tagen.
- Feedback der Tests hinsichtlich neuer Fehler dauert deshalb zu lange.
- Durch den Einsatz von Test-Impact-Analyse können wir 90% der neuen Fehler in 2% der Zeit finden.
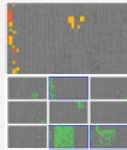
Best Presentation Award (Java Forum Stuttgart, 2018), Best Paper Award (QS-Tag, 2018)

▶ cqse.eu/yt-vortrag-test-impact-analyse

Faktenbasiertes Debuggen von Software Teams

Vortrag über die Nutzung und Analyse von Code, Versionshistorien, Ausführungsinformationen aus Test- und Produktivsystemen, um faktenbasierte Entscheidungen zu erleichtern und Problemen früh entgegen zu wirken.

- Häufig gibt es Probleme, wenn Software-Architektur und Organisationsstruktur nicht zusammenpassen.
- Dynamische und statische Analysen können helfen Symptome zu ermitteln, um Ursachen abzustellen.
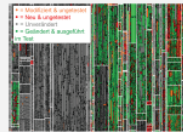
Best Presentation Award (Java Forum Stuttgart, 2017)

▶ cqse.eu/yt-vortrag-faktenbasiertes-debuggen

Vom Wiegen allein wird die Sau nicht fett

Vortrag über Erfahrungen, Lessons Learned und Best Practices aus sieben Jahren Praxiseinsatz von Qualitätsanalysewerkzeugen in unterschiedlichen Teams, Domänen und Firmen.

- Qualitätswerkzeuge sind notwendig, aber nicht hinreichend für Verbesserung der Software-Qualität.
- Unterstützung durch Entwickler und Management notwendig.
- Es muss sich jemand dafür verantwortlich fühlen.

Best Presentation Award (Java Forum Stuttgart, 2015)

▶ cqse.eu/yt-vortrag-nur-wiegen-reicht-nicht

Forschung

Wir forschen aktiv in den Bereichen Softwarequalität, -wartung, -evolution, -test und Statische-Analyse-Werkzeuge.

Eine Liste aller Veröffentlichungen ist hier auf unserer Webseite.

🔗 cqse.eu/veroeffentlichungen

Impulsvorträge

Um unsere Ergebnisse und Erfahrungen weiterzugeben, tragen wir regelmäßig auch in Firmen vor. Die Themen reichen von Qualitätsanalysen über Qualitätscontrolling bis hin zu Testcontrolling oder der Einführung von Reviews.

🔗 cqse.eu/impulsvortraege

Veranstaltungen zu Software Intelligence

Wir organisieren regelmäßig Veranstaltungen und Vorträge zu Software- und Test-Intelligence:

- Software Intelligence Workshop
- Software Intelligence Nights

Wir suchen Verstärkung in München oder Remote

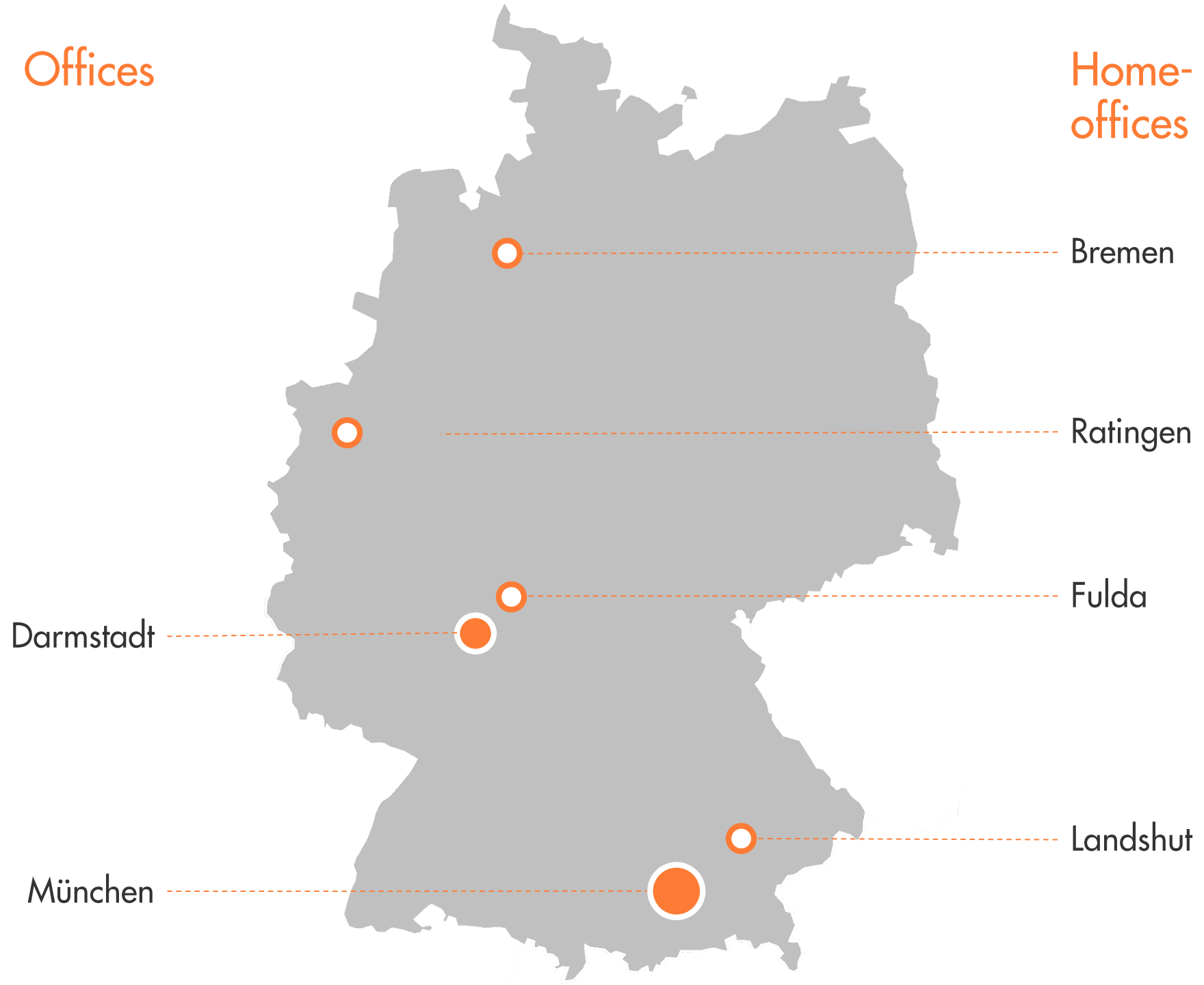### Bei Interesse bitte melden bei: Elmar Jürgens, juergens@cqse.eu

Offices

Home-offices

Bremen

Ratingen

Fulda

Darmstadt

Landshut

München

Kontakt – **Ich freue mich auf Diskussionen** ☺

Dr. Elmar Jürgens · juergens@cqse.eu · +49 179 675 3863

CQSE GmbH
Lichtenbergstraße 8
85748 Garching bei München
www.cqse.eu

CQSE