

# Continuous Quality Control

## Qualität trotz immer kürzerer Releasezyklen

---

JFS 2025

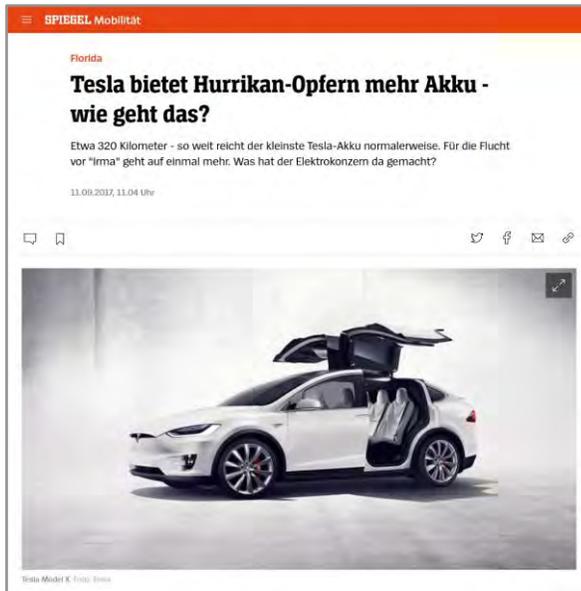
**CQSE**

Dr. Tobias Röhm

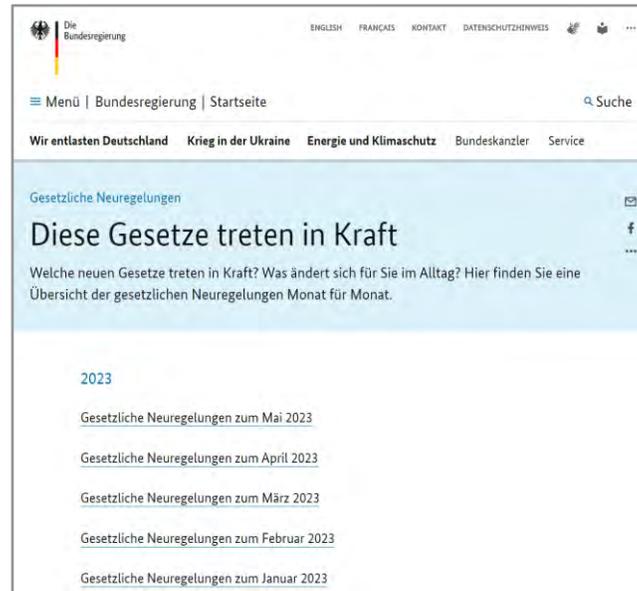
# Dr. Tobias Röhm



# Kontext



Ereignisbasierte  
Over-the-Air-Updates



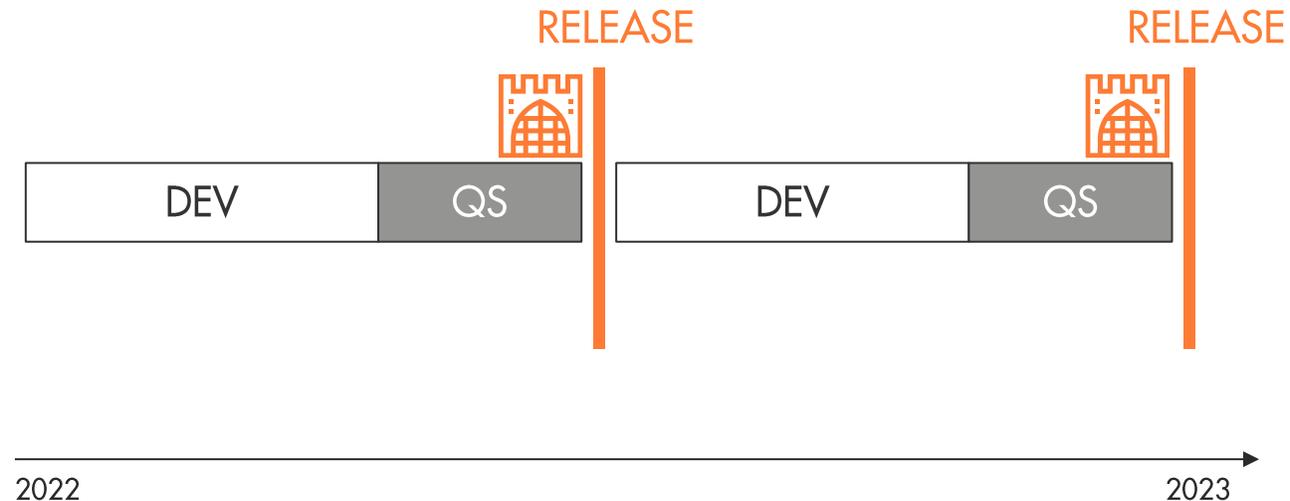
Viele, teilweise rückwirkende  
Gesetzesänderungen



Zeitkritische Security-Patches

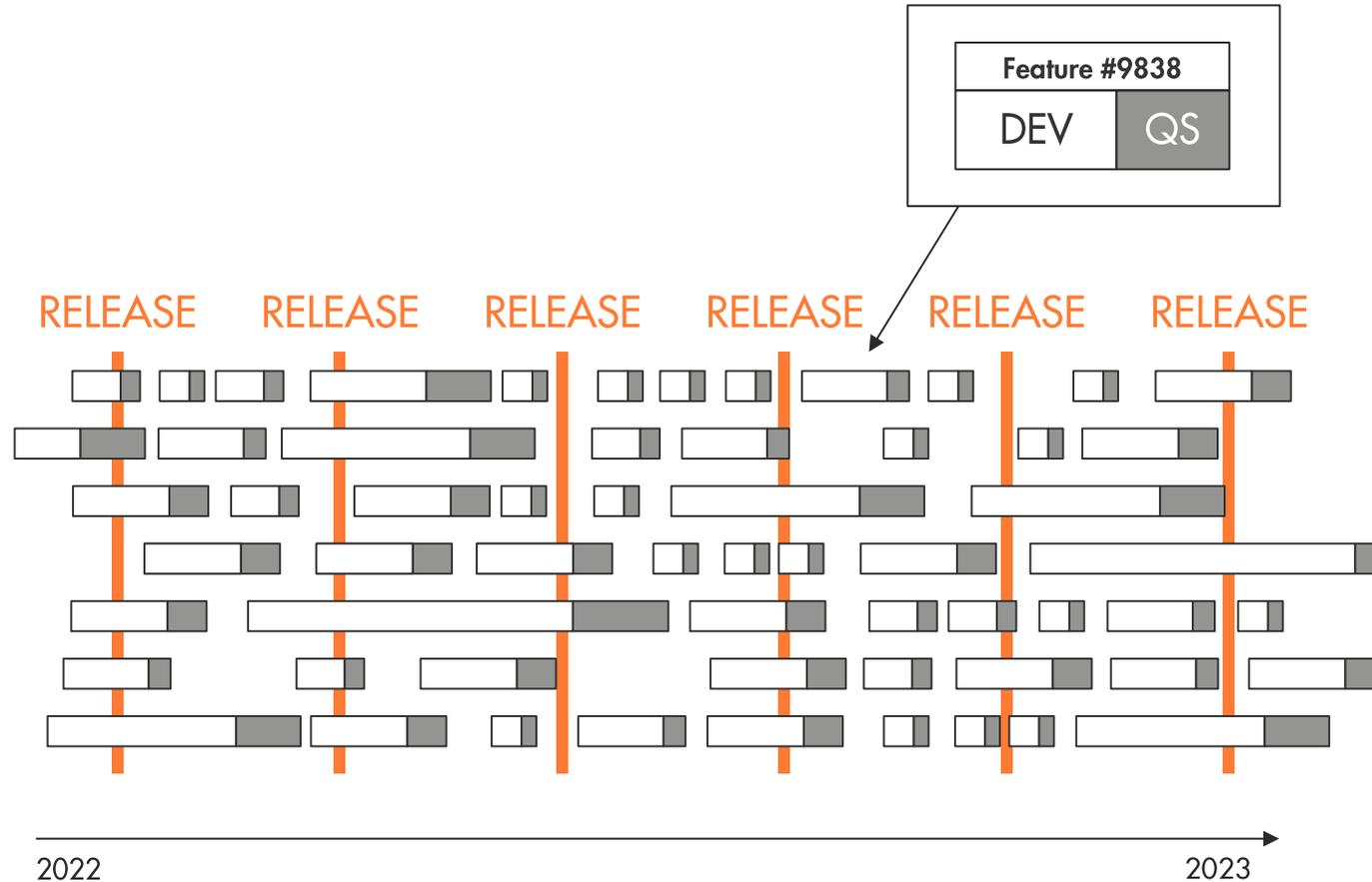
- Software-Updates müssen jederzeit möglich sein
- Software muss jederzeit ein releasefähiges Qualitätsniveau aufweisen

# Problem: Traditionelle QS-Prozesse



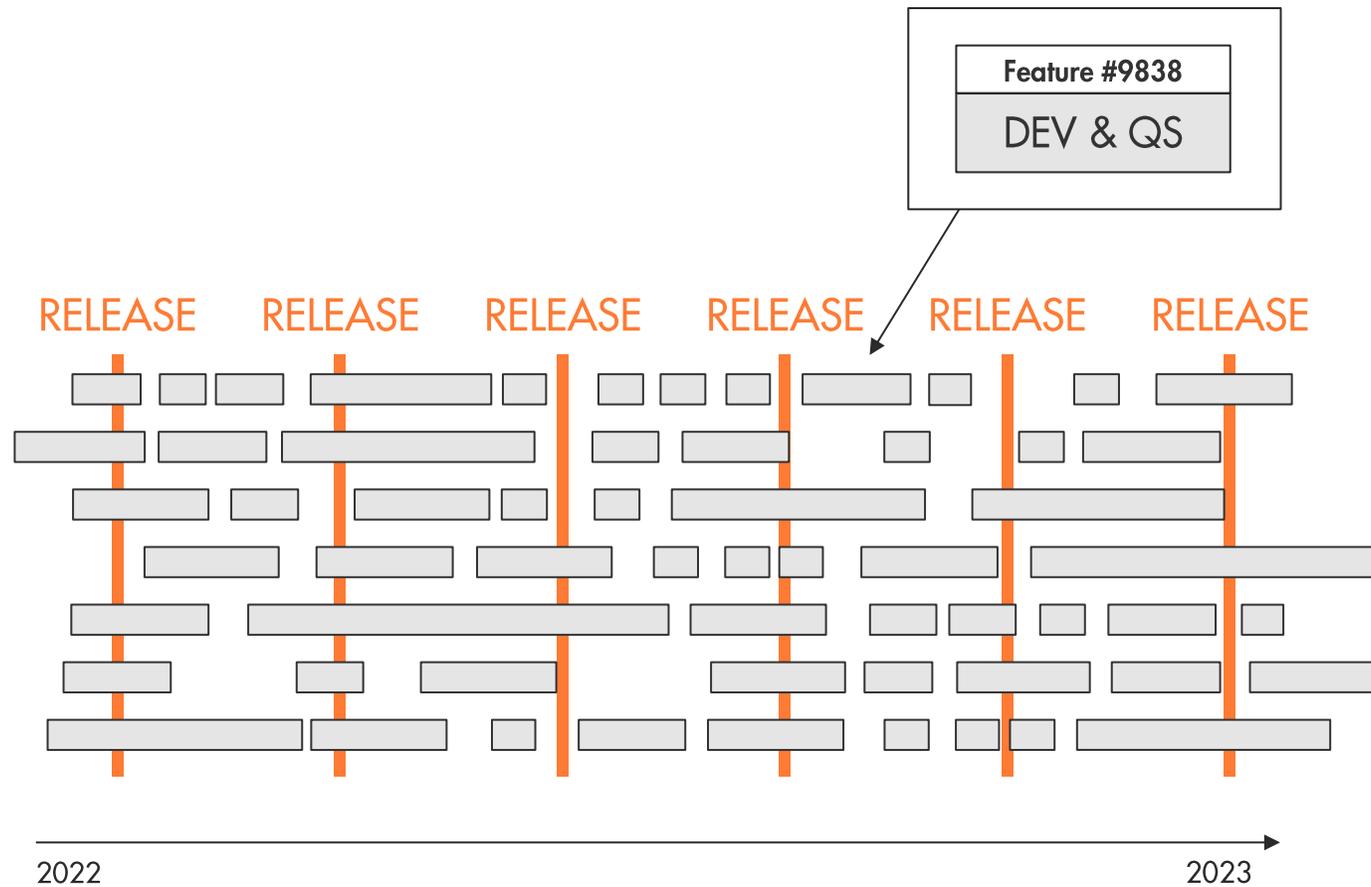
- Traditionelle QS-Prozesse sind (zu) langsam
- Software ist NICHT jederzeit auf einem releasefähigem Qualitätsniveau

# Lösung: Continuous Quality Control (CQC)



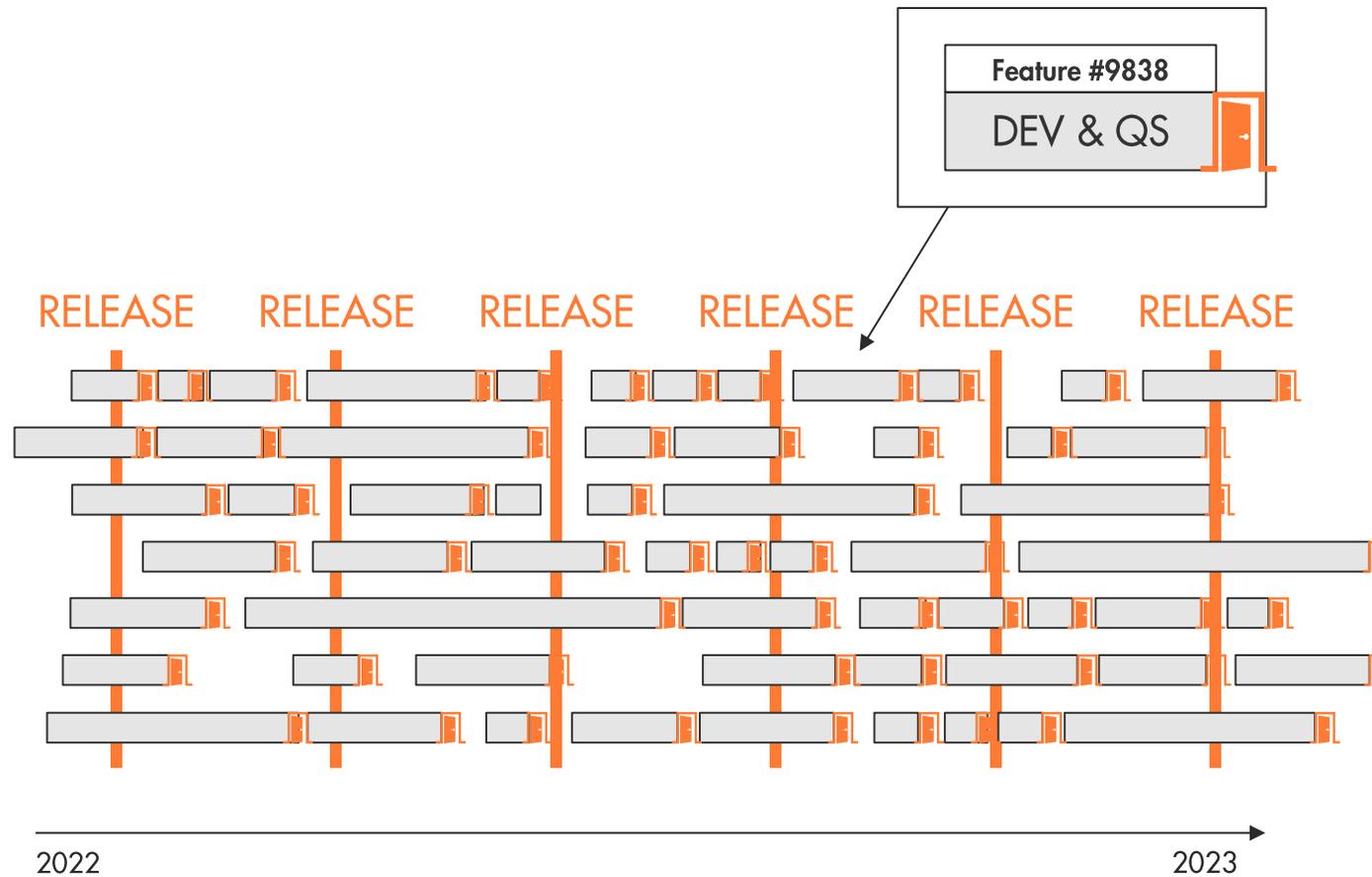
→ Qualitätssicherung auf Ticketebene um zu parallelisieren und zu beschleunigen

# Lösung: Continuous Quality Control (CQC) II



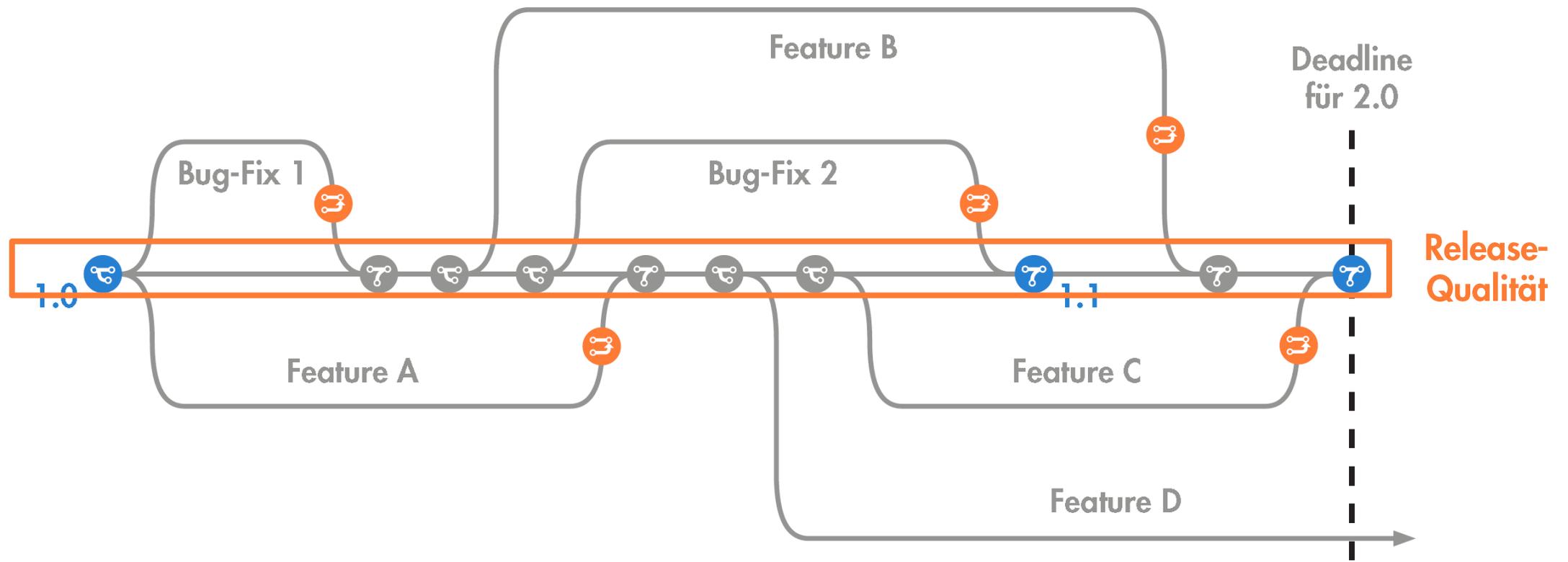
→ Verschmelzung von Entwicklung und QS um zu beschleunigen

# Lösung: Continuous Quality Control (CQC) III



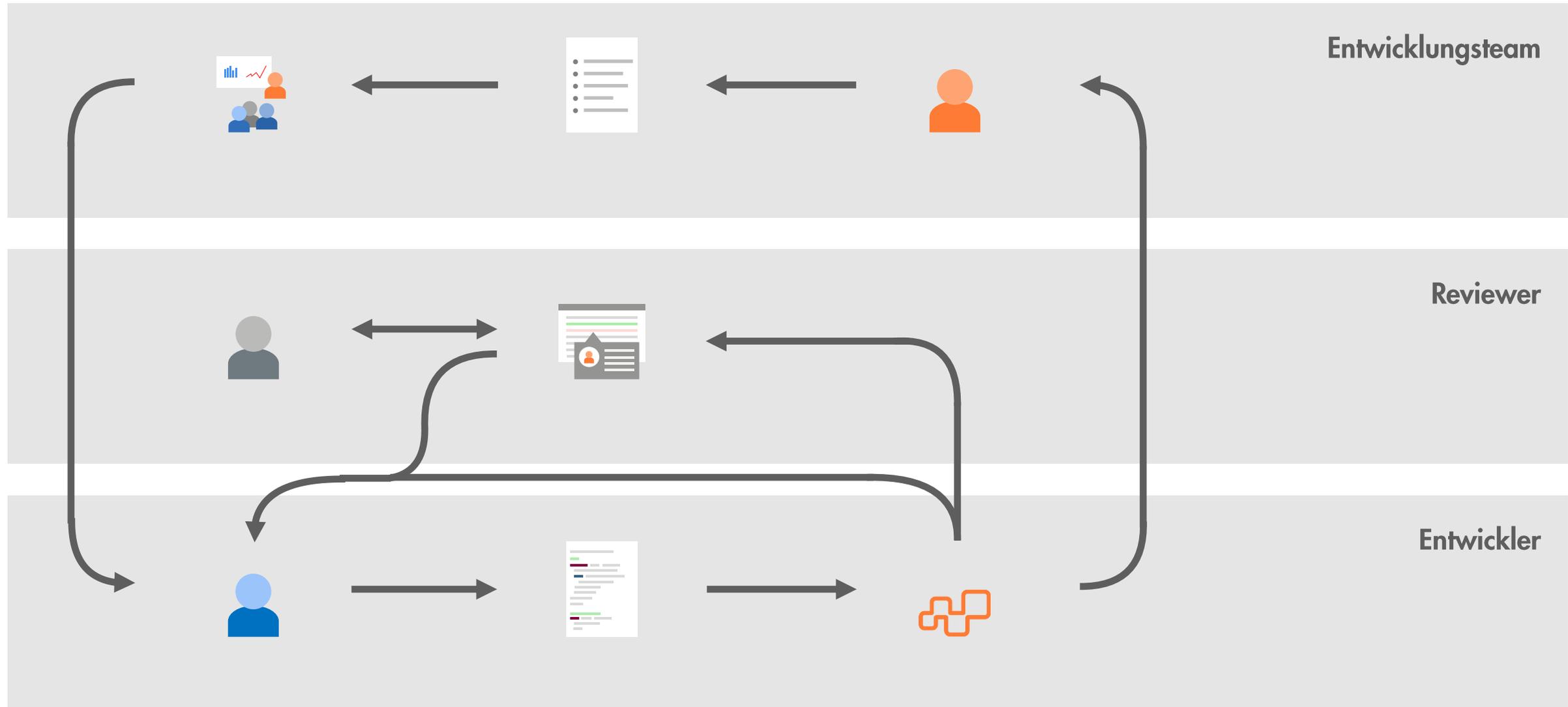
- Qualitätstürchen als frühe, leichtgewichtige Qualitätsprüfung
- Qualität ist jederzeit auf releasefähigem Niveau

# Implementierung durch Feature Branches in Git



→ Merge Request als »Qualitätstürchen«

# Agenda: Continuous Quality Control-Feedbackschleifen



Feedbackschleife 1:  
Statische Codeanalyse hilft Entwicklern,  
Fehler & Q-Defizite zu vermeiden

# Feedbackschleife 1: Statische Codeanalyse hilft Entwicklern, Fehler & Q-Defizite zu vermeiden

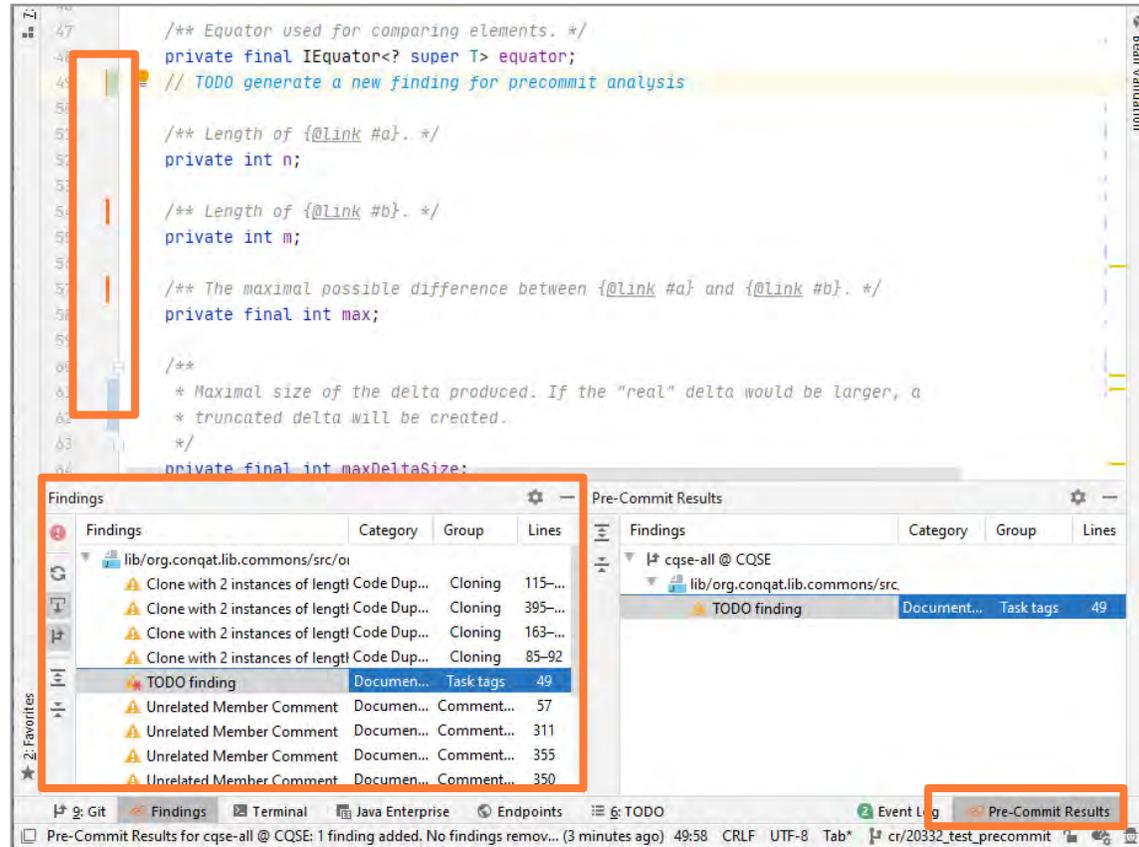


# Übersicht Statische Codeanalyse



→ Statische Codeanalyse schafft Transparenz bezüglich Code- & Architekturqualität

# IDE-Integration & Precommit-Analyse



Precommit-Analyse ermöglicht Q-Feedback bereits vor VCS-Commit

Inkrementelle Analyse erlaubt schnelle Ergebnisse auch für komplexe Analysen

→ Feedback aus statischer Codeanalyse ist schnell, fokussiert und integriert nutzbar

# Findings-Churn & Qualitätsziele für gewachsene Systeme

 **Share more code between dispatchSessionStorageEvents() and dispatchLocalStora...** Yesterday 17:34  
by Chris Dumez as revision ffa7b160 in main (github)  
Files: 1 changed  
Findings 

 **CachedResourceLoader::allCachedSVGImages() reparses resource URLs unnecessari...** Yesterday 17:26  
by Ch...  
Files:  
Findi...

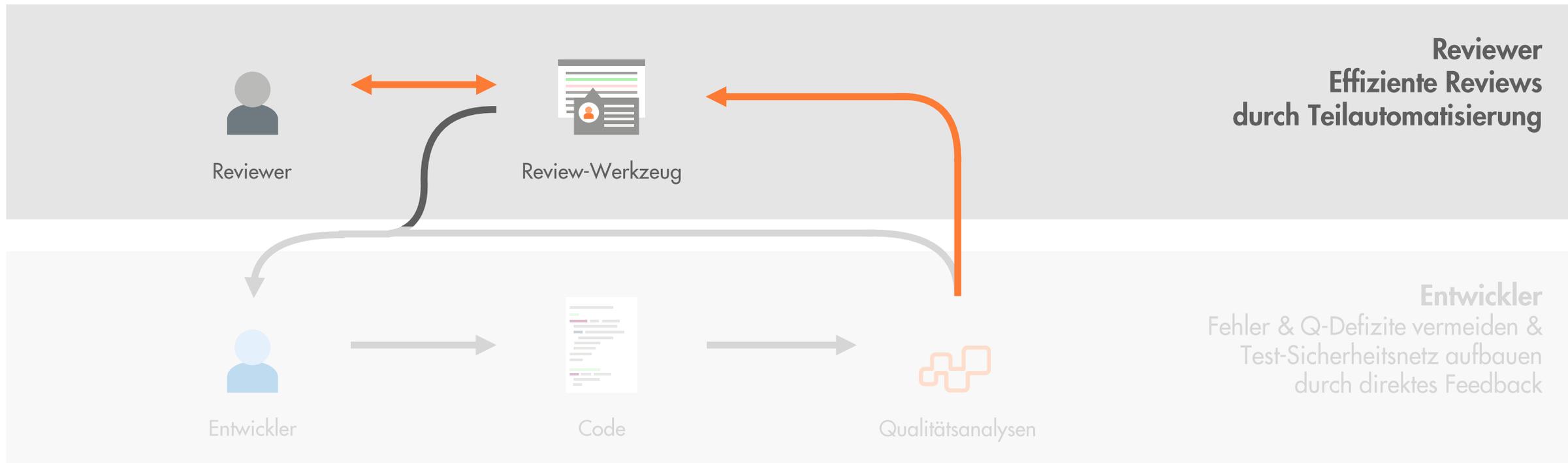
Unterscheidung zwischen »neuen Findings« und »Findings in geändertem Code« erlaubt schrittweise Q-Verbesserung von gewachsenen Codebasen mit der Pfadfinderregel.

 **[WinCairo][WK2] animations/par... is timing out https://bug...** Yesterday 13:20  
by Fujii Hironori as revision 2d295... in main (github)  
Files: 2 changed  
Findings 

 **[git-webkit] Link to GitHub wiki https://bugs.webkit.org/show\_bug.cgi?id=2370...** Yesterday 13:11

Feedbackschleife 2:  
Statische Codeanalyse und Test-Gap-Analyse  
ermöglichen Reviewern effiziente Code Reviews

# Feedbackschleife 2: Stat. Codeanalyse und Test-Gap-Analyse ermöglichen Reviewern effiziente Code Reviews



# Annotation von Findings aus statischer Codeanalyse an Merge/Pull Requests

The screenshot shows a pull request titled "Modify the code #7". At the top, it indicates "Fix method not being called" and "ac2cbc2". A Teamscale (SWE) finding is highlighted with a red box, showing a "Findings" count of 1. The finding details are as follows:

- Added Findings:** This pull request would introduce 1 new finding.
- DETAILS:** Shows the Teamscale logo and a finding count of 1.
- ANNOTATIONS:** A warning icon is shown next to the text: "Check warning on line 115 in WhiteBoard/WBProcessManager.m". The finding is attributed to "teamscale-swe / teamscale-findings" and "WhiteBoard/WBProcessManager.m#L115". The message is: "TODO (AS) We should switch this to always log." A link to the finding details is provided: "https://teamscale.my-company.com/findings.html#details/sam-faq-plms-project-whiteboard?merge\_request\_tga\_d".

The screenshot shows a pull request titled "PR #4". It displays a code diff for two files: "jabref/TableGUI.java" and "jabref/logic/bibtex/BibEntryWriter.java". Two Teamscale finding annotations are highlighted with red boxes:

- Annotation 1:** A warning icon is shown next to the text: "Check warning on line 35 in jabref/tableGUI.java". The finding is attributed to "Teamscale App / Teamscale Findings" and "jabref/jabRefGUI.java#L35". The message is: "Interface comment missing".
- Annotation 2:** A failure icon is shown next to the text: "Check failure on line 166 in jabref/logic/bibtex/BibEntryWriter.java". The finding is attributed to "Teamscale App / Teamscale Findings" and "jabref/logic/bibtex/BibEntryWriter.java#L166-L168". The message is: "Catch clause catches generic exception 'Exception'".



Reviewer/Test(er)



Funktioniert Ticket-Implementierung?  
Ging aus Versehen etwas kaputt?

Anwendung

# Annotation von Testergebnissen an Merge/Pull Requests

## Cr/20002 new report template

Edited 9 hours ago by build

 **Request to merge** `cr/20002_new_report_t...`  **into** `teamscale/v5.4.x` Open in Web IDE Check out branch 

The source branch is 31 commits behind the target branch

 **Pipeline #89447 failed for 040f1ce9 on cr/20002\_new\_report\_t...** ✓ ✓ ✗ »

 Test summary contained 5 failed test results out of 4318 total tests Expand

 **Merge**  Delete source branch  Squash commits [?](#)

> **8 commits** and **1 merge commit** will be added to `teamscale/v5.4.x`. [Modify merge commit](#)

*You can merge this merge request manually using the [command line](#)*



Reviewer/Test(er)

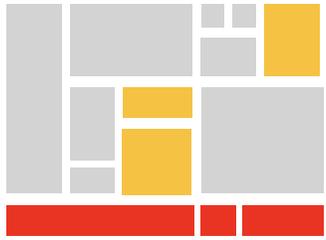


Funktioniert Ticket-Implementierung?  
Ging aus Versehen etwas kaputt?

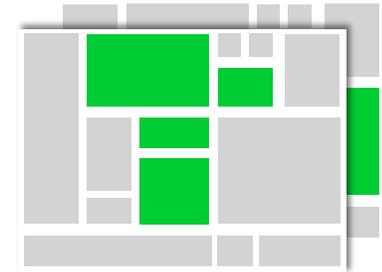
Wurden alle Codeänderungen getestet?

Anwendung

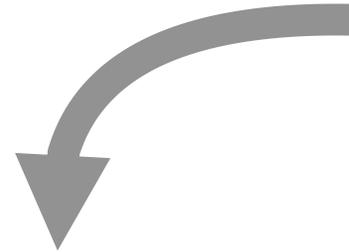
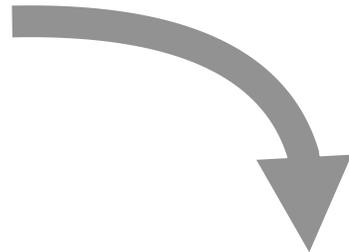
# Test-Gap-Analyse



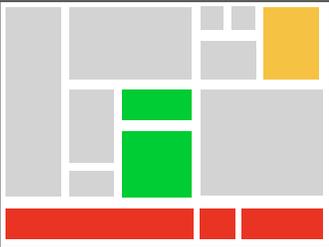
Codeänderungen



Testcoverage



 Teamscale



Test Gaps =  
Ungetestete Codeänderungen

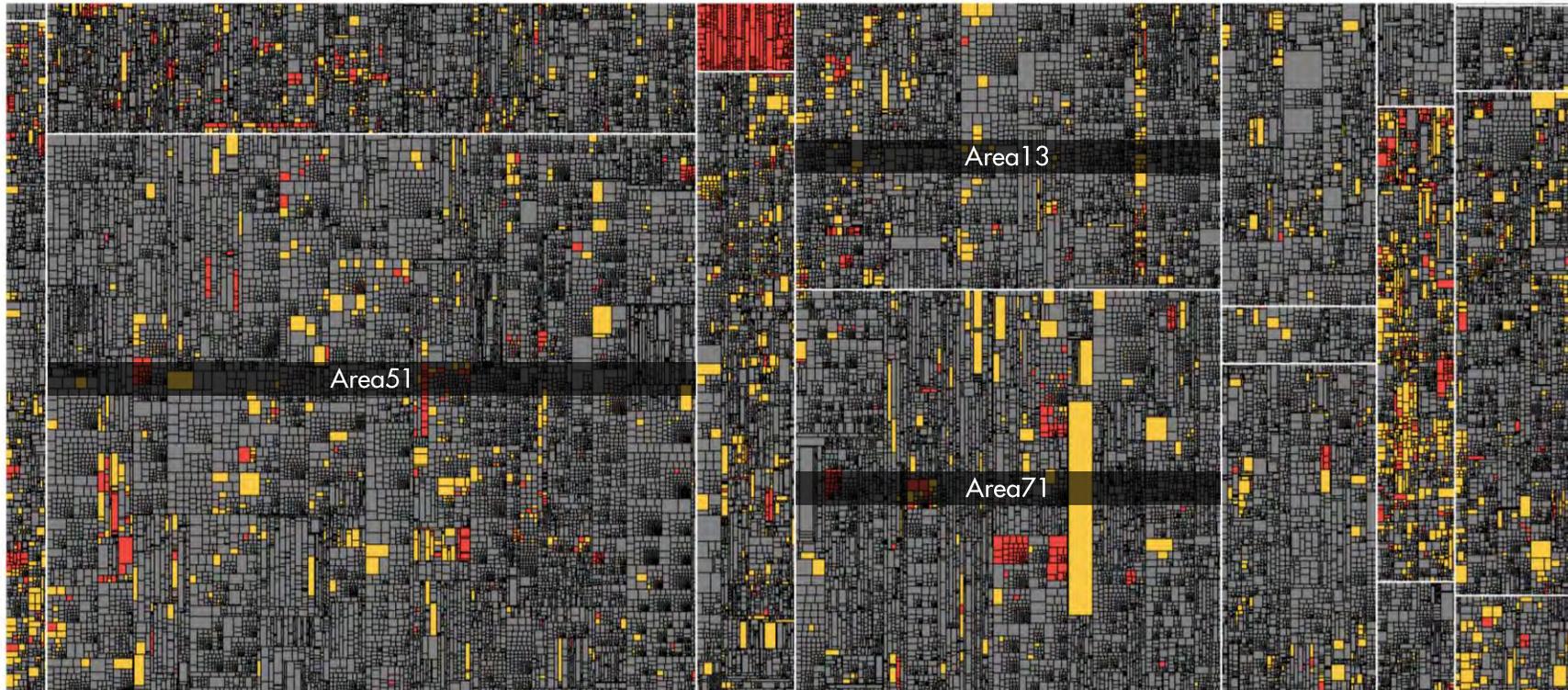
Test-Gap-Analyse identifiziert ungetestete Codeänderungen. Diese haben ein hohes Fehlerrisiko.

Did we test our changes? Assessing alignment between tests and development in practice. (Eder et al. 2013)

# Codeänderungen

Churn

Jul 10 2022 - Aug 10 2022

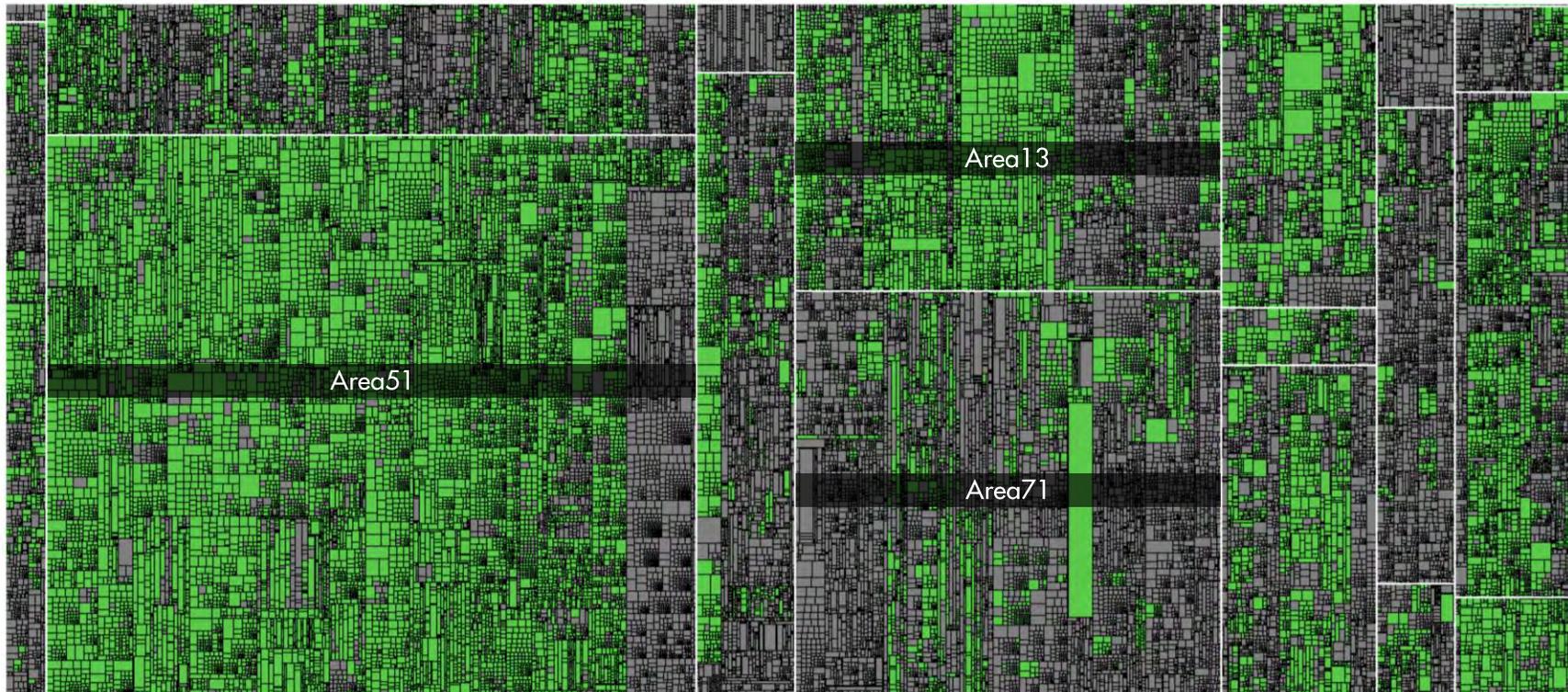


Methoden wurden...

- ... nicht geändert
- ... hinzugefügt
- ... geändert

# Testcoverage

Function Execution  
Jul 10 2022 - Aug 10 2022 | Execution Ratio: 43.79%

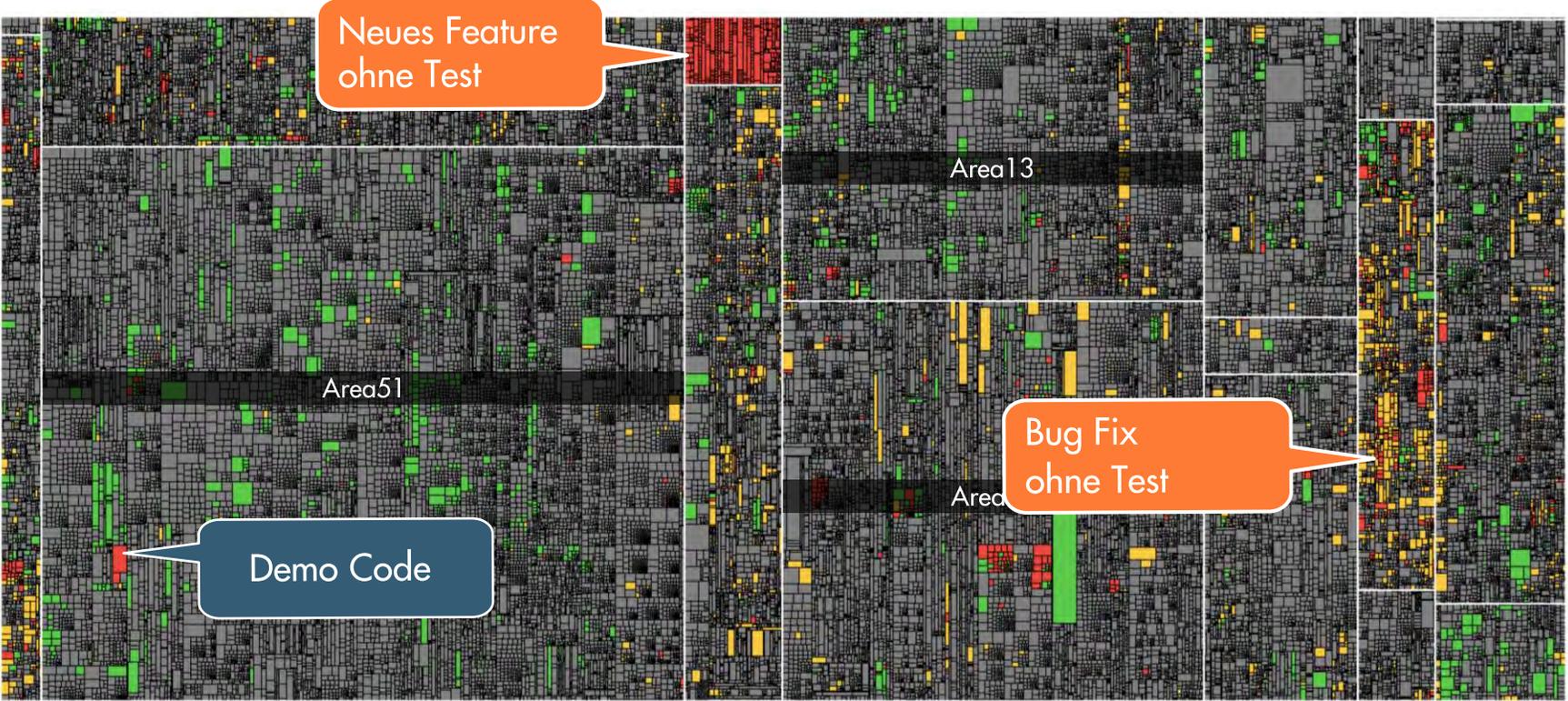


Methoden wurden...

- ... nicht getestet
- ... im Test ausgeführt

# Test Gaps (= geänderter, ungetesteter Code)

Test Gaps  
Jul 10 2022 - Aug 10 2022 | Test Gap: 63%



Methoden wurden...

- ... nicht geändert
- ... hinzugefügt & nicht getestet
- ... geändert & nicht getestet
- ... geändert & getestet

# Annotation von Test Gaps an Merge/Pull Requests

Code Pull requests 2 Projects Security Insights

## Modify the code #7

Open asteckermeier wants to merge 2 commits into [ ] from merge\_request\_tga\_demo\_5

Conversation 0 Commits 2 Checks 1 Files changed 1

Fix method not being called ac2cbc2

Teamscale (SWE)

teamscale-findings Resolve

### Added Findings

This pull request would introduce 1 new finding

DETAILS

Teamscale Findings 1 2 2 Test Gaps (67%)

WhiteBoard/WBProcessManager.m#L115: TODO (AS) We should switch this to always log. (view in Teamscale)

ANNOTATIONS

Check warning on line 115 in WhiteBoard/WBProcessManager.m

teamscale-swe / teamscale-findings

WhiteBoard/WBProcessManager.m#L115

TODO (AS) We should switch this to always log.

https://teamscale.my-company.com/findings.html#details/sam-faq-plms-project-whiteboard?merge\_request\_tga\_d

View more details on Teamscale (SWE)

## 7 Modify the code

Request to merge merge\_request\_tga\_demo\_5 into [ ] OPEN NEGATIVE

This change contains 3 commits and introduces 1 new finding

Test Gaps

May 21 2022 12:22-Now | Test Gap: 67% Coverage sources Sydney-IOS-TOT

Findings 1 2 2

New findings

Message	Location	Finding Group
TODO More implementation needed	FrontBoard/FBProcessManager.m:136	Documentation > Task Tags > Comments/Task ...

# Test Gaps je Ticket

**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 [https://git.cqse.eu/cqse/teamscale/merge\\_requests/2963/](https://git.cqse.eu/cqse/teamscale/merge_requests/2963/) PDash Task QA-Contact  
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**



- Test-Gap-Analyse schafft Transparenz über Test Gaps
- Test-Gap-Analyse unterstützt beim effizienten Aufbau von Test-Sicherheitsnetz

# Code Review-Kommentar in GitLab

Discussion 1 Commits 2 Pipelines 2 Changes 1 Show all activity 1/1 thread resolved

Andreas Sewe @sewe started a thread on an old version of the diff 4 months ago  
Resolved by Lars Heinemann 4 months ago [Toggle thread](#)

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

```
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,
298 -             tokenElements.get(i));
299 -             if (changeType != null) {
300 -                 result.add(new RepositoryChangeEntry(path,
301 -                     requiredNode.getRevision().getRevision(), changeType,
302 -                         commit, changeInfo.getOriginPath(),
303 -                             changeInfo.getOriginCommit()));
304 +         BasicTokenElementInfo element = tokenElements.get(i);
305 +         addChangeEntry(requiredNode, result, changeSet, path, element);
```

Andreas Sewe @sewe · 4 months 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.

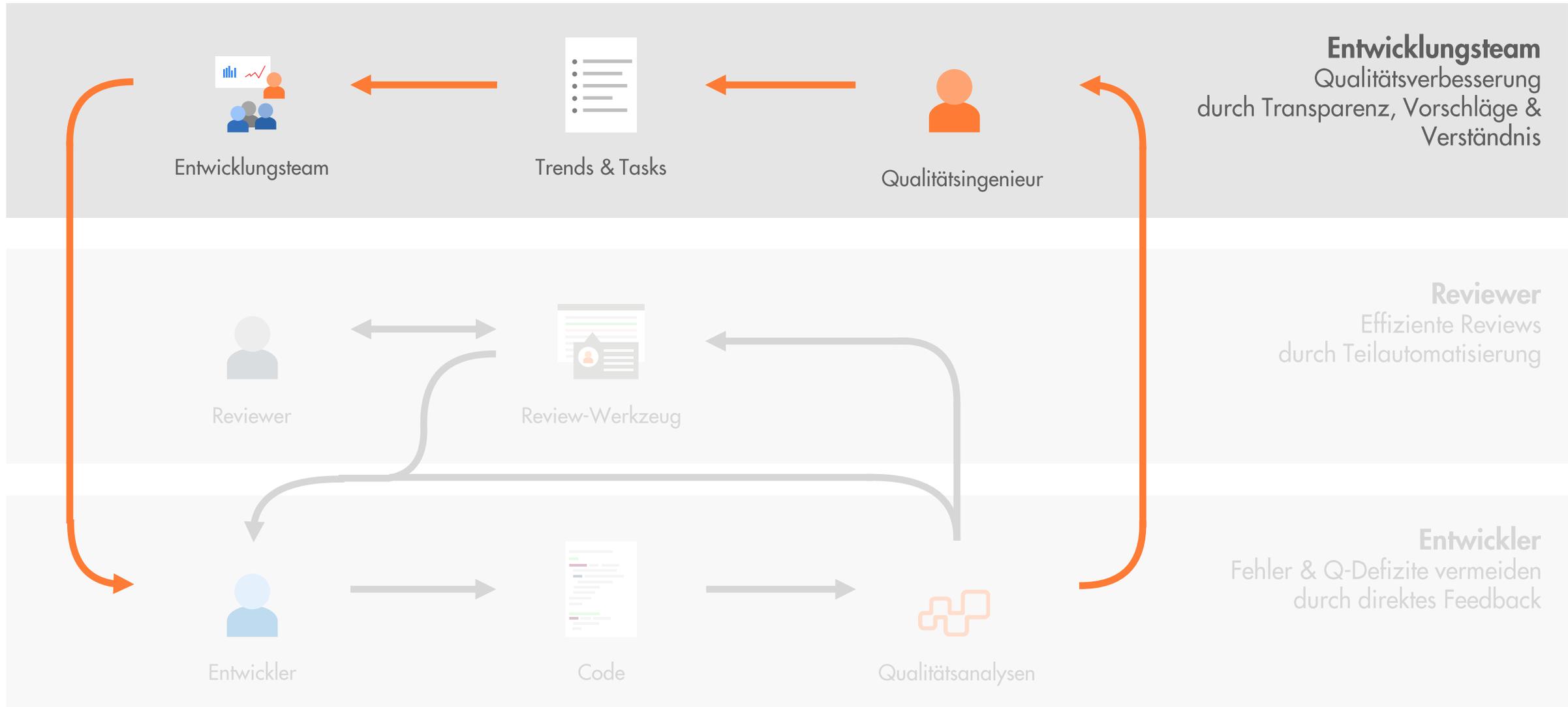
```
computeChangeEntry(requiredNode, changeSet, path, element).ifPresent(result::add);
```

Lars Heinemann @heinemann changed this line in version 2 of the diff 4 months ago

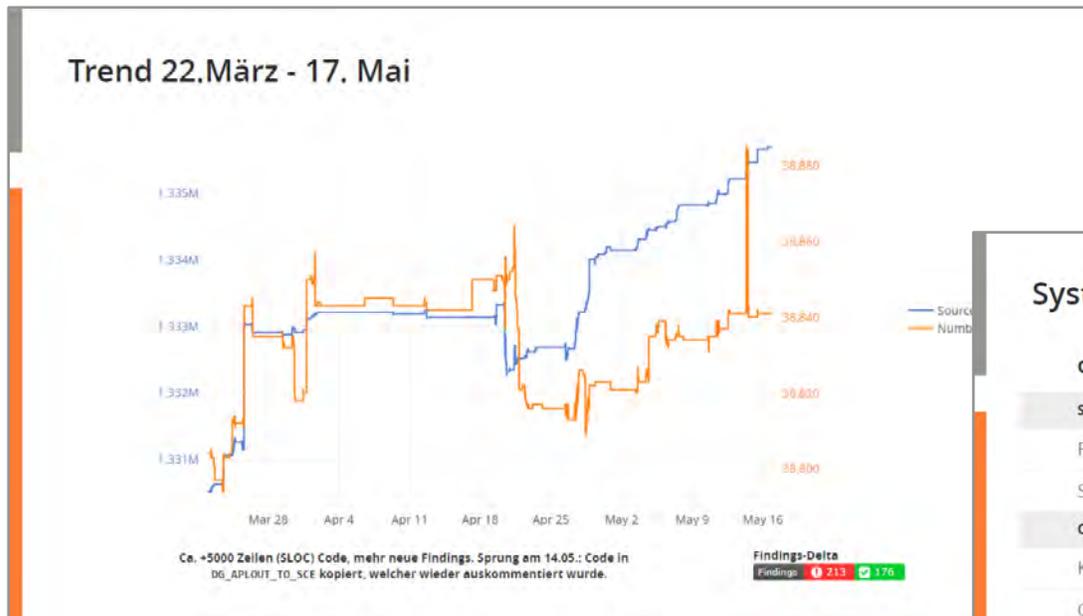
→ Hohe Eingangsqualität erlaubt effiziente, menschliche Code Reviews

Feedbackschleife 3:  
Q-Retrospektiven unterstützen Entwicklungsteams  
bei Qualitätssicherung & -verbesserung

# Feedbackschleife 3: Q-Retrospektiven unterstützen Entwicklungsteams bei Qualitätssicherung & -verbesserung



# Analyse & Diskussion von Qualitätstrends



### System Quality Overview

Quality Indicator (QI)	Trend	Quality Indicator (QI)	Trend
<b>Security</b>		<b>Redundanz</b>	
Rote Security-Findings	→	Clone Coverage	↘
Security-Findings je 1.000 Codezeilen	↗	<b>Codestruktur</b>	
<b>Codeanomalien</b>		Struktur: Prozedurlänge	→
Korrektheit	↗	Struktur: Schachtelungstiefe	→
Codeanomalien je 1000 SLOC	→		

→ Trendanalyse schafft Transparenz für Team und Management

# Konkrete Verbesserungsvorschläge

**Task 34 - Redundanz zwischen UI5\_TRANSLATION\_TOOL und TRANSLATION\_TOOL**

created by [redacted] Sep 16 2021 15:22, last updated Sep 17 2021 17:19

Assignee: [redacted]

Status: Open

Resolution: None

Description: Der Code im neuen Paket [redacted] UI5\_TRANSLATION\_TOOL ist oftmals redundant zum Code in [redacted] TRANSLATION\_TOOL. Da beide Pakete wohl ähnliche Logik implementieren, sollte die Redundanz hier reduziert werden, z. B. durch verwenden von Basisklassen.

Tags: 3. Retro, Redundanz

Findings: 0 open, 17 resolved, 0 blacklisted

- Clone with 2 instances of length 18 in [redacted] TRANSLATION\_TOOL/CLASS [redacted] EL-TRANSLATION.abap:979
- Clone with 2 instances of length 17 in [redacted] UI5-TRANSLATION-TOOL/CLASS [redacted] EL-UI5-TRANSLATION.abap:1208

**Task 16 - Fehlende Best Practice für Berechtigungsprüfungen von Reports und Transaktionen**

created by [redacted] Apr 08 2021 09:14, last updated Sep 19 2021 21:49

Assignee: [redacted]

Status: Closed

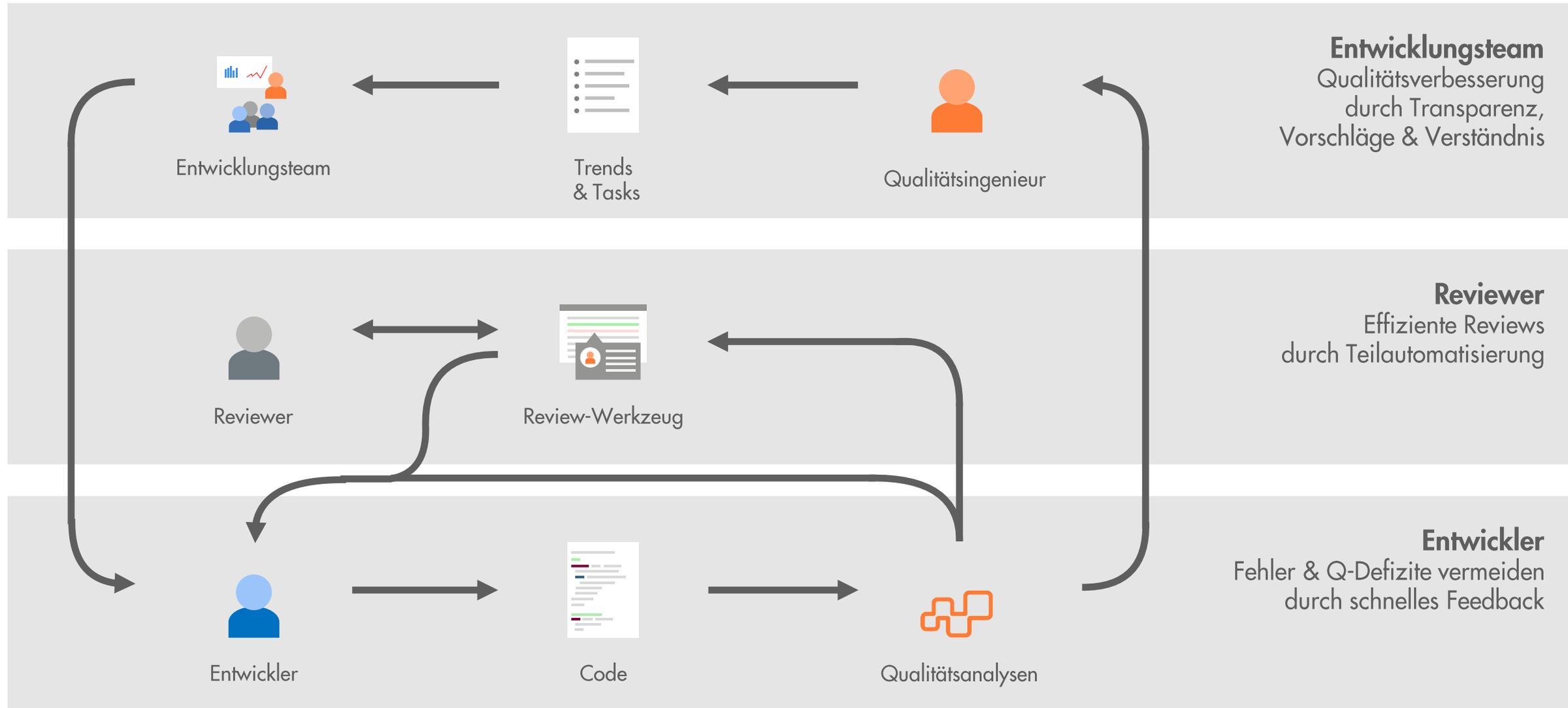
Resolution: Fixed

Description: Aufgrund des Feedbacks des [redacted] Teams wurden die Teamscale-Prüfungen für die Existenz von Berechtigungsprüfungen von Reports und Transaktionen (genauer: von Aufrufen einer Transaktion im Code mittels CALL TRANSACTION) deaktiviert. Deshalb fehlt aktuell eine Best Practice für die Berechtigungsprüfung für Reports und Transaktionen. Wir empfehlen, eine solche Best Practice zu definieren und mit Teamscale nachzuhalten, ob diese eingehalten wird.

Tags: 1. Retro, Security

- Verbesserungsvorschläge ermöglichen Einplanung und konkrete Verbesserung
- Diskussionen schaffen gemeinsames Qualitätsverständnis und Best Practices

# Continuous Quality Control-Feedbackschleifen



Was brought Continuous Quality Control?

Einführung stat.  
Codeanalyse-Werkzeug



→ Werkzeug hat i.d.R. nur einen kurzfristigen Effekt

Einführung stat.  
Codeanalyse-Werkzeug

Start Qualitäts-  
retrospektiven



→ Kombination aus Werkzeug und Retrospektiven  
hat i.d.R. einen nachhaltigen Effekt

# Nutzen der Klonanalyse für Munich Re

<input checked="" type="checkbox"/> All	7178
<input checked="" type="checkbox"/> Architecture	290
<input checked="" type="checkbox"/> Architecture Conformance	290
<input checked="" type="checkbox"/> Code Duplication	1608
<input checked="" type="checkbox"/> Code Clones	1608
<input checked="" type="checkbox"/> Comprehensibility	2180
<input checked="" type="checkbox"/> Bad Practice	1313
<input checked="" type="checkbox"/> Design Flaws	74
<input checked="" type="checkbox"/> Explicit Findings Management	21
<input checked="" type="checkbox"/> Formatting	225
<input checked="" type="checkbox"/> Modernization	3
<input checked="" type="checkbox"/> Naming	327
<input checked="" type="checkbox"/> Test Smells	76
<input checked="" type="checkbox"/> Unused Code	141
<input checked="" type="checkbox"/> Correctness	350
<input checked="" type="checkbox"/> API Misuse	42
<input checked="" type="checkbox"/> Concurrency	8
<input checked="" type="checkbox"/> Discouraged APIs	307
<input checked="" type="checkbox"/> Error-prone Practices	84
<input checked="" type="checkbox"/> Possible Bugs	108
<input checked="" type="checkbox"/> Resource Leaks	1
<input checked="" type="checkbox"/> Disabled Tests	19
<input checked="" type="checkbox"/> Disabled Tests	19
<input checked="" type="checkbox"/> Documentation	818
<input checked="" type="checkbox"/> Comment Completeness	762
<input checked="" type="checkbox"/> Malformed Comments	25
<input checked="" type="checkbox"/> Task Tags	31

$$500 \frac{PT}{Jahr}^*$$

Munich Re spart durch Einsatz von Clone Detection jährlich ca. 500 PT Aufwand für Fehlerbehebung

$$\text{Ersparnis Aufwand} = 6\%^*$$

Munich Re spart durch Einsatz von Clone Detection jährlich 6% Aufwand durch vermiedene Redundanz ein.

→ Continuous Quality Control hat großen Nutzen

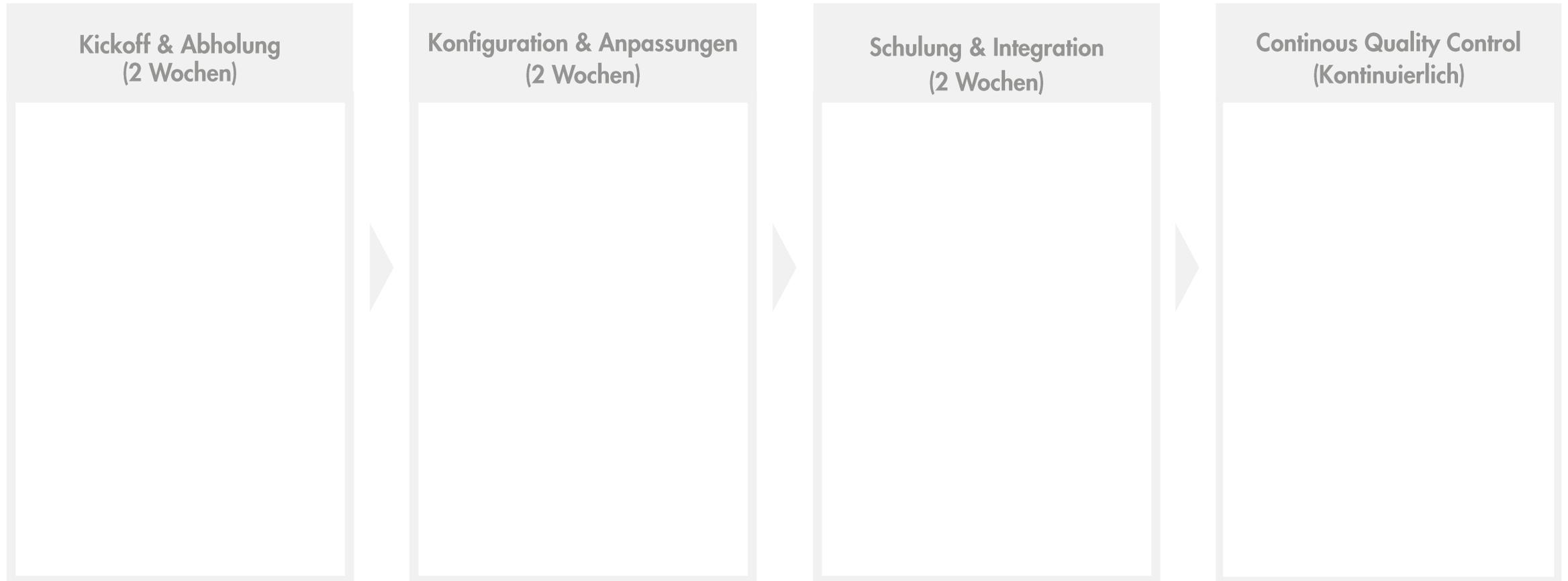
\* Übertragbarkeit muss geprüft werden; für ausführliche, konservative Herleitung siehe [CQSE-Workshop „Copy, Paste und dann ...?“](#)

Wie führt man Continuous Quality Control ein?

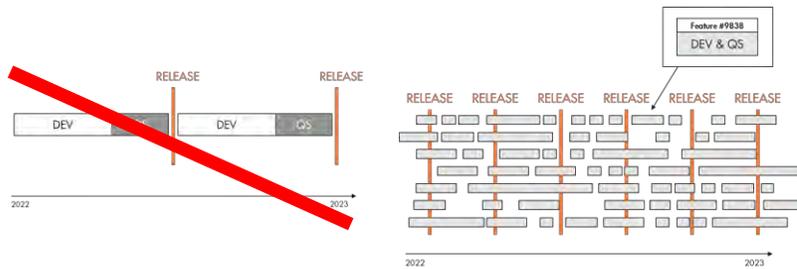


→ Technische, organisatorische und soziale Faktoren berücksichtigen

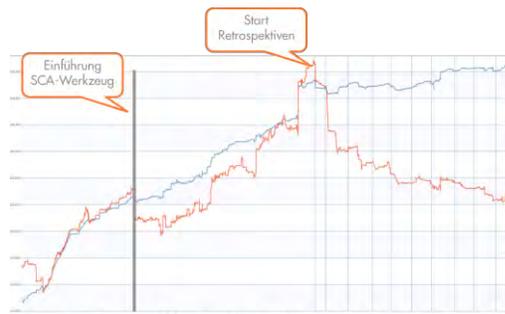
# Onboarding-Prozess



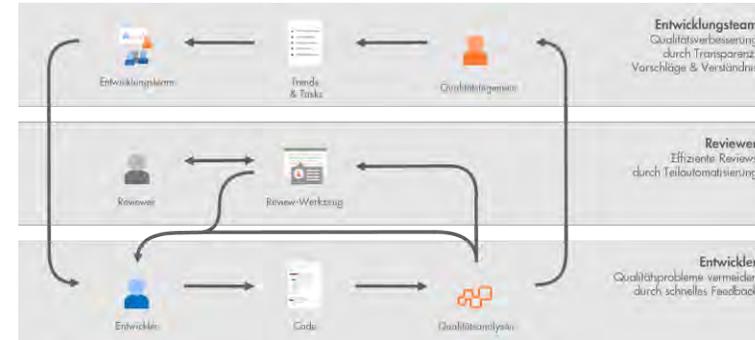
# Zusammenfassung



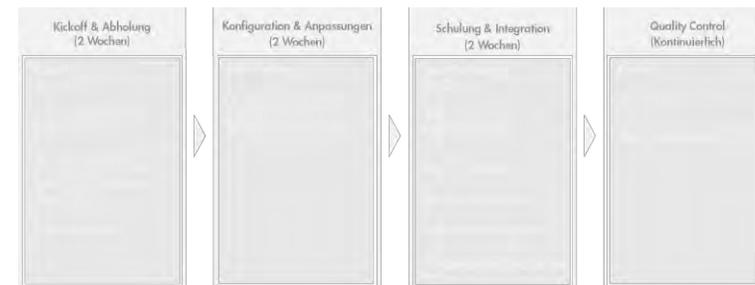
→ Releasefähigkeit erfordert jederzeitige Releasequalität und kontinuierliche QS



→ CQC hat großen Nutzen (auch bei gewachsenen Anwendungen)



→ CQC ermöglicht dies durch Feedbackschleifen für Entwickler, Reviewer und Entwicklungsteam



→ Einführung von CQC erfordert planvolles Vorgehen

# Continuous Quality Control

Qualität trotz immer kürzerer Releasezyklen

Dienstag, 11. November 2025

10:30 bis 12:00 Uhr

online und kostenlos



Dr. Tobias Röhm



Tobias Wiese

Jetzt anmelden: [tmscl.me/cqc2511-tjfs](https://tmscl.me/cqc2511-tjfs)

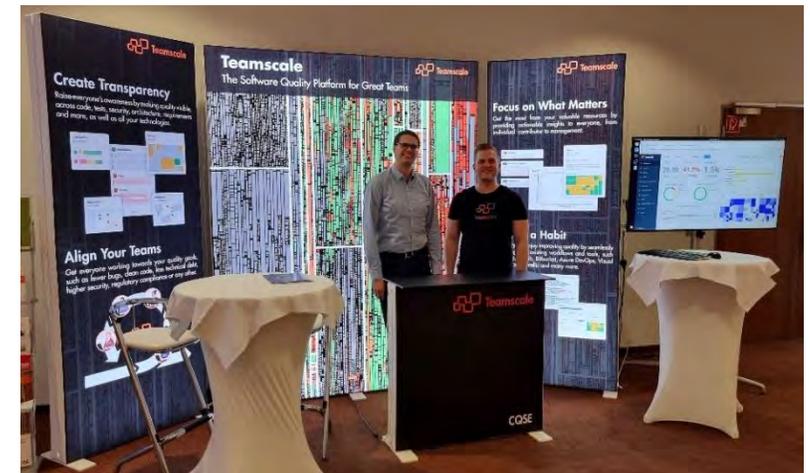
# Folien & Kontakt - Ich freue mich auf den Austausch 😊



Vortragsfolien:  
[tmscl.me/jfs\\_2025\\_releasezyklen](https://tmscl.me/jfs_2025_releasezyklen)



Dr. Tobias Röhm  
[roehm@cqse.eu](mailto:roehm@cqse.eu)  
[tmscl.me/coffee-tobias-roehm](https://tmscl.me/coffee-tobias-roehm)



CQSE-Stand