

# Vom Wiegen wird die Sau nicht fett

## Wirksame Qualitätsanalysen für PL/SQL

# Über Mich

## Forschung

- Clone Detection, Architekturanalyse, ...
- PC Mitglied von MSR, ICPC, ICSE, ...



## Beratung

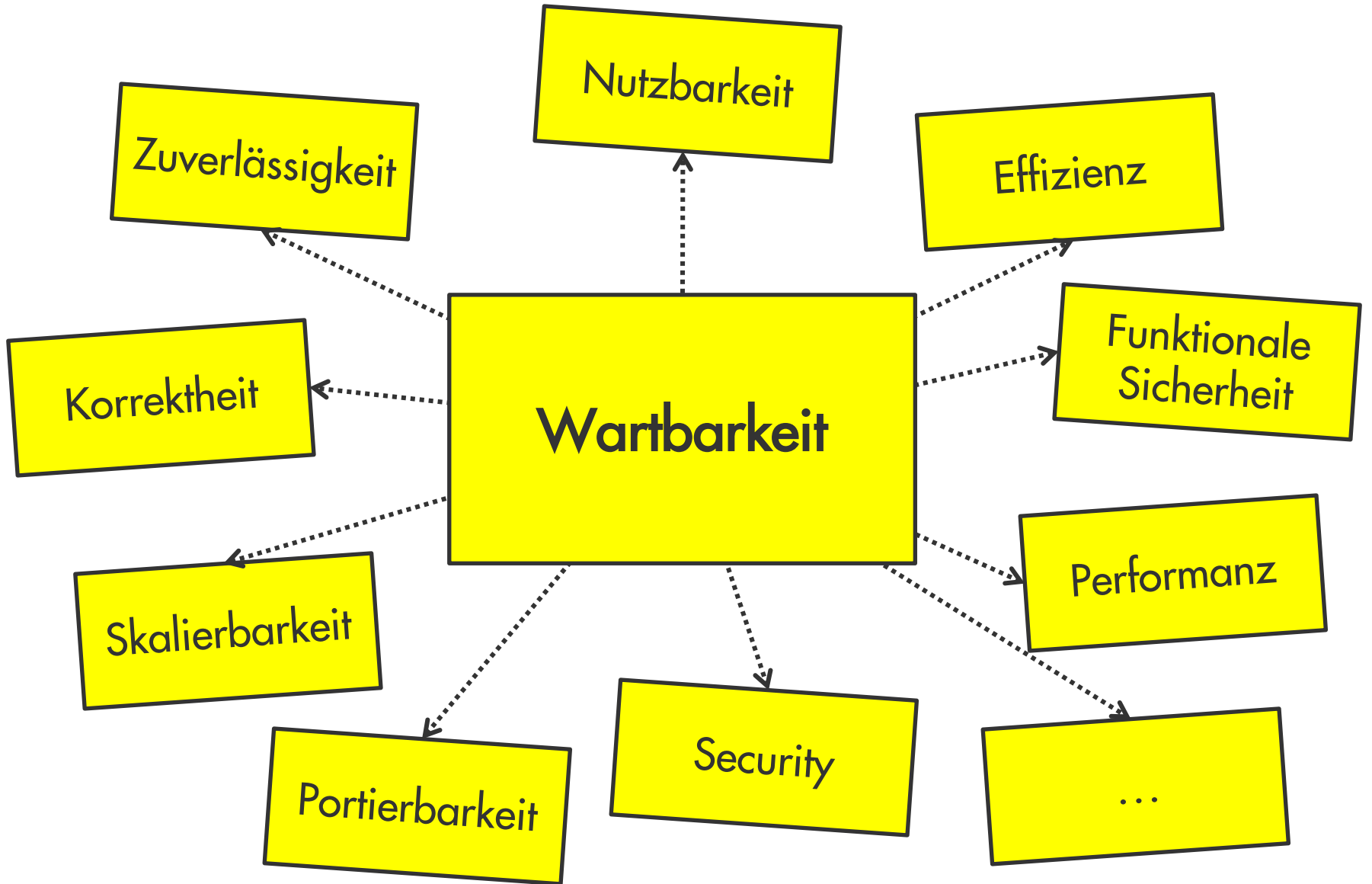
- Gründer
- Qualitäts-Bewertung & Qualitäts-Controlling



## Gesellschaft für Informatik

- Zum Junior-Fellow ernannt
- Erfahrungsaustausch Forschung <-> Praxis





```
// Utilities for arrays of elements
public String showElements(ModelElement[] elements, String nomsg) {
    boolean found = false;
    StringBuffer res = new StringBuffer();
    if (elements != null) {
        Index.getInstance().setCurrentRenderer(
            FlatReferenceRenderer.getInstance());
        for (int i = 0; i < elements.length; i++) {
            ModelElement el = elements[i];
            res.append(showElementLink(el)).append(HTML.LINE_BREAK);
            found = true;
        }
        Index.getInstance().resetCurrentRenderer();
    }
    if (!found && nomsg != null && nomsg.length() > 0) {
        res.append(HTML.italics(nomsg));
    }
    return res.toString();
}
```

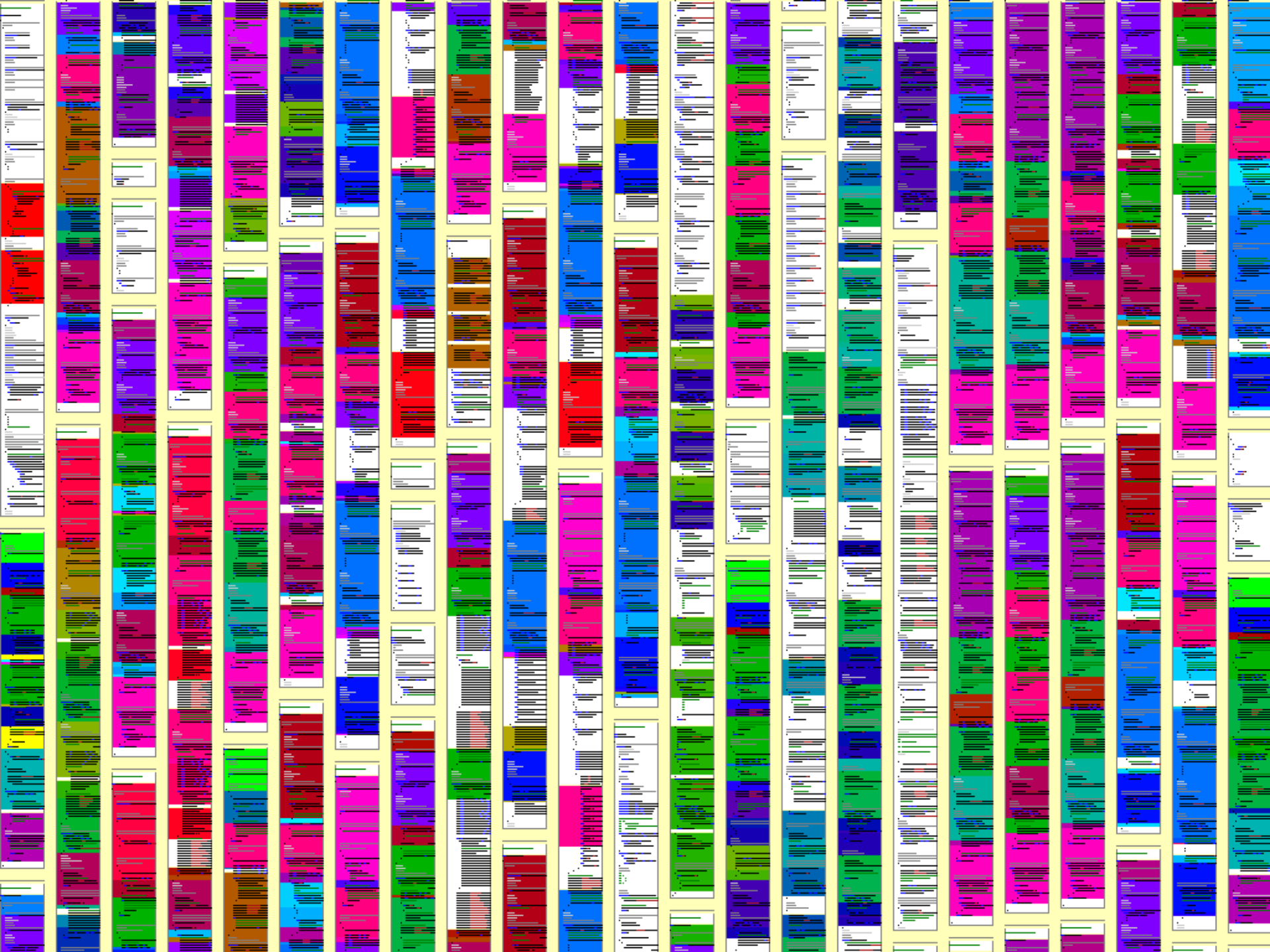
```
// Utilities for arrays of elements
public String showElements(ModelElement[] elements, String nomsg) {
    boolean found = false;
    StringBuffer res = new StringBuffer();
    if (elements != null) {
        Index.getInstance().setCurrentRenderer(
            FlatReferenceRenderer.getInstance());
        for (int i = 0; i < elements.length; i++) {
            ModelElement el = elements[i];
            res.append(showElementLink(el)).append(HTML.LINE_BREAK);
            found = true;
        }
        Index.getInstance().resetCurrentRenderer();
    }
    if (!found && nomsg.length() > 0) {
        res.append(HTML.italics(nomsg));
    }
    return res.toString();
}
```

```
// Utilities for arrays of elements
public String showElements(ModelElement[] elements, String nomsg) {
    boolean found = false;
    StringBuffer res = new StringBuffer();
    if (elements != null) {
        Index.getInstance().setCurrentRenderer(
            FlatReferenceRenderer.getInstance());
        for (int i = 0; i < elements.length; i++) {
            ModelElement el = elements[i];
            res.append(showElementLink(el)).append(HTML.LINE_BREAK);
            found = true;
        }
        Index.getInstance().resetCurrentRenderer();
    }
    if (!found && nomsg != null && nomsg.length() > 0) {
        res.append(HTML.italics(nomsg));
    }
    return res.toString();
}
```











## Studie

Munich Re  

- Über 100 Fehler in produktiver Software



- 52% aller ungewollten Unterschiede fehlerhaft

Juergens, Deissenboeck et al: *Do Code Clones Matter?* ICSE 2009



## Example: Unclosed Cursors

### Cursor 'Get\_Schedule\_Id' opened but not closed in the same procedure

Code Anomalies / PL/SQL Checks

An opened cursor is not closed within the same procedure.

in SchedReleaser.plsql:2756



```
2751      OPEN Get_Old_Schedules;
2752      FETCH Get_Old_Schedules INTO current_;
2753      CLOSE Get_Old_Schedules;
2754
2755      IF (current_ > keep_) THEN
2756      OPEN Get_Schedule_Id;
2757      FETCH Get_Schedule_Id INTO del_doc_no_, del_schedule_no_;
2758      WHILE (current_ > keep_ AND Get_Schedule_Id%FOUND) LOOP
2759          Delete_Single_Schedule___ (customer_no_, ship_addr_no_, del_doc_no_
2760          current_ := current_ - 1;
2761      FETCH Get_Schedule_Id INTO del_doc_no_, del_schedule_no_;
```

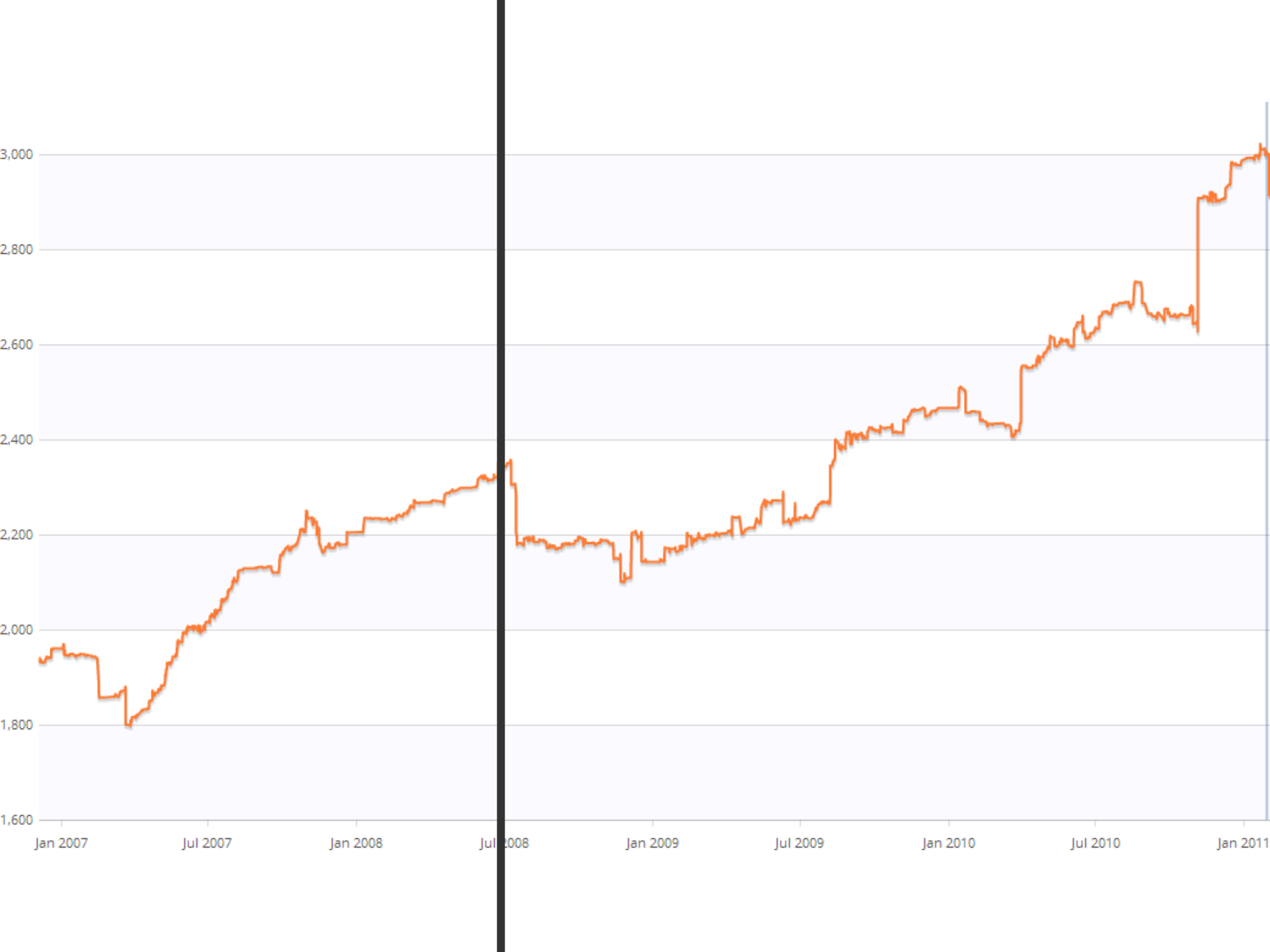
## Example: Unclosed Cursors

### Cursor 'Get\_Schedule\_Id' opened but not closed in the same procedure

[Code Anomalies / PL/SQL Checks](#)

An opened cursor is not closed within the same procedure.

```
2755     IF (current_ > keep_) THEN
2756         OPEN Get_Schedule_Id;
2757         FETCH Get_Schedule_Id INTO del_doc_no_, del_schedule_no_;
2758         WHILE (current_ > keep_ AND Get_Schedule_Id%FOUND) LOOP
2759             Delete_Single_Schedule___ (customer_no_, ship_addr_no_, del_doc_no_,
2760             current_ := current_ - 1;
2761             FETCH Get_Schedule_Id INTO del_doc_no_, del_schedule_no_;
2762         END LOOP;
2763     END IF;
2764 END IF;
2765 END Delete_Old_Schedules___;
2766
```





+



=





BAKERS BOX

CP - Carls milk Note  
33250

soft cloths

For use in Plastic Food Containers, Lamp Shades, Games, Stereo Speakers, Etc.

Sm Files  
base  
K4

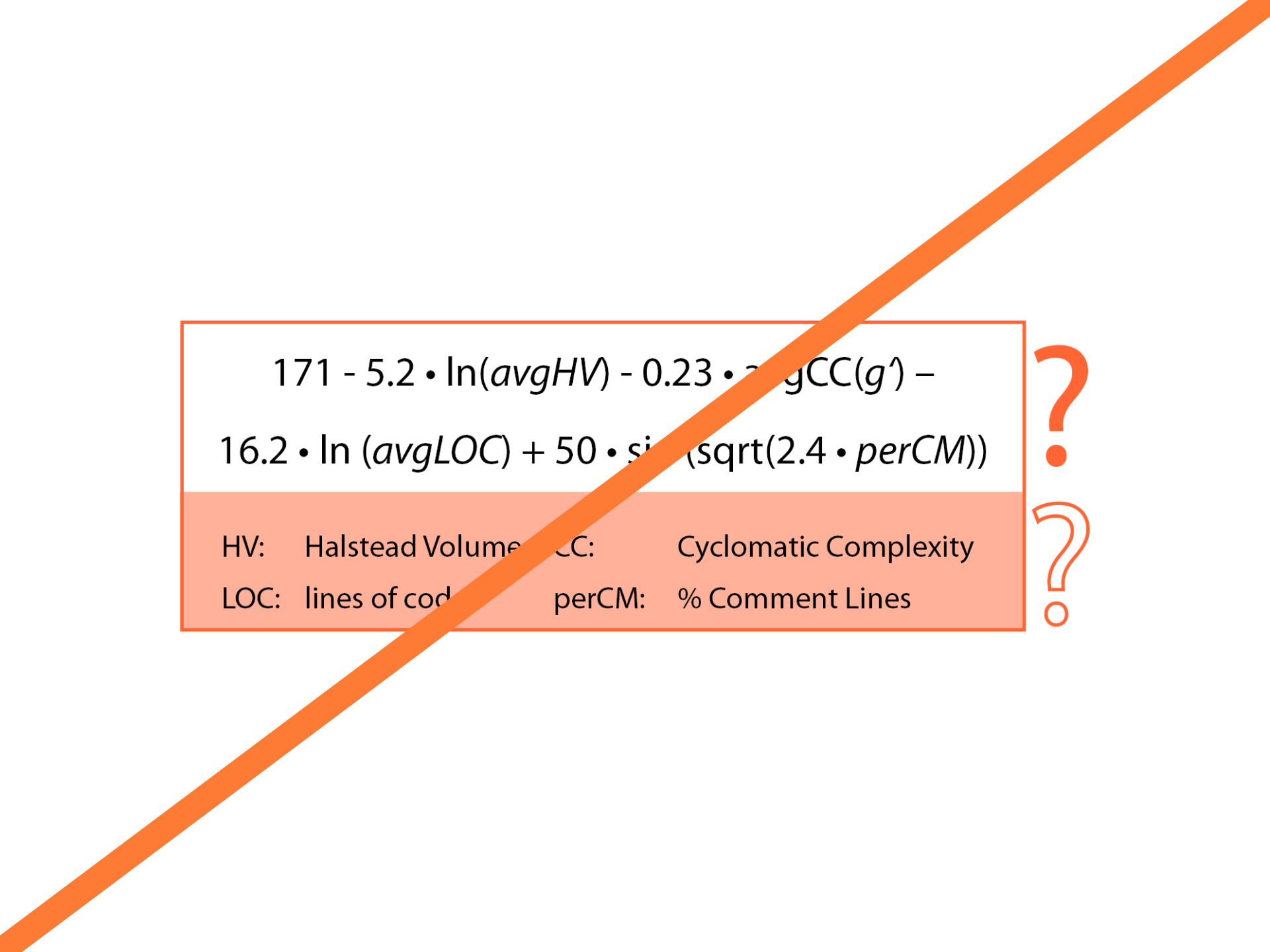
SOLE F80







WARNING  
BRIGHT LIGHT  
- Do not stare into light beam  
- Do not view at close range  
Failure to do so will cause  
permanent eye damage

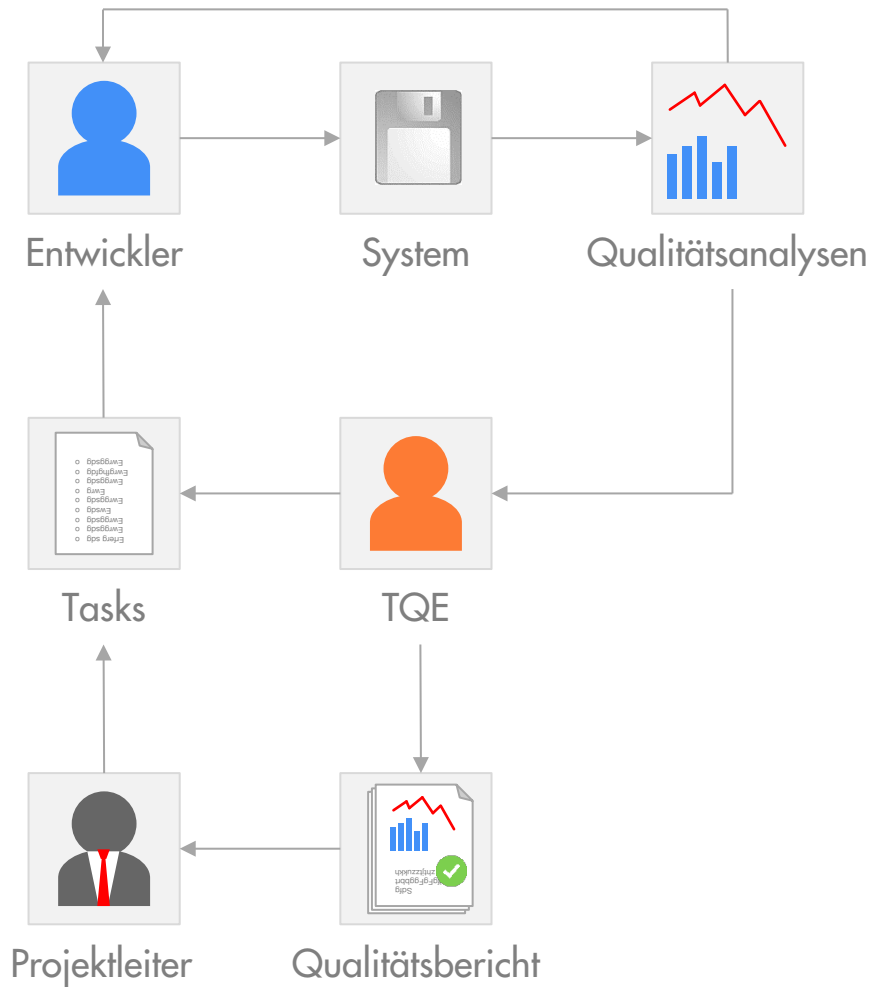

$$171 - 5.2 \cdot \ln(\text{avgHV}) - 0.23 \cdot \ln(\text{avgCC}(g)) - \\ 16.2 \cdot \ln(\text{avgLOC}) + 50 \cdot \sin(\text{sqrt}(2.4 \cdot \text{perCM}))$$

HV: Halstead Volume      CC: Cyclomatic Complexity  
LOC: lines of code      perCM: % Comment Lines



# Best Practice: Zielorientierte Analysen

- Objektiv
- Auswirkungen von Code-Änderungen verständlich
- Actionable
- Nachvollziehbarer Zusammenhang zu Wartungstätigkeit



## Schnelles Feedback

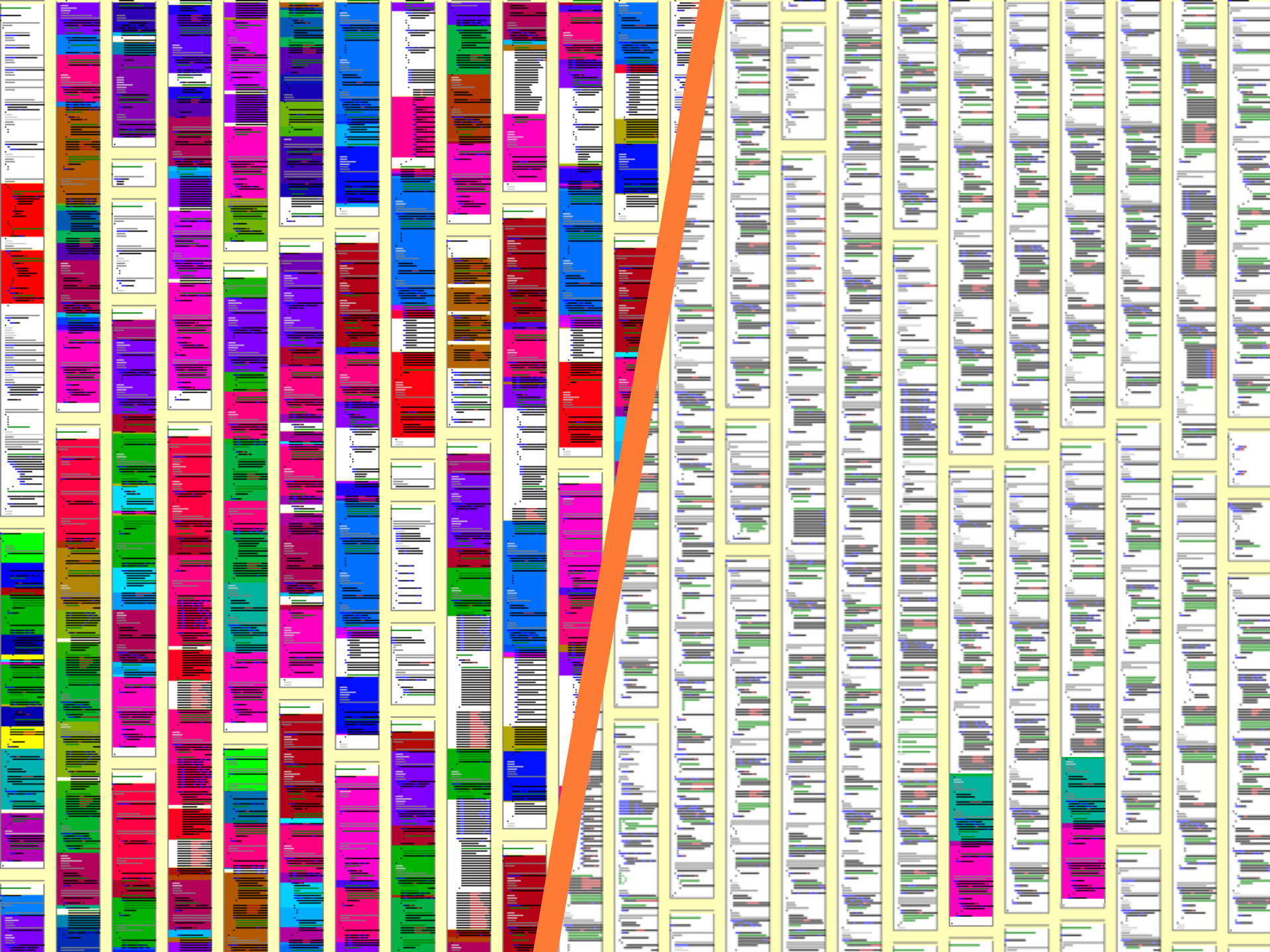
Unmittelbare Reaktion auf  
Qualitätsdefizite

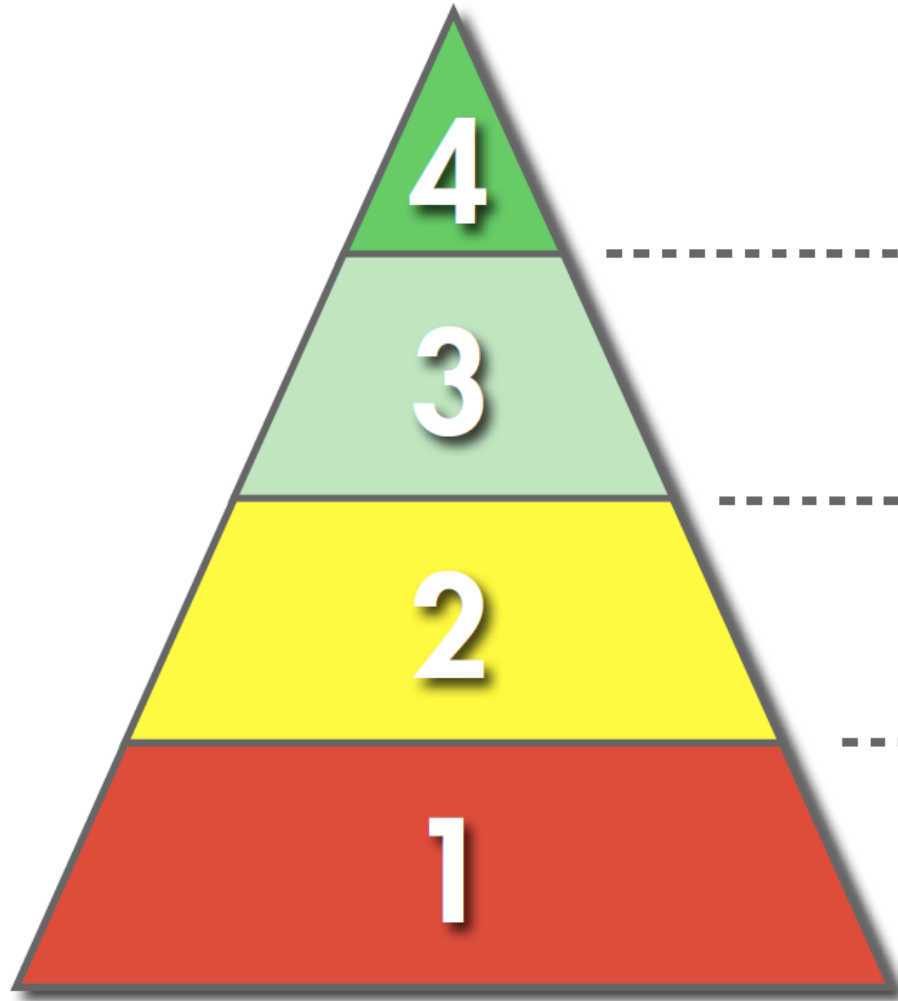
## Konkrete Verbesserungen

Definition von Tasks

## Hohe Transparenz

»Features vs. Qualität«-  
Entscheidungen



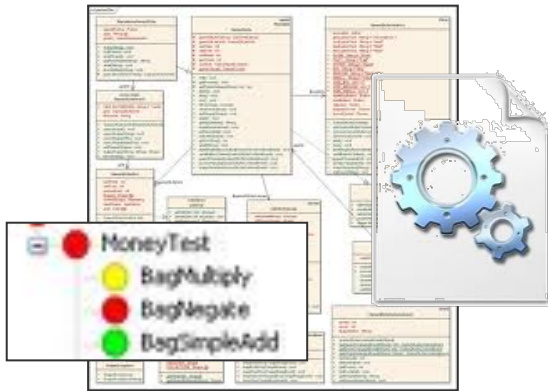


**Keine Defizite**

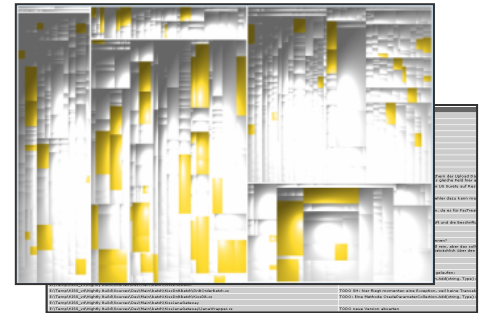
**Keine Defizite in geändertem Code**

**Keine neuen Defizite**

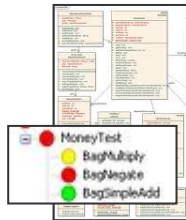
**Egal**



conQAT



Baseli



Latest V



Laufzeit?

Findings-Tracking?

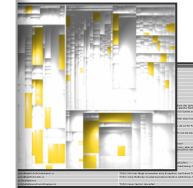
Feature-Branches?

Archivierung?

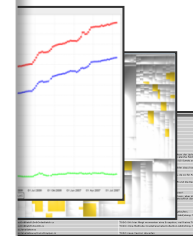
Analyseänderungen?

...?

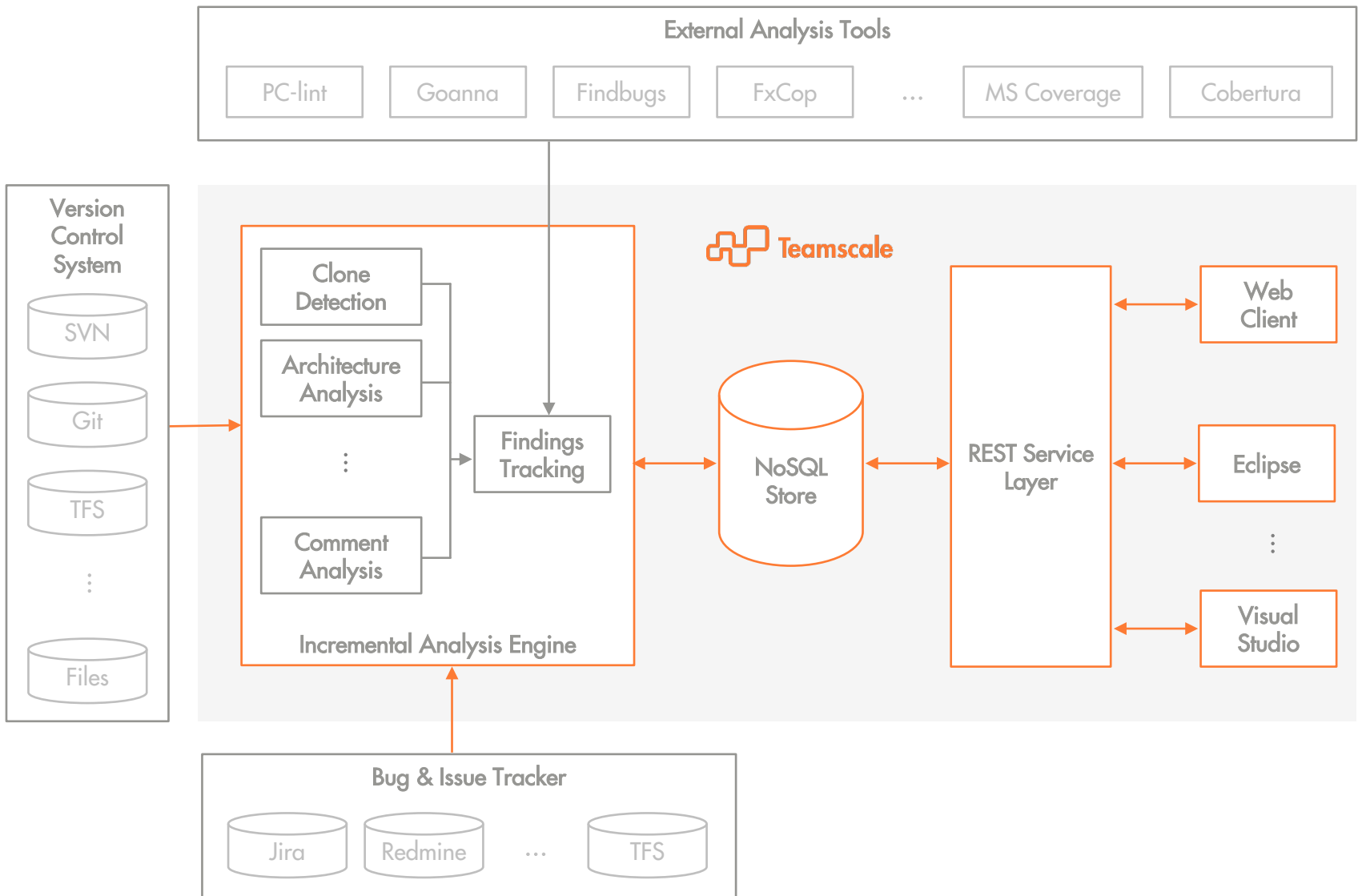
aseli  
ard



nboard  
l Delta









### ACT-1270 Fixing Inconsistent handling of serializable process variables

by [Victoria King](#) in revision [e1aa41b4b133d269980fff3f81d008da8f21a109](#) (git)

Jun 29 2012

16:05

changed 2 files

**-4** findings



### ACT-1258 Merging Pablo's work into trunk

by [Jacob Nelson](#) in revision [9e664a1f0676cedcbe03415a253e8c3e4a58944c](#) (git)

Jun 29 2012

14:41

added 3 files, changed 2 files

**-1** findings



### Fix for ACT-1059: Task#setDelegationState(DelegationState) was not saved in database

by [Michael Harris](#) in revision [1f48dcad04bc4a621e60af047fb121ae161bca30](#) (git)

Jun 28 2012

21:45

changed 3 files

**+2** findings



### ACT-991 Removed user id from exception message in order not to leak sensitive information

by [Michael Harris](#) in revision [e9a09424e6309c854c44ac5d08740a8ffb082fc9](#) (git)

Jun 28 2012

15:26

changed 2 files

**-2** findings



Project: cqse All With Branches Branch: master Timetravel: All Commits.



**Merge remote-tracking branch 'origin/cr/9905'** Oct 20 2016 14:34  
 by [Lars Heinemann](#) in revision [3e9579a715a156d2e629fcb02a44f3c1d3ad0b5b](#) in branch master  
 (repository1)  
 Merged from [cr/9905](#)

Files: 3 changed  
 Findings: 0



**Merge branch 'master' into cr/10552\_branch\_coverage** Oct 20 2016 14:24  
 by [Benjamin Hummel](#) in revision [25692692f99563e104e167204f860ceec5fa92aa](#) in branch cr/10552\_branch\_coverage (repository1)  
 Merged from [master](#)

Files: 13 added, 152 changed, 3 deleted  
 Findings: 59 8 69



**CR#9766: reviewed** Oct 20 2016 14:22  
 by [Lars Heinemann](#) in revision [ed133b7b146e1d5051c19995a0be058f48a0c0b9](#) in branch master  
 (repository1)

Files: 2 changed  
 Findings: 8



**CR#10552 GREEN (with Thomas)** Oct 20 2016 14:19  
 by [Benjamin Hummel](#) in revision [31b76706e1d6a7600762d5881f5850ef8da06bad](#) in branch cr/10552\_branch\_coverage (repository1)

Files: 7 changed  
 Findings: 11



**CR#0: trying to fix test** Oct 20 2016 13:24  
 by [Lars Heinemann](#) in revision [87d74fd92ba06f9b3f43be63a641008bdf8ed0a1](#) in branch master  
 (repository1)

Files: 1 changed  
 Findings: 0



fixed: latest change is no longer lost when assigning entry to a keyword group while it is being edited

by jzieren in revision [e0ca9a51b50c8b01f579f4eef79028bff6c34028](#) (git)

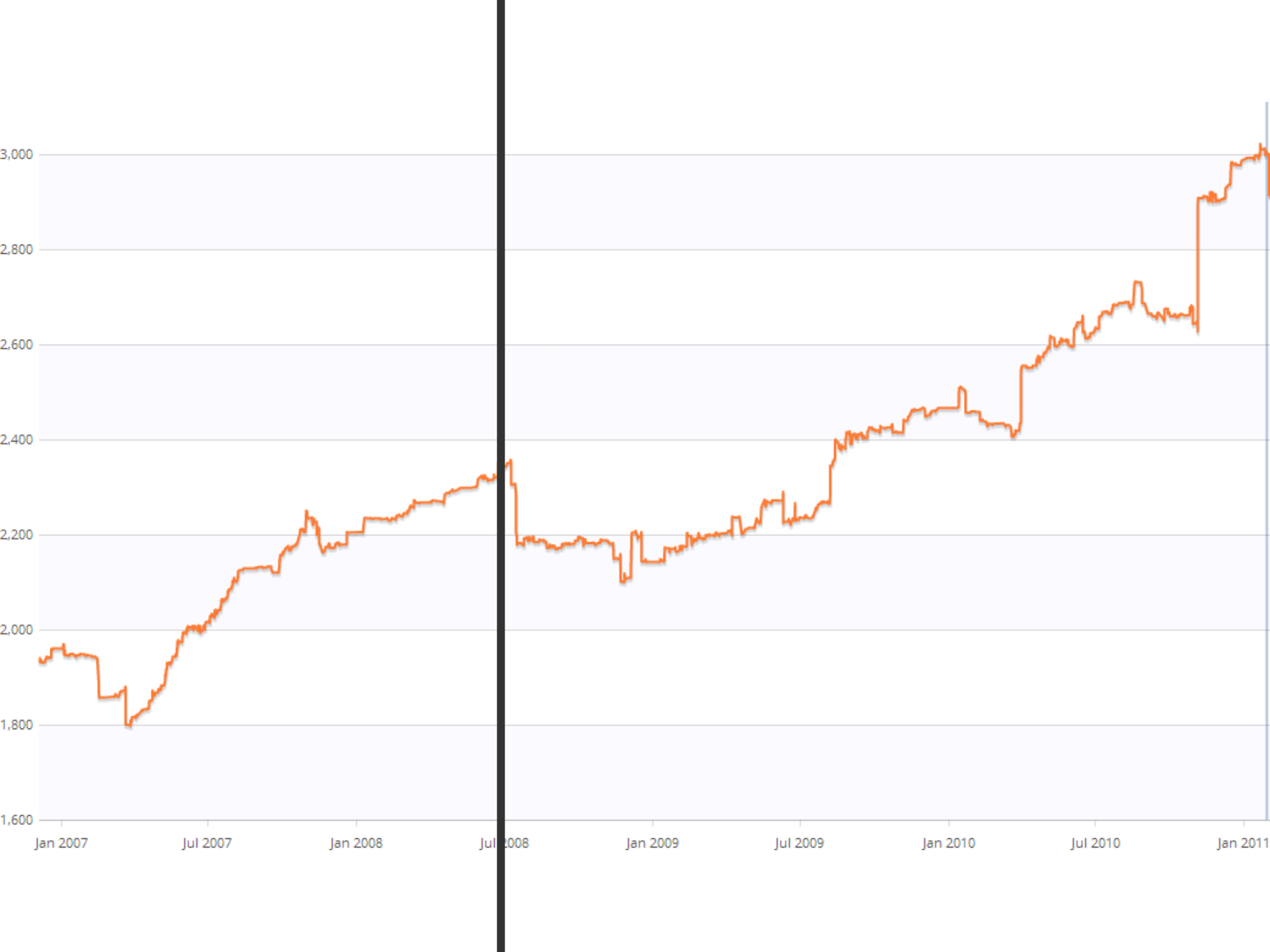
May 26 2005  
15:58

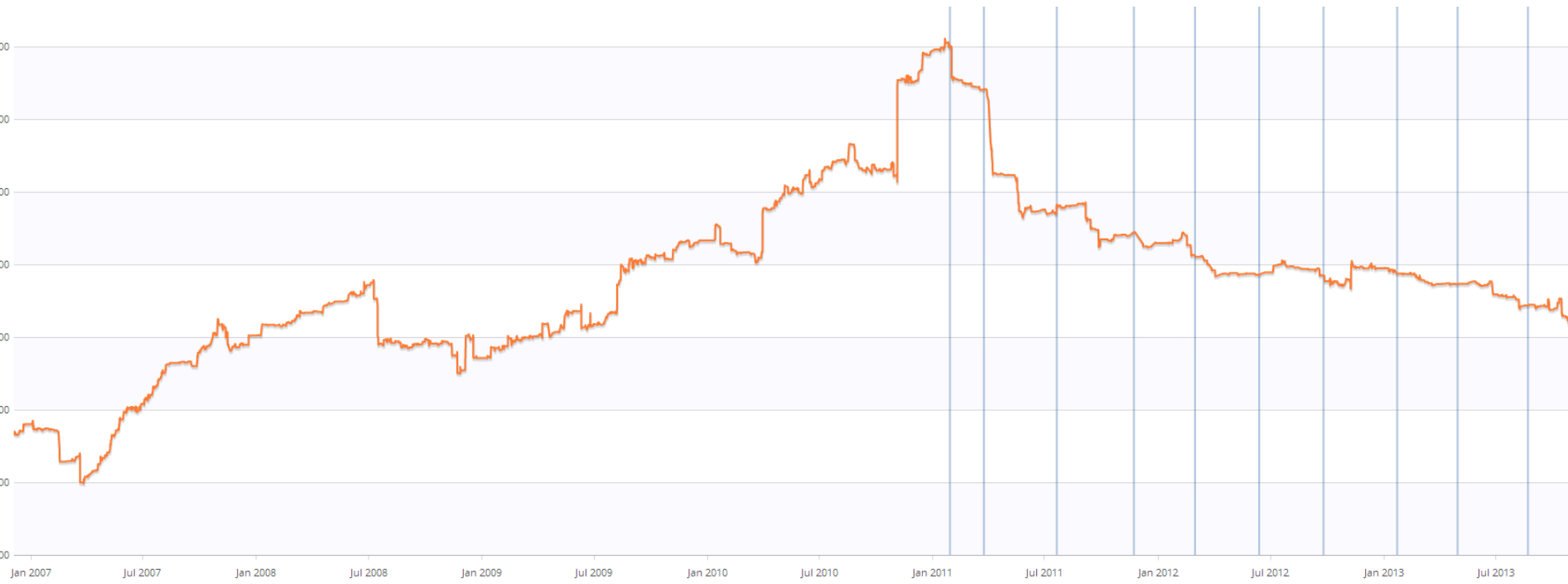
**1** alerts:

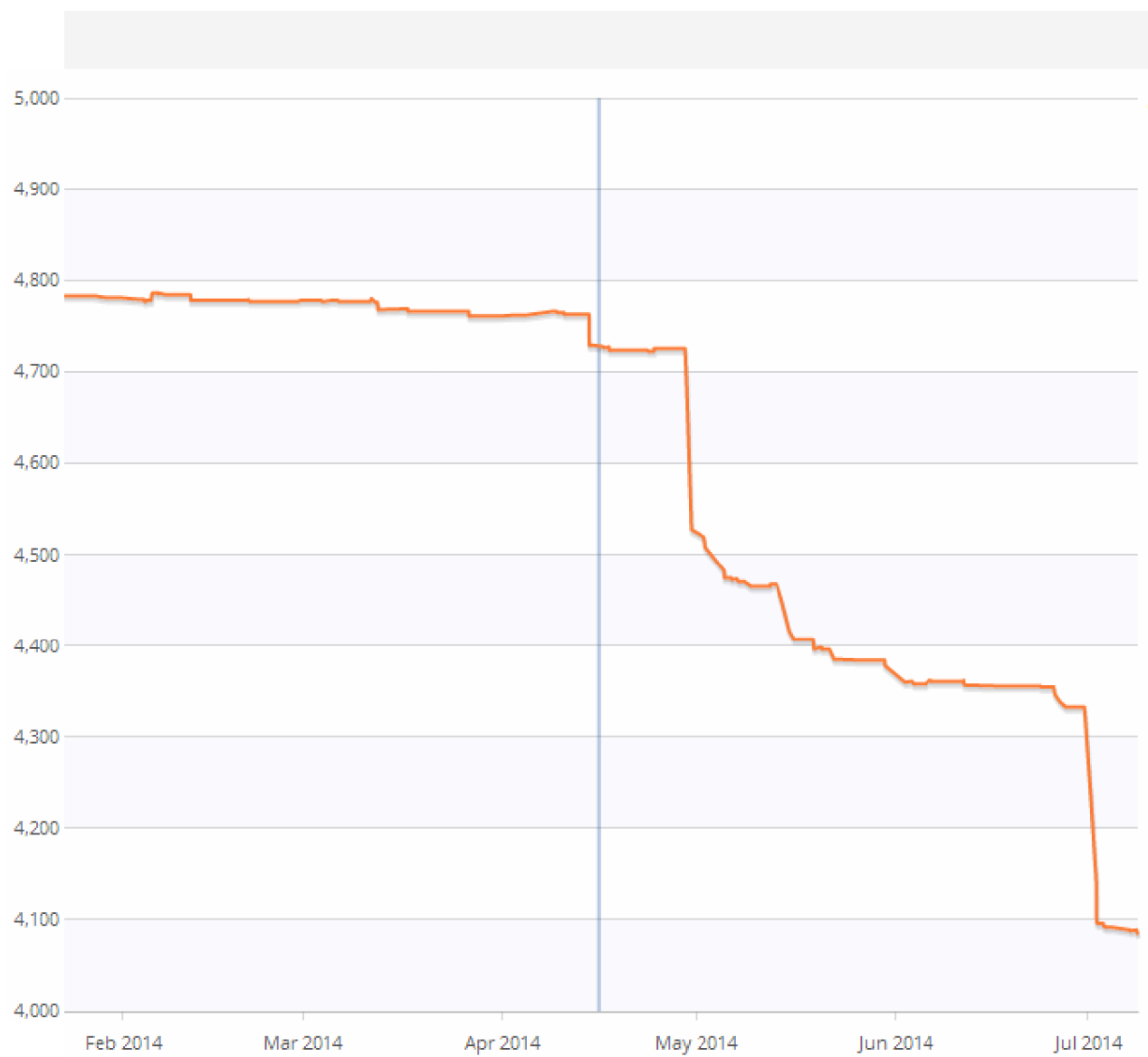
Message	Context
Found potential inconsistent clone change in RightClickMenu.java	<a href="#">[Broken clone]</a> <a href="#">[Old clone finding]</a> <a href="#">[Code change]</a>

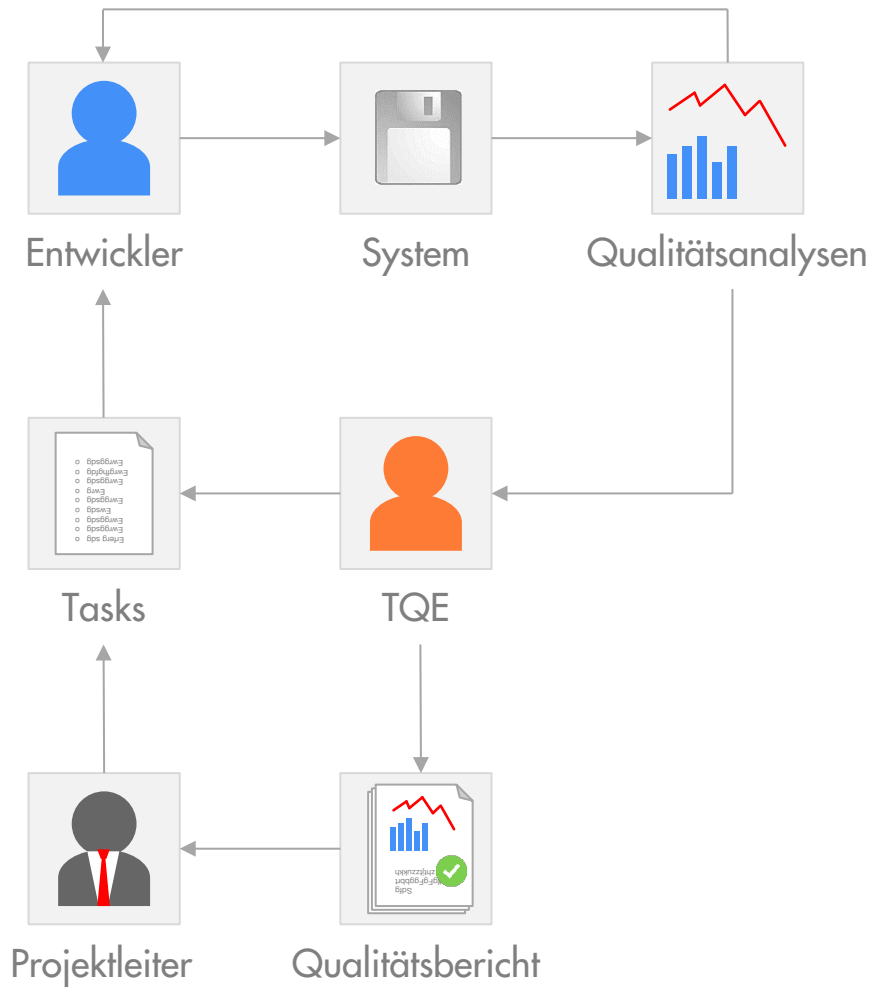
**2** removed findings:

Message	Location	Finding Group
<a href="#">Clone with 2 instances of length 10</a>	<a href="#">src/java/net/sf/.../RightClickMenu.java:366-380</a>	Code Duplication / Cloning
<a href="#">Clone with 2 instances of length 10</a>	<a href="#">src/java/net/sf/.../RightClickMenu.java:340-354</a>	Code Duplication / Cloning









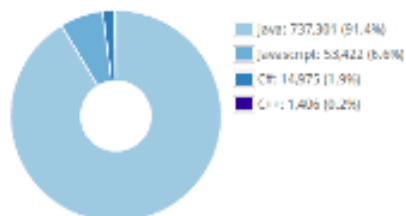
**Schnelles Feedback**  
 Unmittelbare Reaktion auf  
 Qualitätsdefizite

**Konkrete Verbesserungen**  
 Definition von Tasks

**Hohe Transparenz**  
 »Features vs. Qualität«-  
 Entscheidungen



LOC Distribution for cqse-all



Clone Coverage for cqse-all



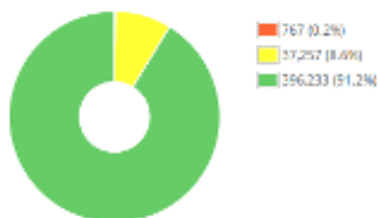
Findings Density for cqse-all



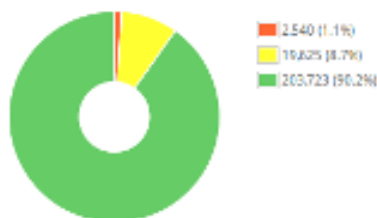
Review Rating for cqse-all



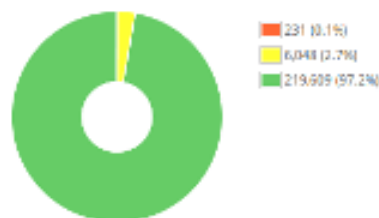
File Length for cqse-all



Method Length for cqse-all



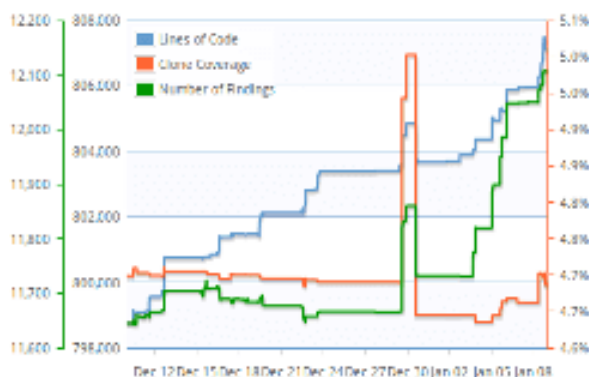
Nesting Depth for cqse-all



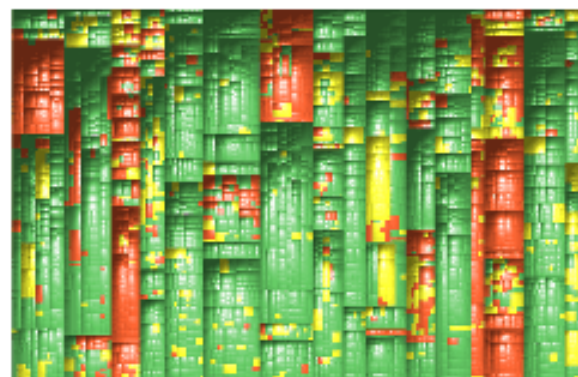
Comment Completeness for cqse-all



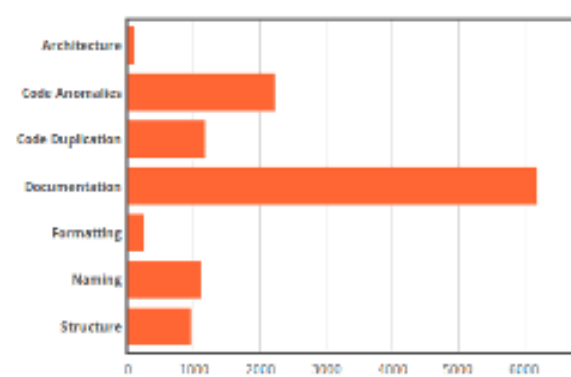
30 Day Trend for cqse-all



Rating Treemap for cqse-all



Findings Summary Bar Chart for cqse-all

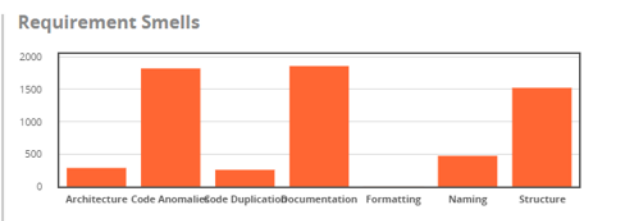


# Sustainability

### Requirements

**How precise are the requirements?**

- Number of requirement smells per number of requirements ● ↑
- Ratio of incomplete requirements ● →



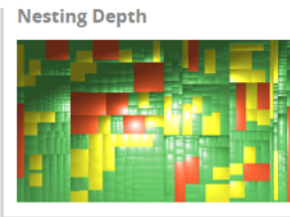
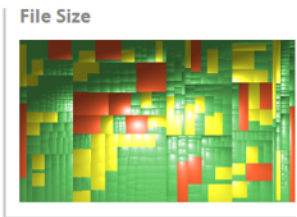
### Architecture and Design

**Do architecture violations exist?**

- Number of critical AUTOSAR violations ● ↓
- Number of architecture violations ● →

**Does the architecture comply with AUTOSAR?**

- Number of findings in AUTOSAR models ● →



### Implementation

**Is the Code MISRA compliant?**

- Number of MISRA violations ● ↑
- MISRA violations per kLOC ● ↑

**Does the code contain redundancies?**

- Number of code clones ● →

**Is the code in compliance with the coding guidelines?**

- Number of coding guideline violations ● →

### Cloning

**19.7k**  
**+648**



### Complexity

**17**

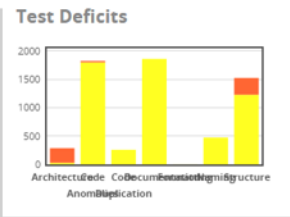
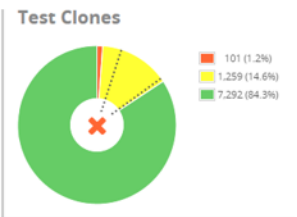
### Test

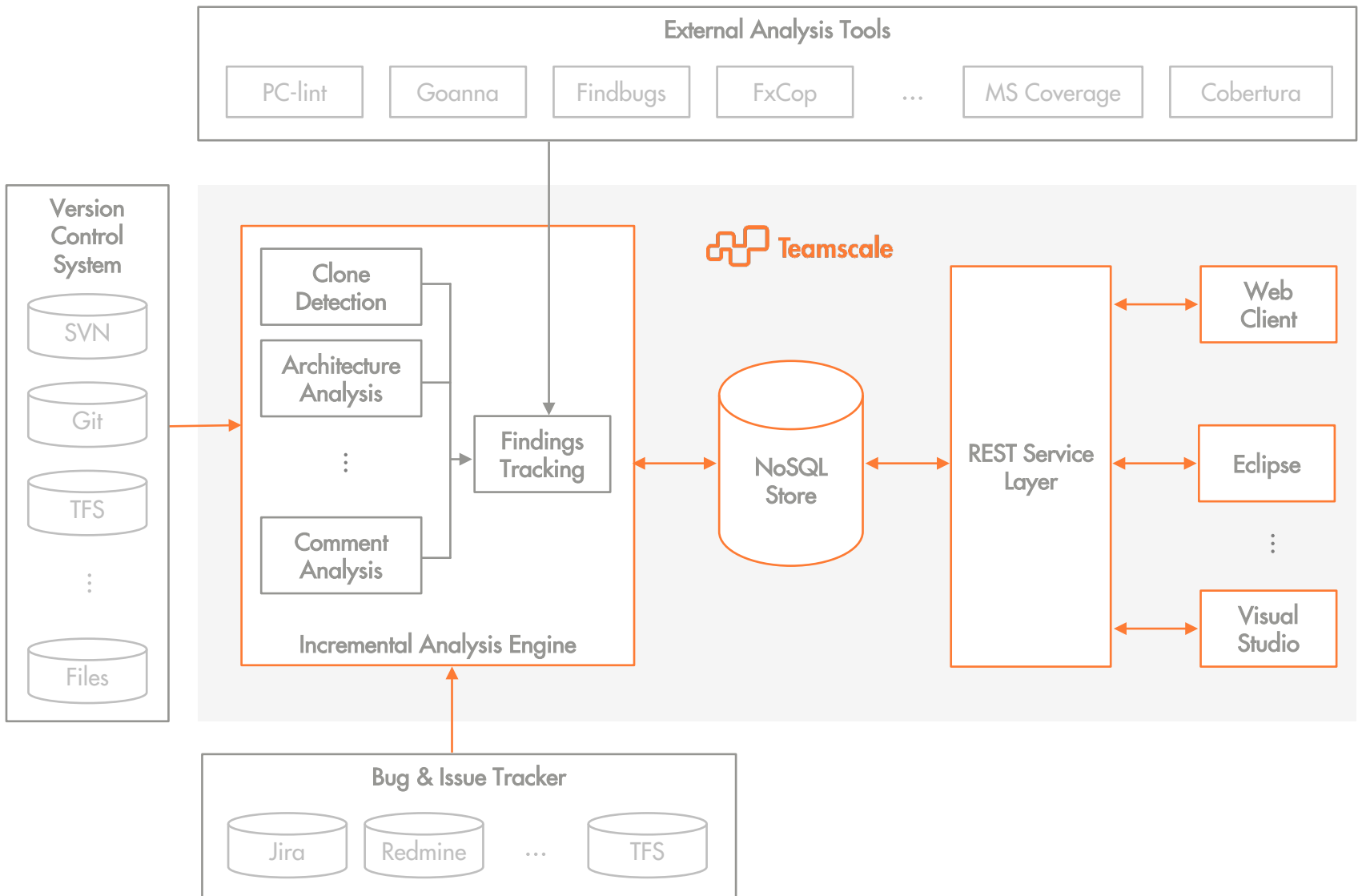
**What portion of the code is covered by tests?**

- Unit-Tests C0/C1 Code Coverage ● ↑
- SIL-Tests C0/C1 Code Coverage ● ↑
- HIL-Tests C0/C1 Code Coverage ● ↑

**Does the test code contain redundancies?**

- Number of code clones ● →







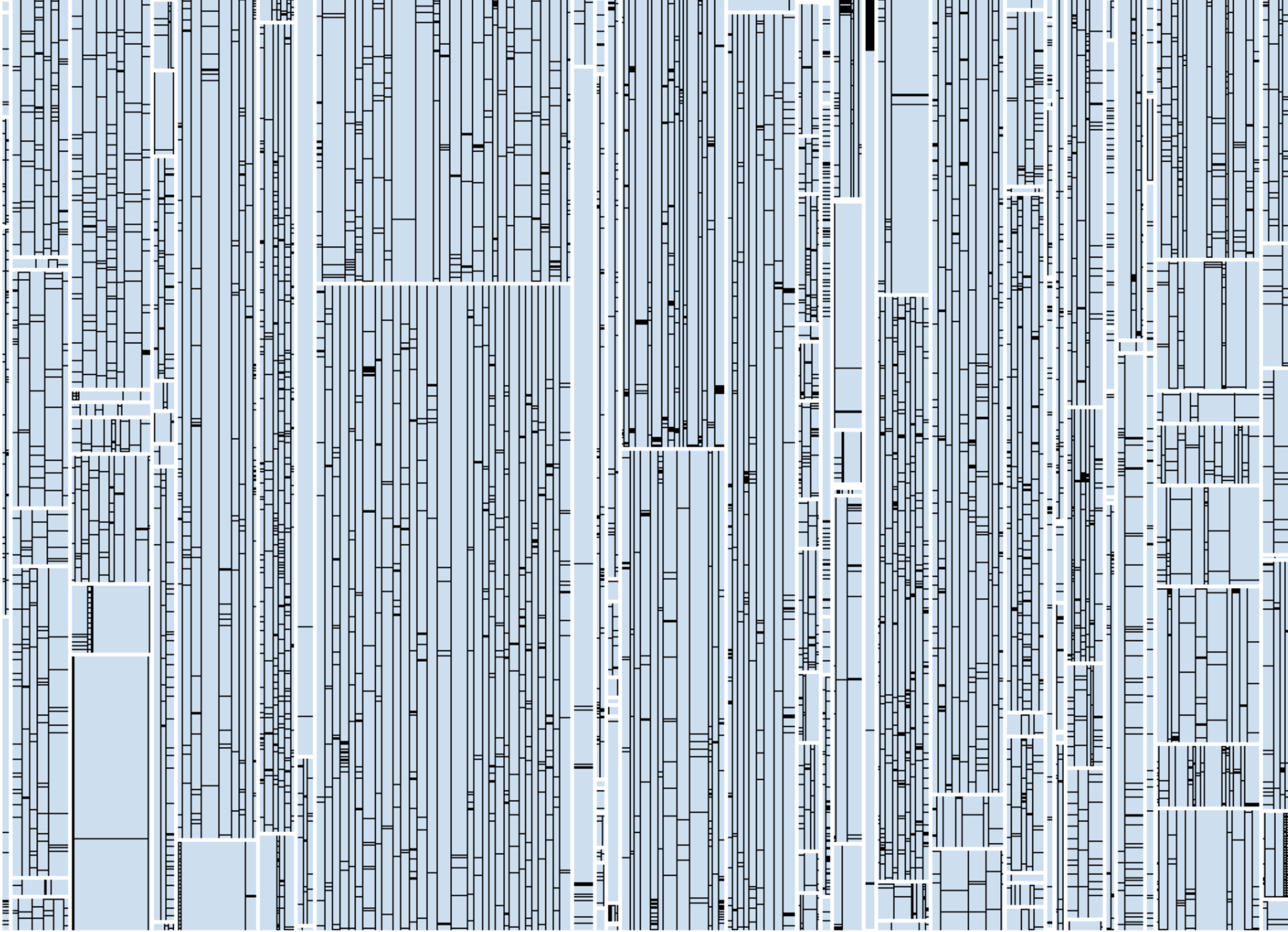
GUI.Dialogs

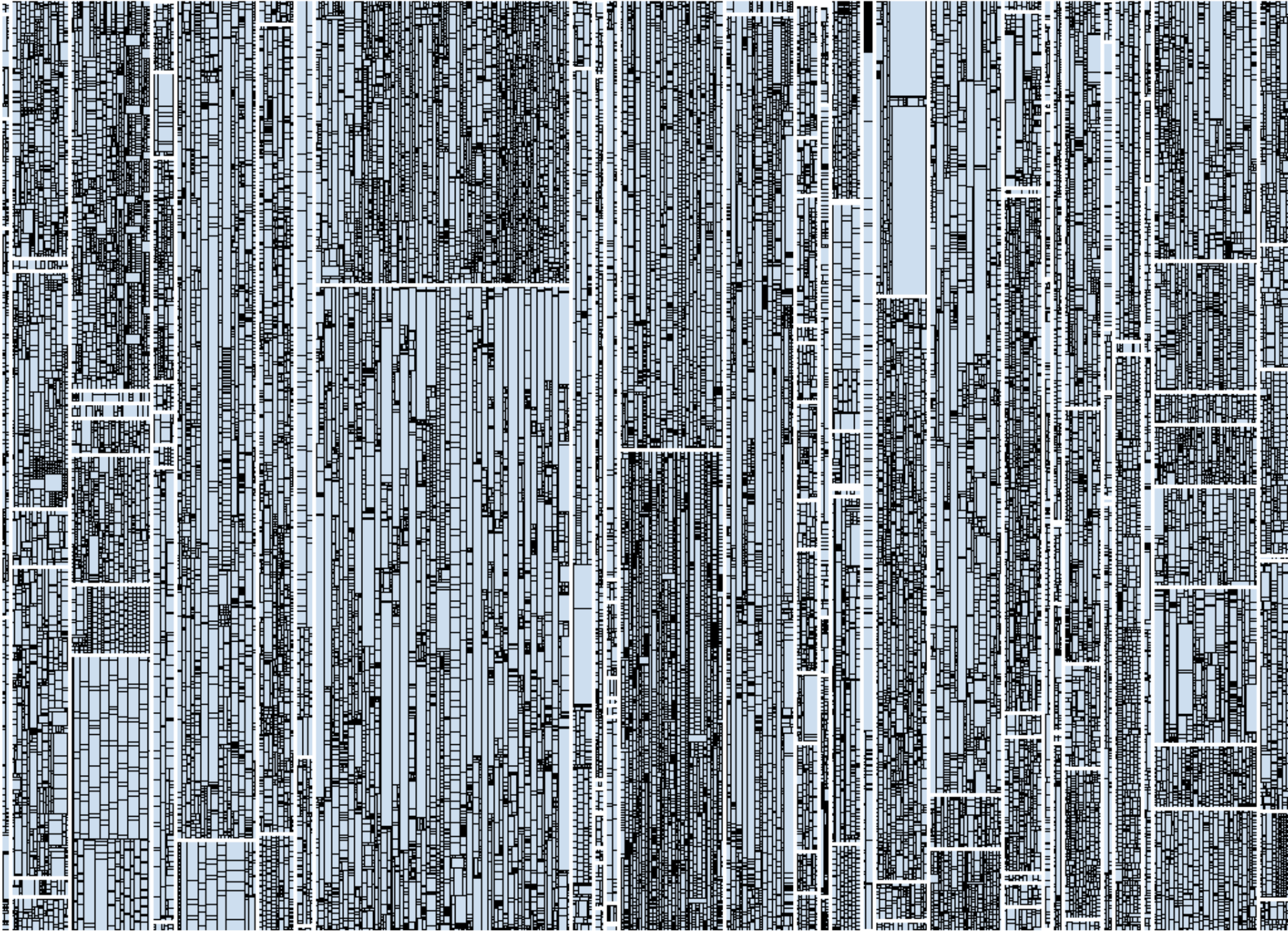
UI Controls

GUI.Base

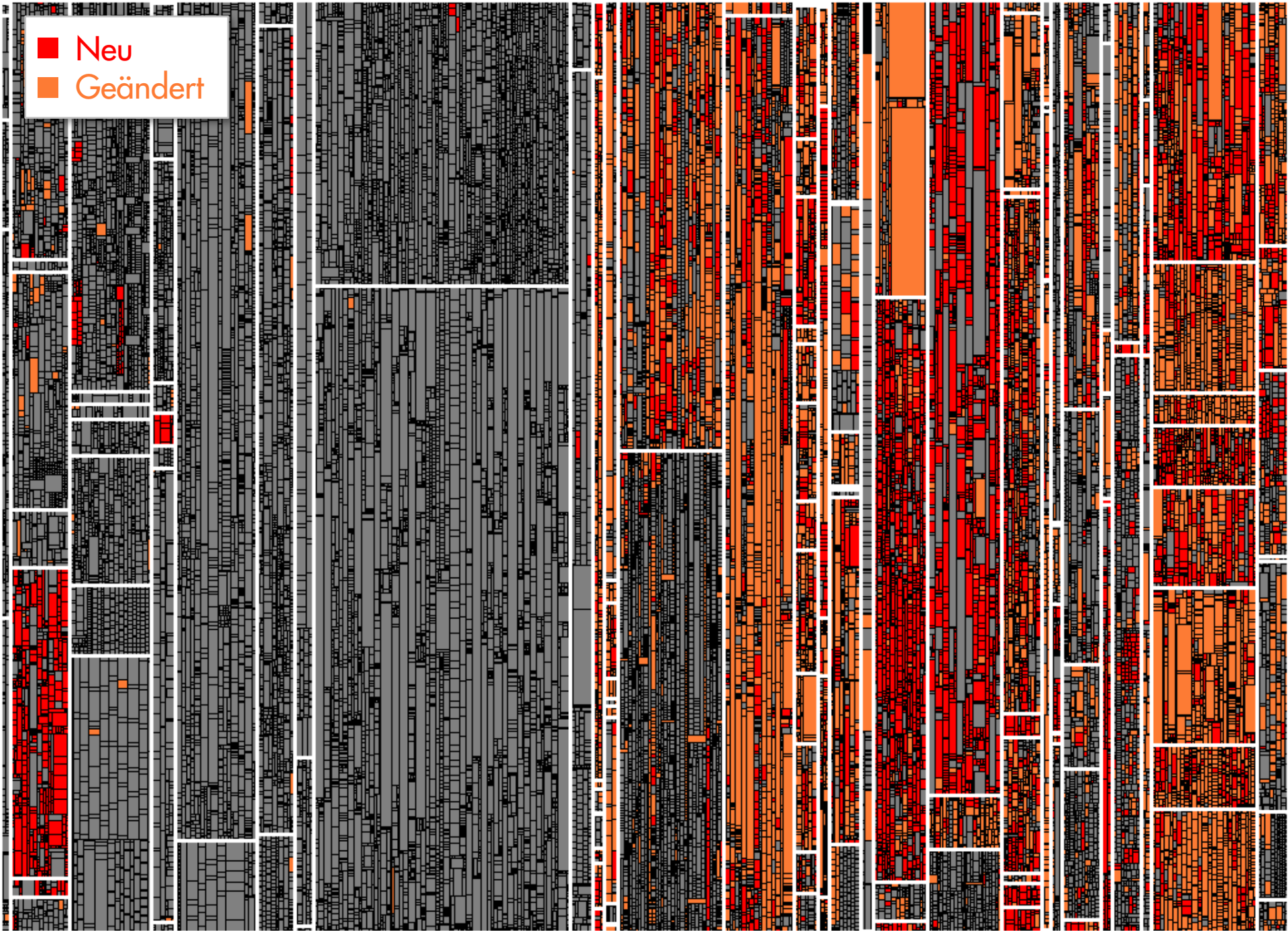
Authentication

Data Validation



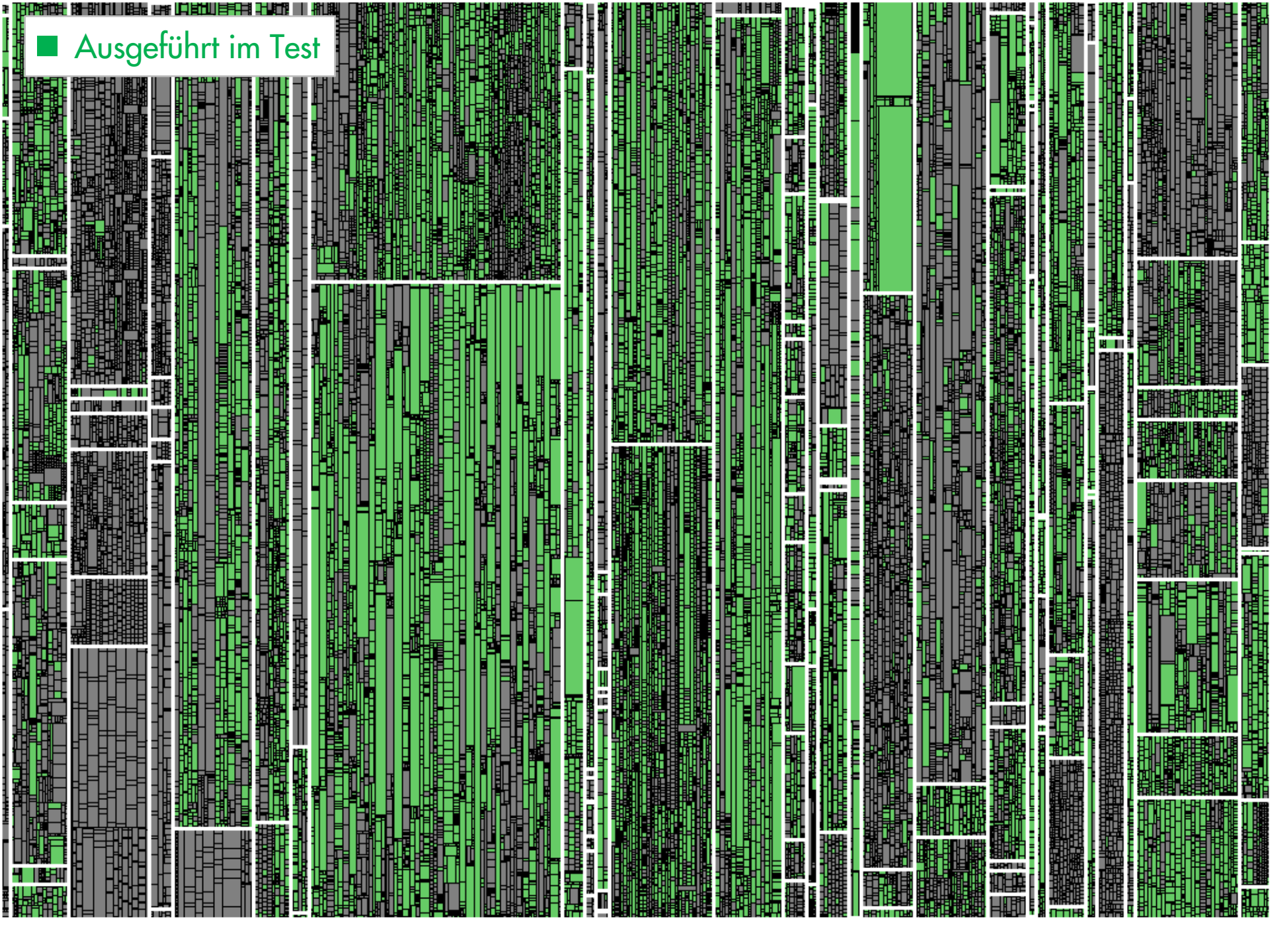


- Neu
- Geändert

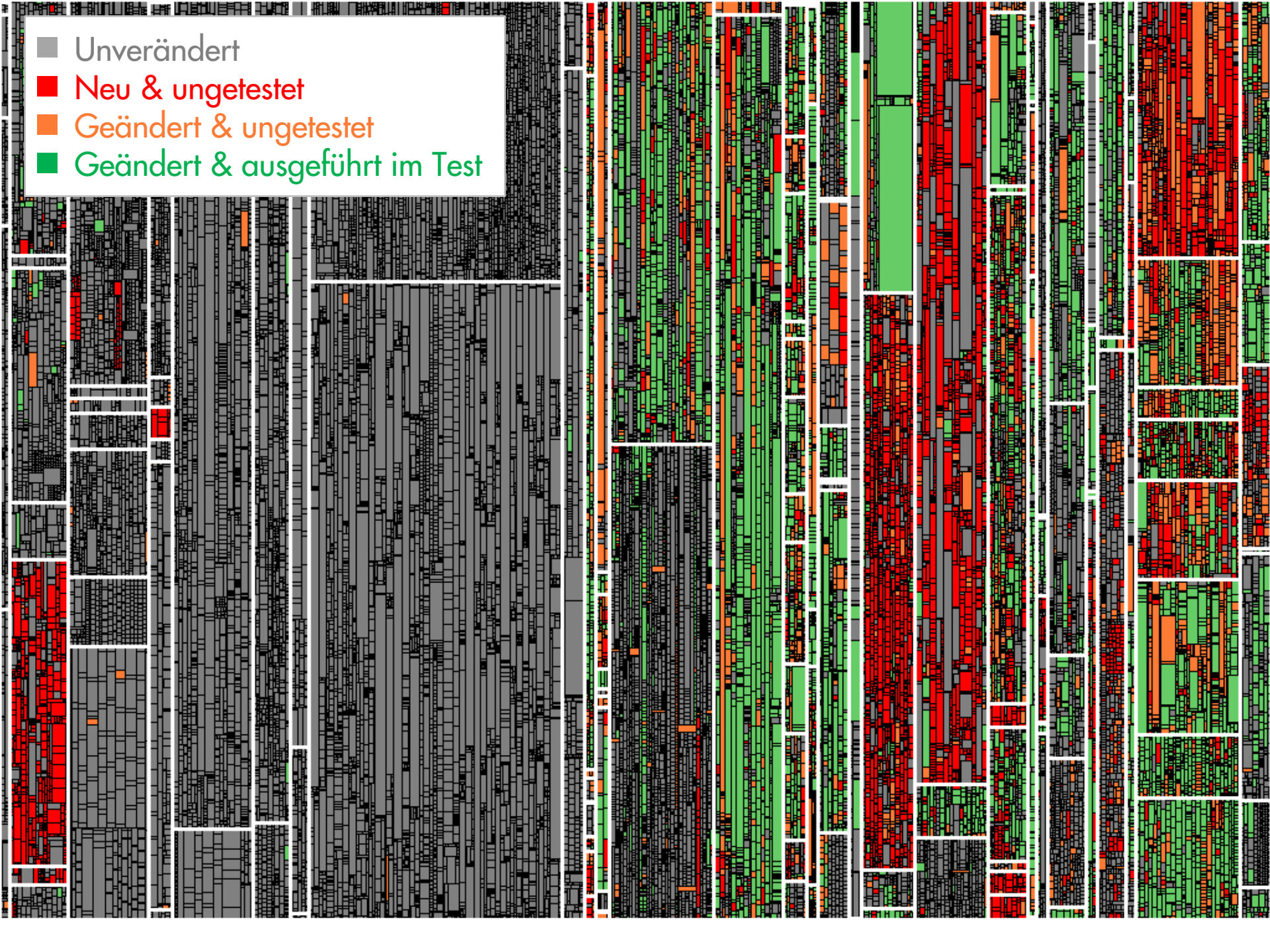




■ Ausgeführt im Test



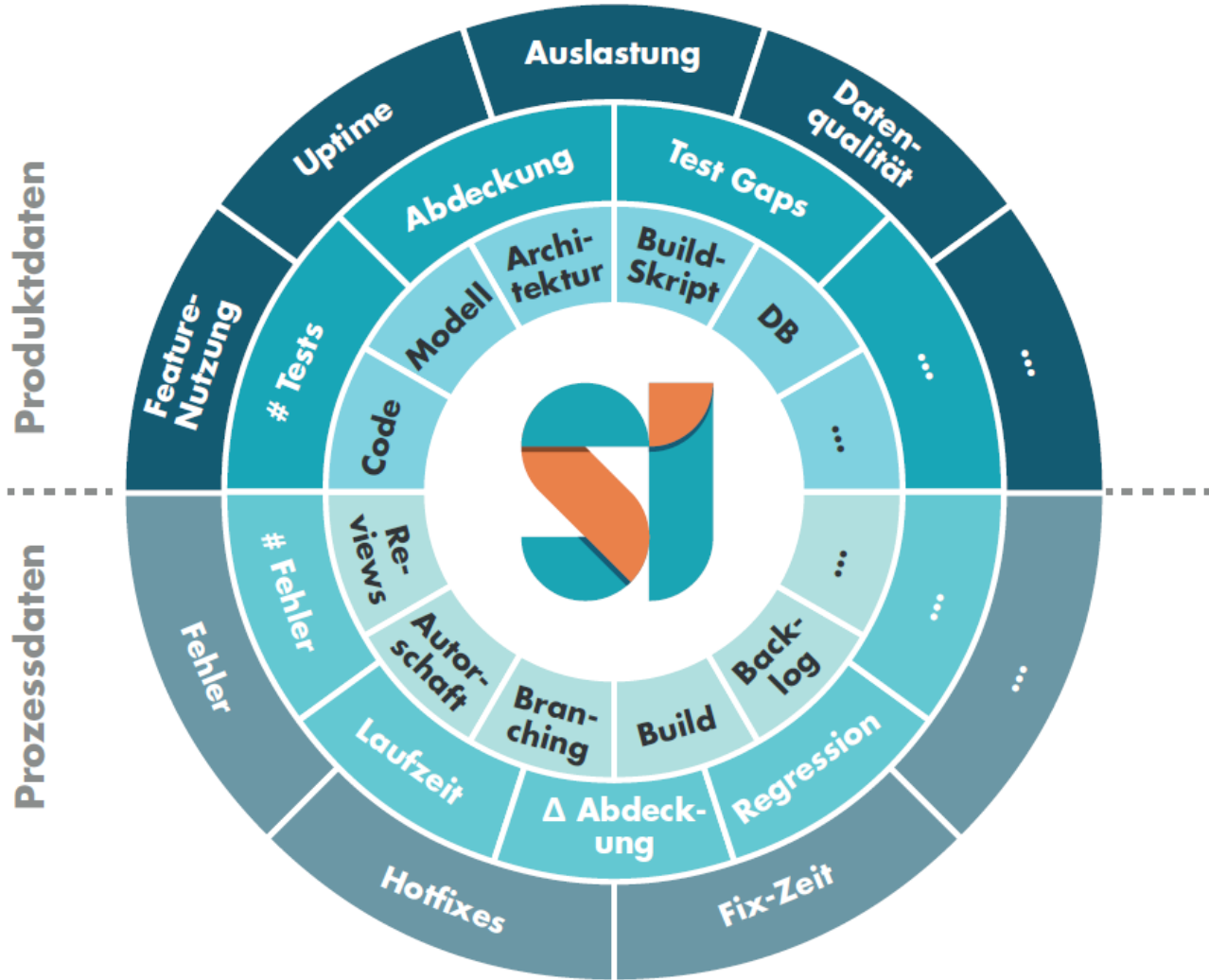
- Unverändert
- Neu & ungetestet
- Geändert & ungetestet
- Geändert & ausgeführt im Test



# Business Intelligence



»A set of **techniques** and **tools** for the acquisition and transformation of **raw data** into meaningful and useful information for **business analysis purposes**.«



## Checking PL/SQL with Teamscale

Posted on 03/15/2017 by [Dr. Andreas Göb](#)

Throughout this blog, there have been several posts explaining how you can use [Teamscale](#) to keep your code-base clean and prevent quality decay. Since we have a lot of enterprise customers, we are often confronted with the question whether our approach works only with modern languages and infrastructure used by small software vendors and start-ups, or if we can also offer analysis for mature technologies mainly used in the enterprise.

My colleague [Christian](#) already reported on Teamscale's analysis capabilities in the [ABAP](#) (and more recently [SAP HANA](#)) world. In this post, I give some examples of checks we recently developed for PL/SQL and which are included in Teamscale 3.2, which is published around the same time as this post.



## Examples for Checks in PL/SQL

As introduced recently on this blog, Teamscale features a flexible source code parsing and analysis facility and allows developing custom static code checks with a few lines of code. Of course, when adding new checks to Teamscale that are local to a single code file, this is what we also use internally.

Don't use SELECT \*

# Wie sag ich's im eigenen Haus?

Mit Vorträgen über Softwarequalität sind wir regelmäßig auf Industriekonferenzen oder Kunden-internen Workshops vertreten.



## Impulsvorträge

Gerne kommen wir auch zu Ihnen ins Haus, beispielsweise für interne Konferenzen oder Workshops. Unsere Themen reichen von Qualitätsanalysen über Qualitätscontrolling bis hin zu Testcontrolling oder der Einführung von Reviews. Oder aber Sie schlagen uns ein Thema Ihrer Wahl vor.

### DAS ANGEBOT

- 🕒 60-90 MIN VORTRAG
- 💬 SOFTWAREQUALITÄT ALS THEMA
- 📍 BEI IHNEN IM HAUS
- € NUR UNSERE ANREISEKOSTEN
- 📅 TERMIN NACH VEREINBARUNG

[IMPULSVORTRAG ANFRAGEN](#)

# Fazit

Qualitätsanalysewerkzeuge sind notwendig, aber nicht hinreichend für Verbesserung der Software-Qualität.

Nachhaltige Verbesserung erfordert die Unterstützung von Entwicklern und Management und die Integration in den Entwicklungsprozess.

Mindestens einer muss sich hierfür verantwortlich fühlen.

# Kontakt

Dr. Elmar Jürgens

[juergens@cqse.eu](mailto:juergens@cqse.eu)

+49 179 675 3863

Dr. Andreas Göb

[goeb@cqse.eu](mailto:goeb@cqse.eu)

+49 176 10155225

[@ElmarJuergens](#)

[@a\\_goeb](#)

[www.cqse.eu/en/blog](http://www.cqse.eu/en/blog)

CQSE GmbH

Lichtenbergstraße 8

85748 Garching bei München