


data-quest 

Codemanufaktur seit 2001



# Qualitätssicherung

aktuelles Modell, Probleme, Lösungen?

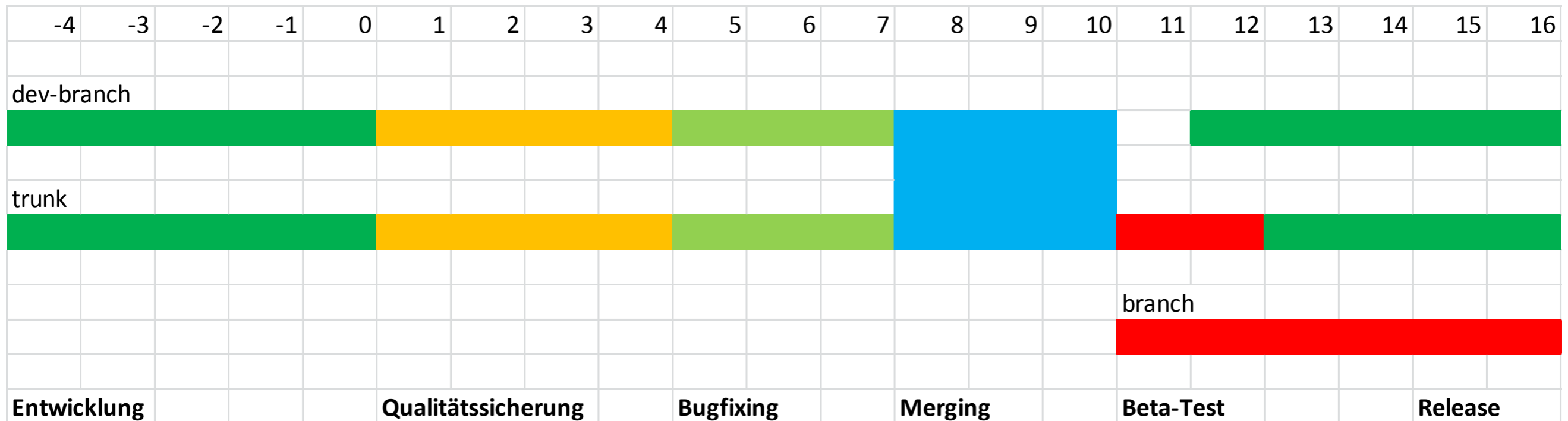
Stud.IP Entwicklungstagung 2018 in Bremen

Stefan Suchi - data-quest

# Verantwortlichkeiten

Gebiet QM	Abnahme StEPs	Veto StEPs & TICs
Code-Qualität	■	■
GUI-Richtlinien		
Funktionalität		
Sicherheit		
Schnittstellen		
Performance		
Qualitätsprüfung deutscher Textstrings		
Entwickler-Dokumentation		
Anwendungs-Dokumentation		

# Zeitlicher Ablauf



# Abbildung im trac

**#7672** **closed TIC (fixed)**

Erstellt vor 7 Monaten

Geschlossen vor 7 Monaten

Zuletzt geändert vor 2 Wochen

## Adminbereich: Anzeigen von Spalten durch Plugins

Erstellt von: Krassmus

Verantwortlicher:

Priorität: normal

Meilenstein: Stud.IP 4.1

Komponente: Administration

Version: trunk

Stichworte:

Beobachter:

Abhängigkeiten:

Parent Tickets:

Code-Review: +

Sicherheit: ?

Code-Konventionen: +

Entwickler-  
Dokumentation: -

Anwender-  
Dokumentation: ?

Funktionalität: +

GUI-Richtlinien: +

Textstrings: +

Schnittstellen: +

# Problem: fehlende Testsysteme

- » Größere Neuentwicklungen in Form von StEPs passieren auf eigenen branches, für die nicht immer ein Testsystem zur Verfügung steht
- » Funktionalität und GUI wird meistens nur von wenigen Personen getestet, häufig nur mit einem Demo-Datensatz

# Problem: Testen im Kontext

- » Frühzeitige trunk-Integration erfordert aktives Einholen von 5 positiven Reviews, häufig unklarer Status
- » Merges in den trunk passieren meistens sehr spät, im Extremfall nur wenige Tage vor dem Beginn des Beta-Tests
- » Keine aktive Nutzung mit echten Daten (z.B. auf dem Developer-Server)
- » Keine Tests der Interaktionen mit anderen StEPs, keine Zeit für Vereinheitlichung

# Problem: Überlastung der Tester

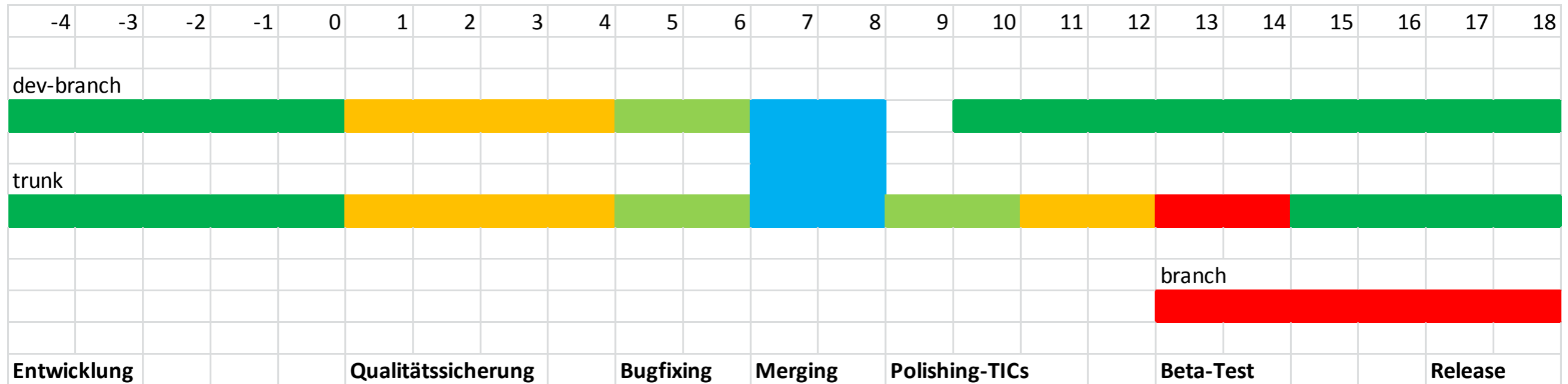
- » Für das Testen aller StEPs und TICs stehen nur vier Wochen zur Verfügung, wenn nicht Entwicklungen vorzeitig abgeschlossen und zum Testen freigegeben werden
- » Einige Aufgaben sind nur mit wenigen Personen besetzt, daher gibt es Probleme bei Urlaub, Krankheit, anderweitigen wichtigen Aufgaben
- » Teilweise Überlappungen der Aufgaben sorgen für Doppelarbeit



# Vorschlag: Polishing-TICs

- » Übernahme der StEPs soll um vier Wochen vorgezogen werden, um mehr Zeit für Tests mit Echtdateien und Vereinheitlichung zu haben
- » Nach dem Mergen sind noch für zwei Wochen spezielle TICs erlaubt, die der Vereinheitlichung und dem Feintuning von GUI und Design dienen  
(*Vorschlag Cornelis Kater*)
- » Zusätzlich zwei Wochen für abschließende Test (auch der Polishing-TICs), Übersetzung, Doku, etc.


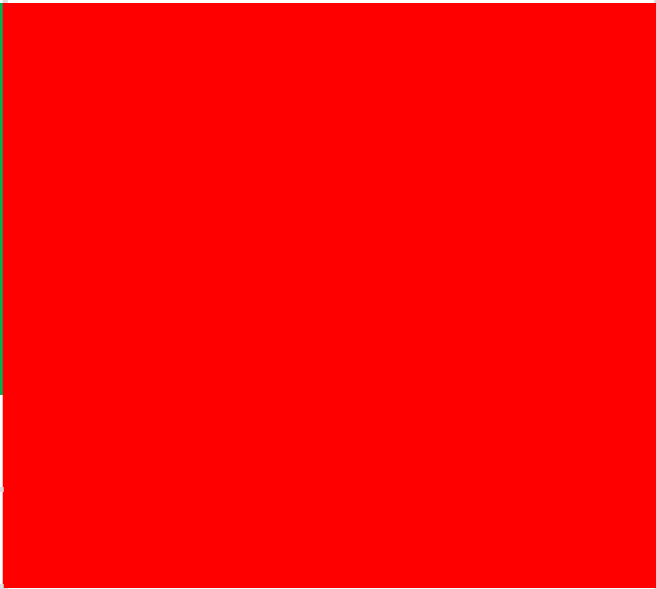
# Vorschlag: Polishing-TICs



# Vorschlag: Neuordnung der Zuständigkeiten

- » Aufteilung der Zuständigkeit „Entwicklerdokumentation“ in die Punkte „API-Dokumentation“ und „Hilfe-Wiki“
- » Zusammenfassung der „API-Dokumentation“ mit dem „Code-Review“, Zusammenfassung „Hilfe-Wiki“ mit der „Anwender-Dokumentation“  
(*Vorschlag Markus Lunzenauer*)
- » Zusammenfassung der Zuständigkeit „Sicherheit“ mit dem „Code-Review“  
(*Vorschlag Stefan Suchi*)

# Vorschlag: Neuordnung der Zuständigkeiten

Gebiet QM	Abnahme StEPs	Veto StEPs & TICs
Code-Qualität + API-Doc + Sicherheit		
GUI-Richtlinien		
Funktionalität		
Schnittstellen		
Performance		
Qualitätsprüfung deutscher Textstrings		
Hilfe-Wiki		

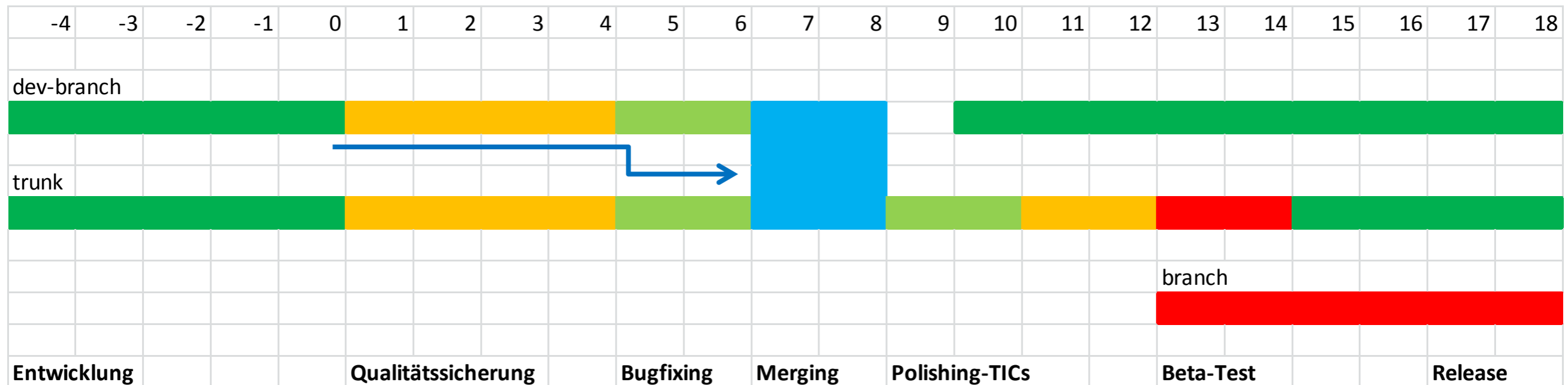
# Vorschlag: nur aktives Veto

- » Streitpunkt: Wer ist verantwortlich für das rechtzeitige Review eines StEPs? Der Autor des StEPs? Oder der jeweilige QM-Beauftragte?
- » Problem: Im Moment kommt es zu de-facto-Vetos durch Nichtstun der entsprechenden QM-Beauftragten, obwohl der Status auf „?“ (zu testen steht).
- » Dies führt am Ende dann häufig zu einem Überschreiten des Zeitplans, was der Idee von Testen im Kontext und Polishing-TICs entgegen steht.

# Vorschlag: nur aktives Veto

- » Meinung: Vetos sollten nur aktiv und begründet ausgesprochen werden können, nicht durch passives Ignorieren erzeugt werden dürfen.
- » Vorschlag: Am Ende der Review-Phase gelten nur gesetzte „-“ als ein Veto. Ein rechtzeitig eingebauter und rechtzeitig zum Testen freigegebenen StEP darf in den trunk, wenn er am Ende der Review-Phase unverändert auf „?“ steht.  
Ein vorzeitiger Einbau erfordert weiterhin „+“ in den relevanten Bereichen.

# Vorschlag: nur aktives Veto



**Vielen Dank!**