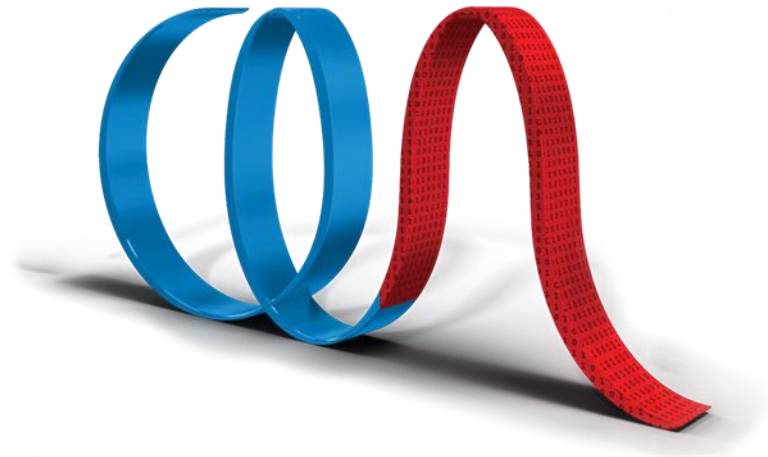


## Beschaffungskonferenz 2019

# Gute Software ist billiger als schlechte Software



We lead from the emerging future!



Marc André Hahn  
Kay Grosskop

# Software ist unternehmenskritisch!

- **Relevanz und Menge** von Software in Unternehmen steigt
- Softwareprojekte werden **immer komplexer**
- Auswirkung von **Fehlern** in der Software werden **immer grösser**
- Es fließen immer **mehr Ressourcen** in Software



# Wartbarkeit als Katalysator der Softwareentwicklung

## Softwarequalität ist messbar.

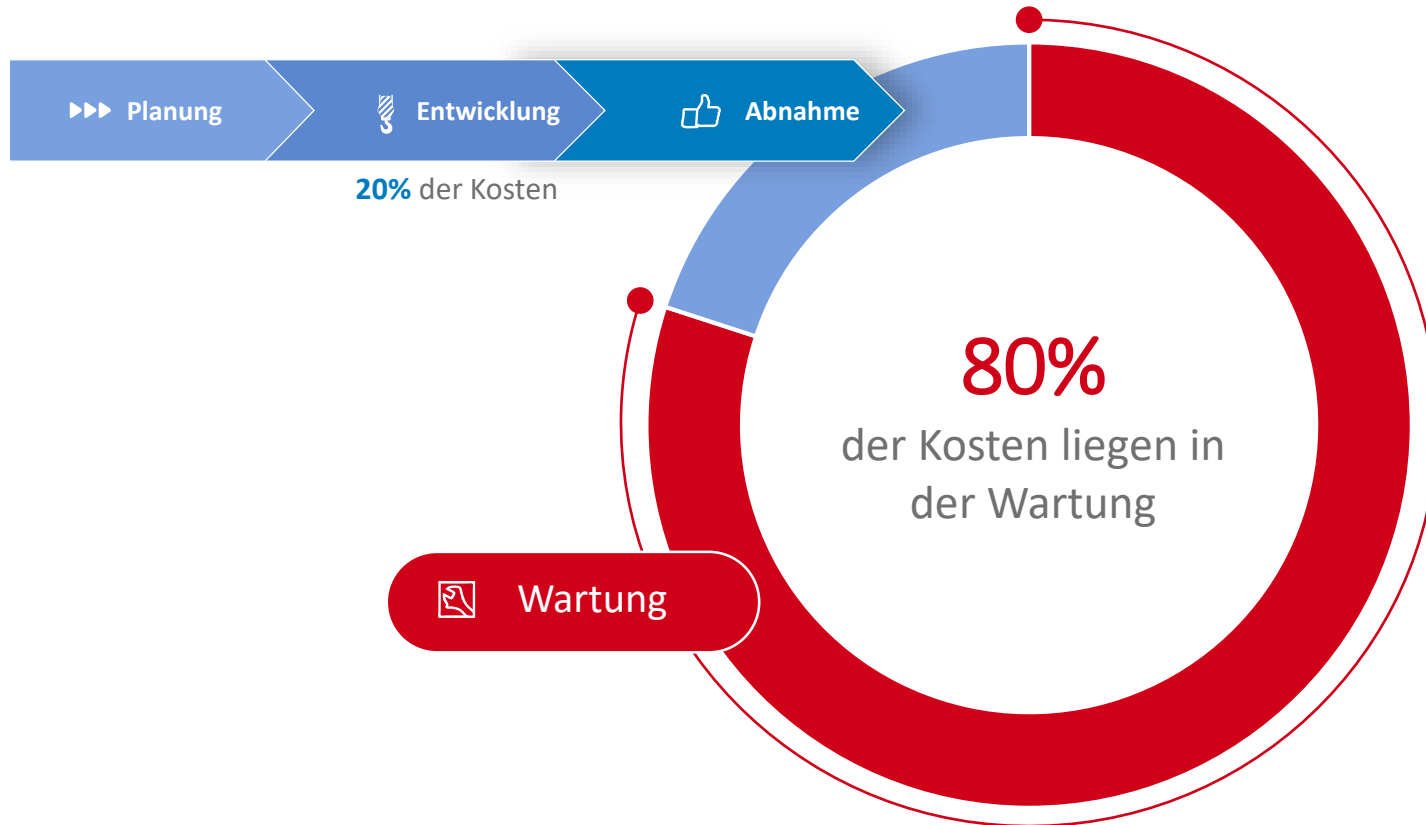
- Der Qualitätsstandard ISO 25010 definiert acht Eigenschaften

## Wartbarkeit ist die Schlüsseleigenschaft

- Einfluss auf **Effektivität** und **Effizienz** von Änderungen am System
- Einfluss auf **Behebung** von **Fehlern**
- Einfluss auf **funktionale Erweiterungen**
- Einfluss auf die **Veränderbarkeit** der anderen **Qualitätseigenschaften**
- Wartbarkeit ist der **Kostentreiber**



# Lebenszyklus von Software



# Objektivität durch Standards

- **Standard:** SIG/TÜViT Bewertungsverfahren basierend auf dem **ISO/IEC 25010** Standard für Software Qualität.
- **Zertifizierung:** «Trusted Product Maintainability»
- **Labor:** nach ISO/IEC 17025 vom TÜViT **zertifiziertes Labor** für die Messung von Software



Zertifikat für Produktwartbarkeit.  
Gemessen nach Standards der TÜViT

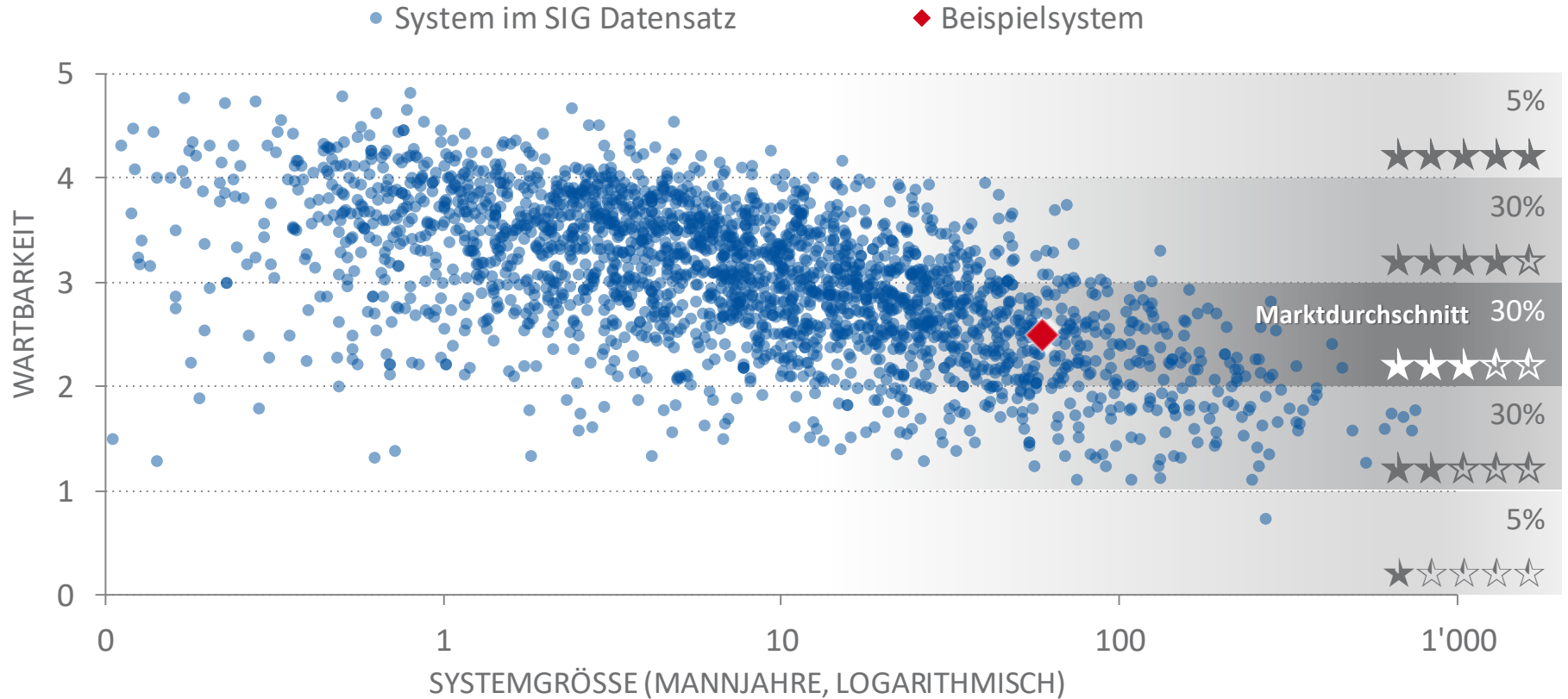


# Von der Messung zum Star-Rating



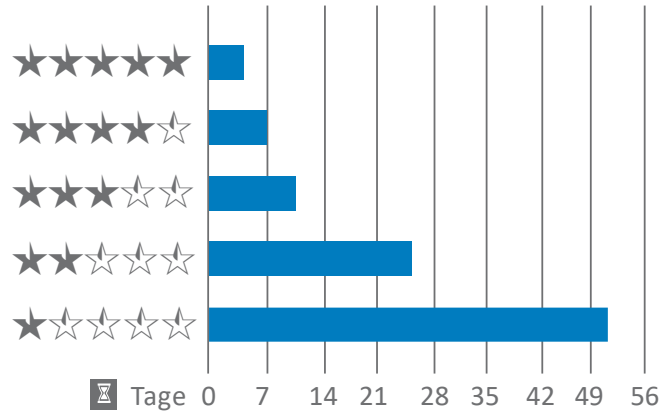
- Einzelne Kennzahlen werden zu einem Rating verdichtet.
- Die Bewertung erfolgt anhand eines Benchmarks.
- Das Rating ist durch den TÜVIT Zertifiziert.

# 5 Milliarden Zeilen Code im Benchmark

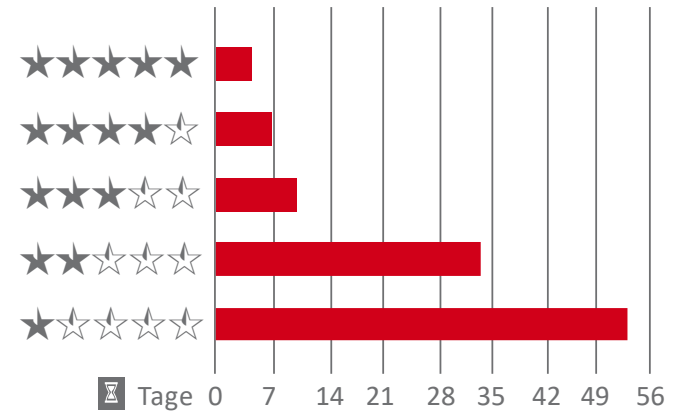


# Auswirkung von Wartbarkeit

## Benötigte Zeit für Erweiterungen

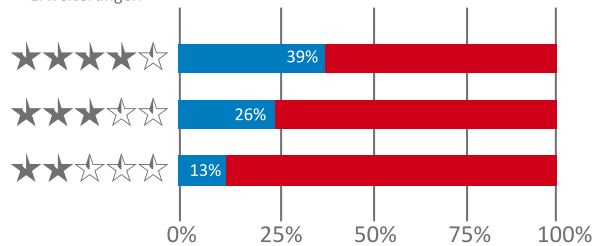


## Benötigte Zeit für Fehlerbehebungen

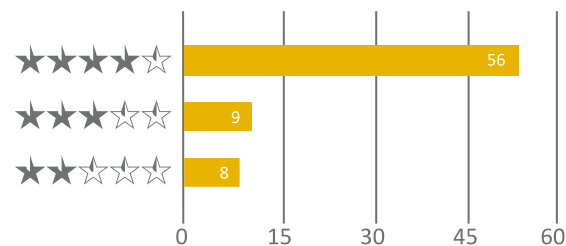


■ Fehlerbehebungen  
■ Erweiterungen

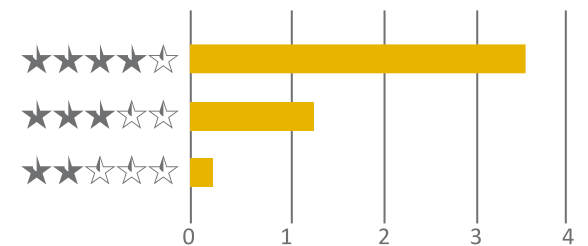
## Effizienz



## Gelöste Probleme pro 100k Zeilen Code



## Gelöste Probleme pro Entwickler pro Monat





## Auswahl von Anwendung in der Praxis

---

# Situation der Entscheidungsträger

## Situation



- Entscheider in den Unternehmen/Organisation müssen weitreichende Entscheidung über Software treffen. Ohne Transparenz über die objektive Qualität von Software und Leistungen der Lieferanten können **keine sinnvollen Entscheidungen** getroffen werden!
- Entscheidungen werden in der Regel nur auf Basis von Informationen getroffen, die der Lieferanten bereitstellt - die Interessen sind jedoch nicht deckungsgleich! Es herrscht eine **Informationsasymmetrie!**

## Folgen



- Abhängigkeit von Lieferanten
- Zu hohe Kosten
- Mangelnde Flexibilität

Einblick in die Qualität von Software und in die Leistungen der Lieferanten spielen in allen Phasen des Lebenszyklusses von Software eine Rolle um **nachhaltige Entscheidungen** zu treffen!

# Anwendungsfälle – Qualität als Entscheidungskriterium



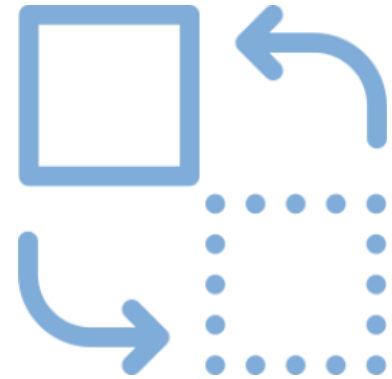
## Beschaffung

von Software



## Wartung

von Software



## Ersatz

von Software



# Qualität in der Beschaffung von Software



## Individualentwicklung

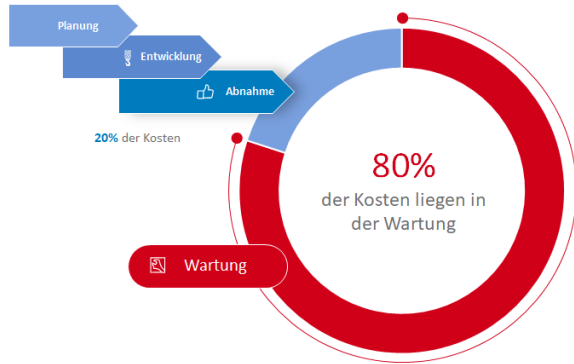
- **Arbeitsprobe** in der Beschaffung als Teil der Evaluation
- **Qualitätssicherung** und Fortschrittskontrolle in der Entwicklungsphase
- **Qualität** als **Abnahmekriterium**
- ...



## Standard-Software

- **Qualität** als Entscheidungskriterium (nicht immer möglich)
- **Qualität** als Hebel in den Preisverhandlungen
- Erarbeitung eines Verbesserungsplan in der Implementierungsphase

# Qualität in der **Wartung** von Software



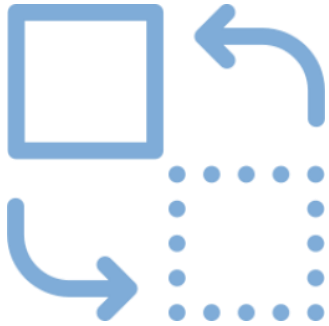
- Herstellung von Informationssymmetrie
- Transparenz über Qualität und Quantität der Lieferantenleistung
- Wirkungsvolle Wartungsverträge
  - Nicht nur Kosten, sondern Leistungen
  - Verantwortung für nachhaltige Wartbarkeit dem Lieferanten übertragen (nachhaltige Wartung)



- Know-how- und Teamentwicklung werden transparent
- Kosten und Leistungen können gegenübergestellt werden



# Qualität in der **Renovation/Ersatz** Entscheidung



- Objektive Grundlage über die Qualität des existierenden Systems sind möglich
  - Qualität des bestehenden Codes
  - Rebuild Value
  - Zukunftstauglichkeit
- Berechnung von Business Cases, ob es wirtschaftlicher ist, den Code zu renovieren oder das System komplett neu zu entwickeln

# Fragen, welche objektiviert werden können – und sollten!

Wie kann ich meine Wartungskosten senken?

Wie produktiv ist das Entwicklerteam?

Ist der Zustand meiner Software ein Risiko für die Unternehmensentwicklung?

Wie kann ich den Fortschritt meines Softwareprojektes objektiv messen?

Entspricht meine Software den Architekturvorgaben?

Wie kann ich die Abhängigkeiten von meinem Lieferanten reduzieren?

Wie hoch ist der Rebuildvalue meiner Software?

Soll ich meine Software renovieren, neu schreiben oder lassen wie sie ist?

Ist die Offerte meines Lieferanten angemessen?

Bekomme ich einen angemessenen Gegenwert von meinem Software-Lieferanten?





Marc André Hahn  
sieber&partners  
Schwanengasse 1, Bern  
Usteristrasse 19, Zürich  
+41 78 686 85 16  
marc.hahn@sieberpartners.com



Kay Grosskop  
sieber&partners, SIG  
Schwanengasse 1, Bern  
Usteristrasse 19, Zürich  
+41 31 566 93 00  
kay.grosskop@sieberpartners.com