



## Der Testreport. Was soll, was darf und was muss drinstehen ?

**Dr. Gerald Zincke, SQS Ges.mbH, Wien**



## **Dipl.-Ing. Dr. Gerald Zincke**

Principal Consultant, SQS Ges.mbH Wien, ISTQB cert. Testmanager

- Studium der Informatik in Linz, Diplomarbeit über Firmware, Abschlussarbeit über “Interaktiver Software Entwurf”
- Projekt Management seit 1987 (Finanz, Fertigung, TK)
- Qualitätssicherung und Test seit 2000
- Bei SQS seit 2009



## **SQS Software Quality Systems**

Der weltweit führende Spezialist für Software-Qualität

- Seit 30 Jahren am Markt, Erfahrung aus über 7000 Projekten
- 3500 Mitarbeiter in 16 Ländern, davon etwa 100 in Österreich
- Beratung, Managed Services, Tools, Training und Konferenzen
- Testcenter in Österreich, Deutschland, Ägypten, Südafrika und Indien

Inhalt

## Themen

---



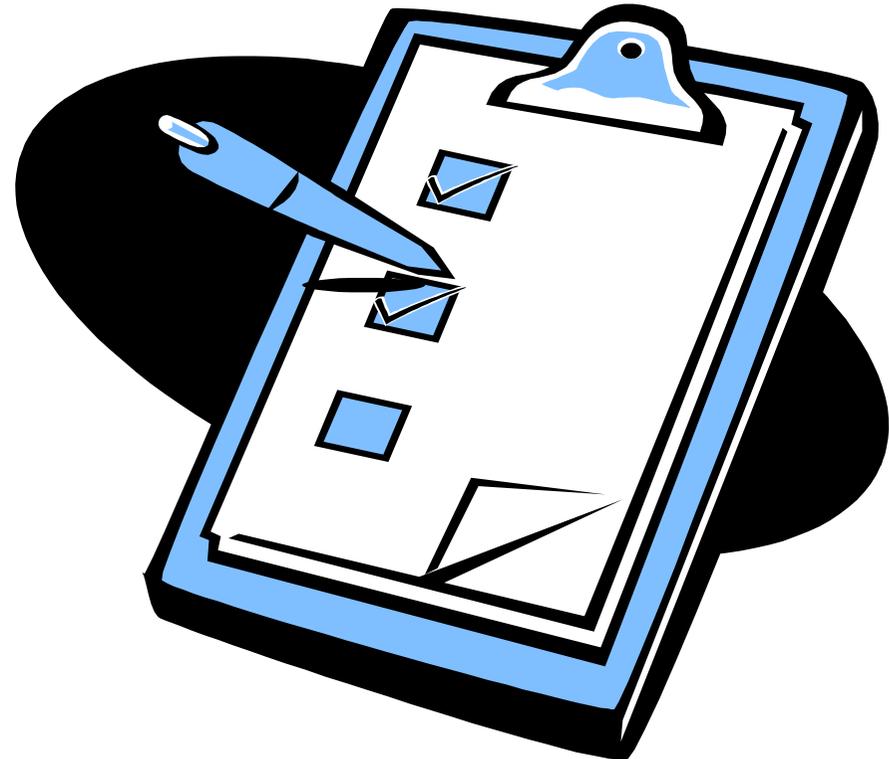
Testergebnisse

Testfortschritt

Testaufwand

Qualität/Reife

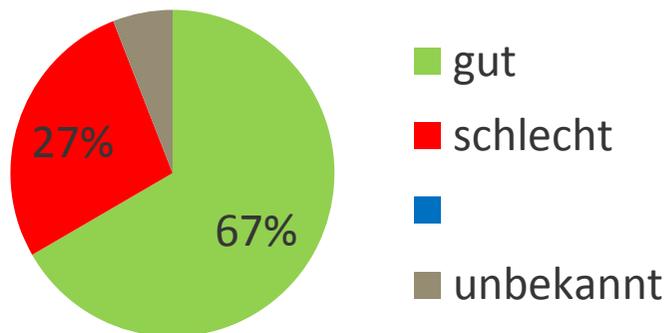
Zusammenfassung



## Abweichungen

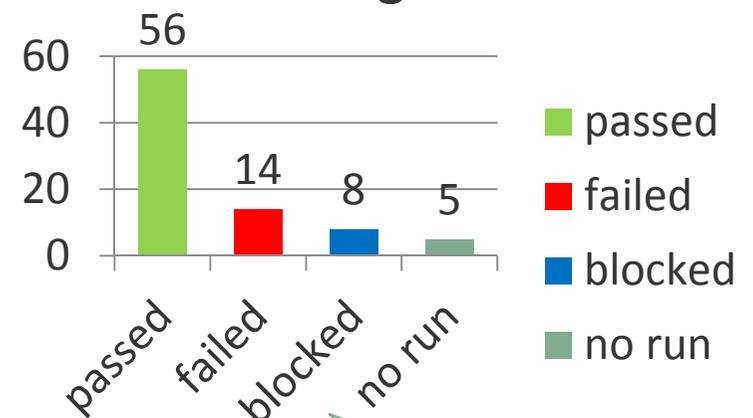
- Pro Testfall failed/blocked, Beschreibung der Abweichung: Für Fehlerbehebung
- Anzahl passed/failed/blocked/no run: Metrik wofür eigentlich ?
- Aber
  - Passed/(failed+blocked), aufgeschlüsselt nach Feature, Teilsystem : Indikator für Produktqualität

### Produktqualität



- $(\text{passed} + \text{failed} + \text{blocked}) / \text{no run}$  : Metrik für Testfortschritt bzw. Regressionssicherheit

### Testergebnis



quantitative  
Messdaten.  
Qualitative Aussage?

## Fehler

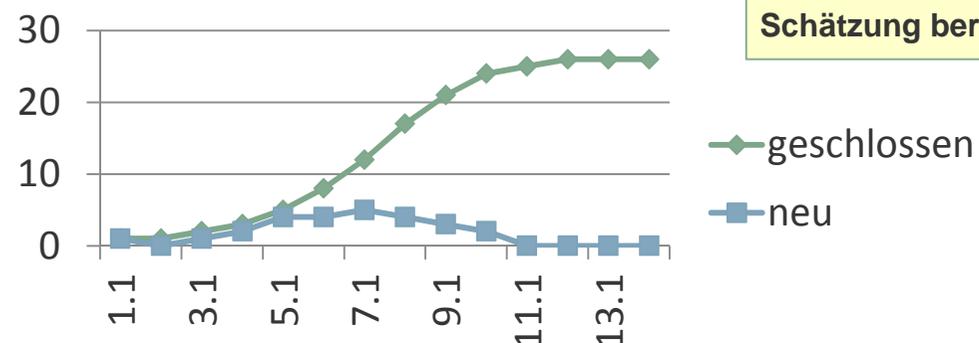
- Anzahl gefundene, offene Fehler: Metrik wofür ?
  - in Arbeit, Bearbeitungsdauer: Metrik für Analyse-Kapazität
  - gefixt , Bearbeitungsdauer: Metrik für Bugfix-Kapazität
  - Verifiziert, Testdauer: Metrik für Bugfix-Verifikations-Kapazität
  - Metrik für Produktqualität :  $(\text{Offene Fehler} + \text{Unbekannte Fehler}) / (\text{geplante})\text{Größe}$   
Wie Anzahl unbekannter Fehler schätzen ? Wie Größe messen ?
- Anzahl gefundener, geschlossene Fehler: Metrik wofür ?

Fehler nicht gleich  
failed Testcase

Keine Messdaten,  
Schätzung berichten?

**Metrik für Entwicklungsqualität:  $(\text{Gefundene Fehler} + \text{Unbekannte Fehler}) / \text{Größe}$**

**Bug Burndown Chart: ab Testbeginn Fortschrittsindikator  
(Zweite Ableitung )**



Keine Messdaten,  
Schätzung berichten?

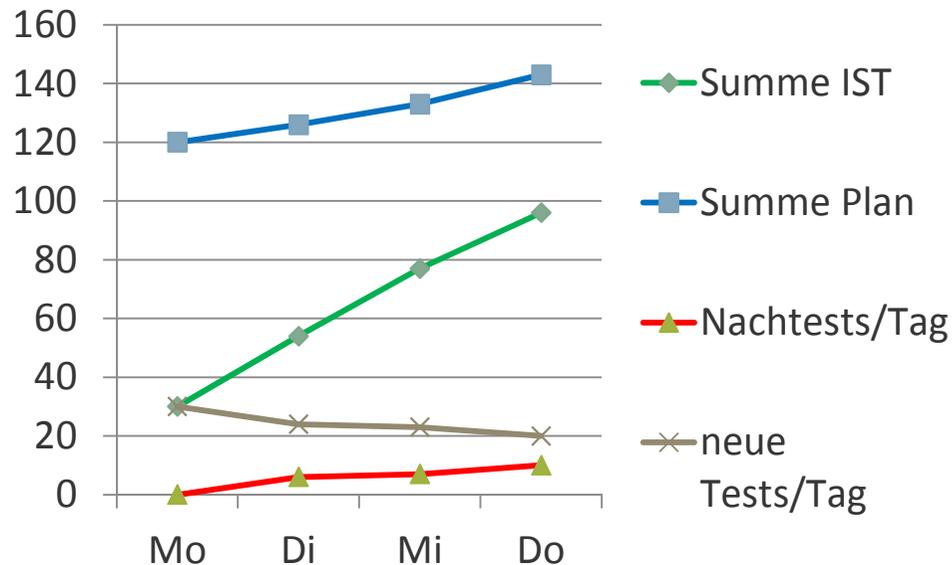
# Fortschritt



## Fortschritt

- Testausführung
- Testfallentwicklung

## Testausführung



Wie kann man die Anzahl der Tests planen ?

Einflussfaktoren:

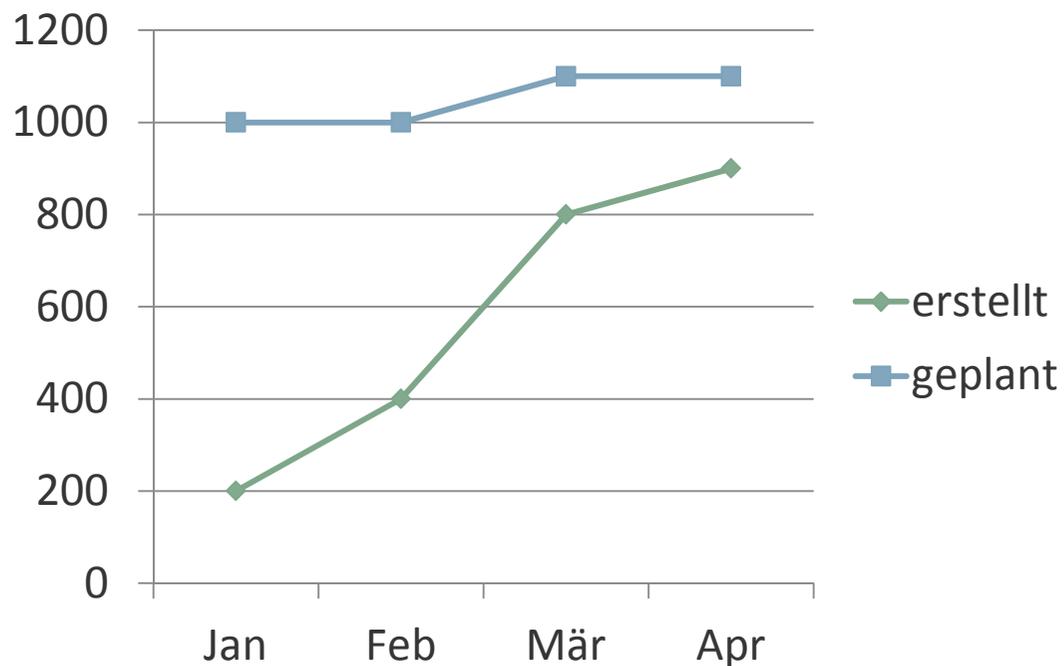
- Anzahl Testfälle, Testabdeckung
- Testproduktivität
- Anzahl der Testkonfigurationen
- Anzahl der Fehler/Nachtests
- Regressionsstrategie

- Anzahl ausgeführter TF/Anzahl geplanter TF : Metrik für Testfortschritt  
Geplante Tests: Größe des Testsets+Fehlernachtests+Regressionstests

Keine Messdaten,  
Schätzung berichten?

## Testfallentwicklung

- Anzahl fertiger TF/Anzahl geplanter TF : Metrik für TF-Entwicklungsfortschritt



## Können wir die Anzahl der geplanten TF schätzen ?

- Einflussfaktoren: Testbasis (Anforderungen), Testmethode/Testabdeckung, Testfall-Granularität, Prozess, TF-Entwicklungs-Produktivität

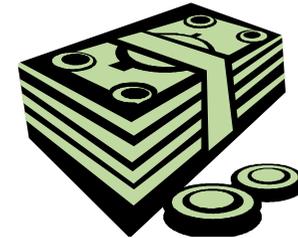
Messdaten,  
verfügbar ?

# Aufwand & Dauer



## Kosten

- Plankosten
- Ist Kosten, Restbudget: Metrik wofür ?
- Erwartete Gesamtkosten – Budget : Indikator zur Kostenkontrolle
- Erwartete Gesamtkosten/Plankosten: Metrik zur Treffsicherheit des Plans



## Personentage

- Plan
- Ist, Restaufwand : Metrik wofür ?
- Erwartete GesamtPT – PlanPT : Indikator zur Personalplanung
- Erwartete GesamtPT/PlanPT : Metrik zur Treffsicherheit des Plans



## Dauer

- Termin
- Bisherige Dauer, Restdauer: Indikator wofür ?
- Plandauer \* Treffsicherheit : Indikator für erwarteten Termin



## Wie Erwartungswerte abschätzen ?

Ist die SW reif für Inbetriebnahme/Auslieferung ?  
Kann und soll Testbericht qualitative Aussagen treffen ?



## Definition für Reife ?

- Vorschlag: Reif = Erforderliche Funktionalität vorhanden  
& Folgekosten offener + unbekannter Fehler „vertretbar“  
& (Kosten für Fortsetzung Bugfixing&Test) > (Nutzen Testfortsetzung)
- Wie berechnen ?
  - Erforderliche Funktionalität= need v.s. nice to have : Entscheidung des Benutzers
  - Fehlerfolgekosten: Thema fürs Risikomanagement bzw. für den SW Betrieb
  - Kosten für Fortsetzung: **Testkosten** + Bugfixing + Kosten der Nichtnutzung
  - Nutzen Testfortsetzung: Reduktion der Fehlerfolgekosten: s.o.

**Woher kommen  
die Daten ?**

## Politische Entscheidung

- Abwägung von Risiken außerhalb des Tests, der SW-Entwicklung, der IT

# Wir sollten berichten, was interessiert

## Wir dürfen nicht berichten, was nicht auf Fakten basiert



### Der Testbericht sollte enthalten

#### Ergebnisse

- Beschreibung der Abweichungen pro failed Testfall
- Indikator Produktqualität: Passed/(failed+blocked)

#### Fortschritt

- Testausführung: (passed+failed+blocked)/no run
- Testfallentwicklung: Anzahl fertiger TF/Anzahl geplanter TF

#### Steigung (zweite Ableitung) im Bug Burndown Chart: Fortschrittsindikator (in Testphasen)

### Der Testbericht kann enthalten

#### Summe passed/failed/blocked : wenig aussagekräftig

#### Fehlerstatistiken: Eher ein Thema fürs Projektmanagement

- Produktqualität : (Offene Fehler + Unbekannte Fehler) /Größe

#### Testaufwand: Eher ein Thema fürs Projektmanagement

### Der Testbericht sollte nicht enthalten

#### Beurteilung der Reife

#### Empfehlung zur Auslieferung/Inbetriebnahme

#### Entscheidung zur Auslieferung/Inbetriebnahme





sqs.com

## **SQS Software Quality Systems Ges.mbH**

Hietzinger Kai 67–69  
1130 Wien, Österreich  
Telefon: +43 1 3193523-0  
Telefax: +43 1 3193523-24  
[info-austria@sqs.com](mailto:info-austria@sqs.com)

Dr. Gerald Zincke  
Mobil: +43 664 88456135  
[Gerald.Zincke@sqs.com](mailto:Gerald.Zincke@sqs.com)

Vielen Dank für Ihre Aufmerksamkeit.