

# Übung: Software-Qualität

Sommersemester 2016

[swq@se.uni-hannover.de](mailto:swq@se.uni-hannover.de)

Funktion, die eine Software-Einheit in einen Zahlenwert abbildet. Dieser Wert ist interpretierbar als der Erfüllungsgrad einer Qualitätseigenschaft der Software-Einheit.  
(IEEE Standard 1061)

- Wir betrachten die Metrik „**Anzahl Commits**“.
- Was kann mit dieser Metrik gemessen werden?
  - Produktivität
  - Schnelligkeit
  - Effizienz (Leistung pro Zeit)
- Wie kann die Metrik manipuliert werden?
  - Durch viele kleine Commits
- Wie kann die Metrik angepasst werden, um die Manipulation zu verhindern?
  - Indem nicht nur die Anzahl der Commits betrachtet wird, sondern auch die Änderungen.

# Metriken zu Qualitätsaspekten

- Überlegen Sie sich zu jedem der unten stehenden Qualitätsaspekte eine entsprechende Qualitätsmetrik. Sofern möglich, geben Sie zu den Metriken auch den jeweiligen Wertebereich an und erläutern Sie, wie das Ergebnis zu interpretieren ist.
  - Flexibilität
  - Portabilität
  - Testbarkeit
  - Verwendbarkeit/Bedienbarkeit (Usability)
  - Korrektheit
  - Effizienz
  - Zuverlässigkeit

# Metriken zu Qualitätsaspekten

## Flexibilität

- Metrik 
$$\frac{\# \text{Abstrakte Klassen} + \# \text{Interfaces}}{\# \text{Gesamtklassen} + \# \text{Interfaces}}$$
- Wertebereich:  $0 \leq x \leq 1$
- Interpretation: Größere Werte bedeuten eine höhere Abstraktion und damit eine höhere Flexibilität für den Austausch von Implementierungen

## Portabilität

- Metrik:
  - Verwendete Programmiersprache
  - Anzahl an nicht standardisierten Features
  - Verhältnis zwischen der Größe der plattformunabhängigen Anteile und der Programmgröße

# Metriken zu Qualitätsaspekten

## Korrektheit

- Metrik:  $\frac{\# \text{erfolgreiche Abnahmetests}}{\# \text{Abnahmetests}}$
- Wertebereich  $0 \leq x \leq 1$
- Interpretation: Höhere Werte bedeuten mehr erfolgreiche Abnahmetests und daher eine höhere Korrektheit

## Effizienz

- Metrik (hier Device Efficiency): Berechnungen pro Sekunde
- Wertebereich: positive rationale Zahl
- Interpretation: Eine höhere Zahl ist besser, da diese anzeigt, dass mehr Berechnungen pro Sekunde durchgeführt werden

# Metriken zu Qualitätsaspekten

## Zuverlässigkeit

- Metrik (hier Robustheit): 
$$\frac{\text{Gesamtzeit} - \text{Downtime}}{\text{Gesamtzeit}}$$
- Wertebereich:  $0 \leq x \leq 1$
- Interpretation: Je näher an 1 desto besser, da dies eine geringere Downtime bedeutet.

## Testbarkeit (Testability)

- Metriken:
  - Zyklomatische Komplexität (McCabe)
  - Durchschnittliche Anzahl der Parameter pro Methode

# Metriken zu Qualitätsaspekten

## Verwendbarkeit (Bedienbarkeit: Usability)

- Metrik: 
$$\frac{\text{Zeit zur Lösung der Aufgabe ohne Software}}{\text{Zeit zur Lösung der Aufgabe mit Software}}$$
- Wertebereich:  $\mathbb{Q}^+$  (positive rationale Zahl)
- Interpretation: Werte größer als 1 sind gut, da sie bedeuten, dass man mit der Software die Aufgabe schneller löst als ohne. Werte kleiner als 1 bedeuten, dass man mit der Software langsamer ist, also schlecht.
- Weitere Metriken:
  - Anzahl ungünstiger Farbkombinationen
  - Anzahl ungünstiger GUI-Elemente (Radio-Buttons vs. Check-Boxes)
  - Anzahl der Verstöße gegen GUI-Richtlinien (z.B. cancel-ok, statt ok-cancel)