

# Softwareanforderungsanalyse

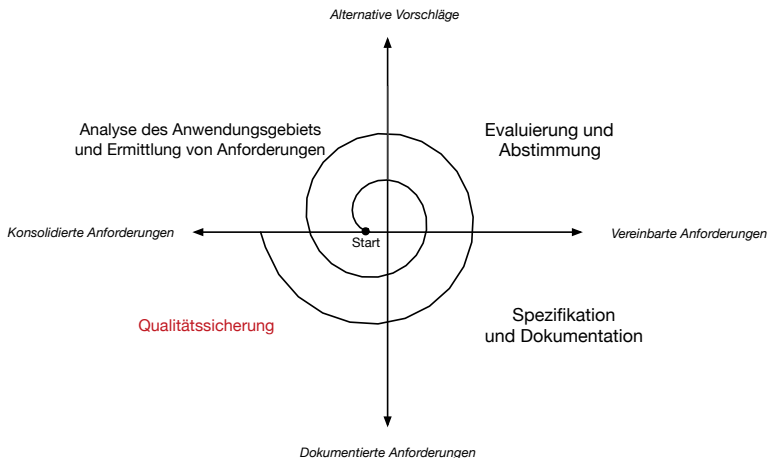
## Qualitätssicherung — Validierung von Anforderungen

Burkhardt Renz

THM, Fachbereich MNI

Wintersemester 2018/19

# Qualitätssicherung, Validierung von Anforderungen



# Übersicht

- Aufgaben der Qualitätssicherung
- Reviews
- Prototypen zur Überprüfung
- Simulation/Animation von Anforderungen
- Formale Überprüfungen

# Grundsätze zur Prüfung von Anforderungen

- **Validieren:** die richtigen Anforderungen spezifiziert?
  - adäquat?
  - vollständig?
- **Verifizieren:** richtig spezifiziert?
  - konsistent?
  - verständlich?
  - eindeutig?
  - widerspruchsfrei?
  - risikogerecht?

# Übersicht

- Aufgaben der Qualitätssicherung
- **Reviews**
- Prototypen zur Überprüfung
- Simulation/Animation von Anforderungen
- Formale Überprüfungen

# Reviews von Anforderungsspezifikationen

- Inspektion der Dokumentation durch Projektmitglieder und/oder externe Experten
- hat sich nicht nur bei Quellcode, sondern gerade bei Anforderungsspezifikationen als nützlich erwiesen
- Verschiedene Formen
  - Walk-through:  
der Autor erläutert Schritt für Schritt seine Spezifikation, die anderen Teilnehmer überprüfen, „Scribe“ notiert Befunde
  - Inspektion: definierter Prozess zur Überprüfung

# Vorgehen bei Inspektion

- ➊ **Planen** der Inspektion  
Reviewer festlegen, Termine vereinbaren, Vorgehen festlegen
- ➋ **Überblick**  
Treffen, bei der die Spezifikation vorgestellt wird:  
Ziel, Aufbau etc.
- ➌ **Individuelles Review**  
Reviewer beurteilen individuell, z.B. nach Check-Listen  
und/oder spezialisiert für bestimmte Fragestellungen
- ➍ **Sammlung der Ergebnisse**  
Treffen aller Reviewer zum Zusammentragen und  
Konsolidieren der Ergebnisse der individuellen Reviews
- ➎ **Ergebnisse dokumentieren**  
Bericht der gefundenen Probleme und Defekte, auch mit  
Verbesserungsvorschlägen

# Richtlinien für Reviews/Inspektionen

- Vorgehen betont **sachlich**  
informativ, genau, konstruktiv
- Reviewer sollten **unabhängig** von Entwicklern der Spezifikation sein
- ... sollten **Experten** für Teilgebiete sein
- ... sollten repräsentativ für **alle** Beteiligten sein
- **Zeitlichen Aufwand** für Review nicht zu knapp kalkulieren – besser mehr kürzere Treffen
- Konzentration bei der Inspektion auf **kritische** und **risikoreiche** Punkte



# Checklisten

Oft werden Checklisten als Leitlinie für die Reviewer verwendet.

## Typen von Checklisten

- **Fehler**listen  
Generische Fragen, die auf typische Fehler führen  
fehlt etwas? Begriffe einheitlich? etc etc
- Listen von Fragen bzgl. **Qualitätsmerkmale**  
Nicht-funktionale Anforderungen im Fokus  
zielt oft auf fehlende Anforderungen
- **Domänenspezifische** Checklisten  
Expertenwissen über das Anwendungsgebiet als Grundlage der  
Fragestellungen beim Review
- **Sprachspezifische** Checklisten  
Grundlage bekannte Fehler/Versäumnisse bei bestimmten  
Spezifikationssprachen  
häufiger verwendet im Code-Review

# Übersicht

- Aufgaben der Qualitätssicherung
- Reviews
- **Prototypen zur Überprüfung**
- Simulation/Animation von Anforderungen
- Formale Überprüfungen

# Prototypen zur Überprüfung

- Sehr **effektive** Form der Überprüfung, da durch Beteiligte unmittelbar erfahrbar
- Arten: **evolutionäre** vs. **Wegwerf**-Prototypen
- Durchspielen von **Szenarien** am Prototypen
- gut geeignet für die Überprüfung der **Adäquatheit** und **Benutzbarkeit**
- nicht so gut geeignet für andere nicht-funktionale Anforderungen wie z.B. **Sicherheit**
- meist kann **Vollständigkeit** nicht geprüft werden

# Übersicht

- Aufgaben der Qualitätssicherung
- Reviews
- Prototypen zur Überprüfung
- **Simulation/Animation von Anforderungen**
- Formale Überprüfungen

# Simulation/Animation von Anforderungen

## Begriffe

- **Simulation**: Ausführung eines Modells
- **Animation**: Suggestive Visualisierung des simulierten Modells in seiner Umgebung

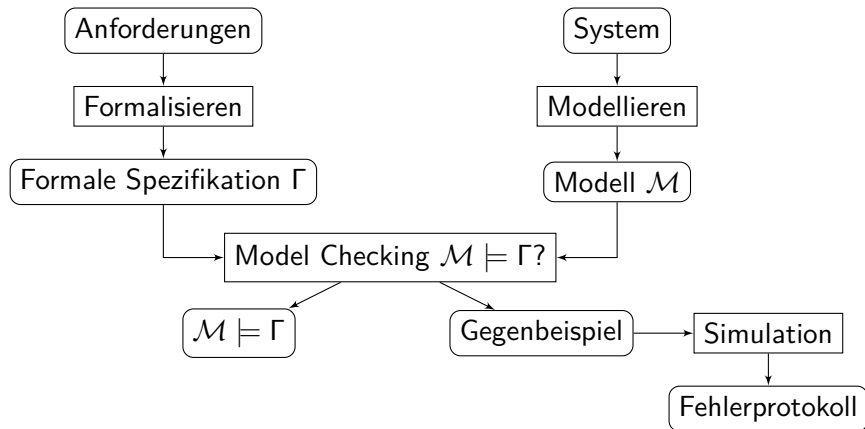
## Beispiele

- Jaza Animator für Z
- Rhapsody für State Charts
- Alloy Analyzer für Alloy
- ProB Animator and Model Checker

# Übersicht

- Aufgaben der Qualitätssicherung
- Reviews
- Prototypen zur Überprüfung
- Simulation/Animation von Anforderungen
- **Formale Überprüfungen**

# Vorgehen beim Model Checking



# Beispiele für Model Checking mit SPIN

- Untersuchung National Highway Traffic Safety Administration Study of Unintended Acceleration in Toyota Vehicles 2011
- Steuerung der Sturmflutwehr in Rotterdam 1996
- Lucent PathStar Access Server – Telefonvermittlungsanlage 2000
- Standards für die Übermittlung von Daten medizinischer Geräte 1996 - heute
- ...
- Needham-Schroeder (siehe <https://homepages.thm.de/~hg11260/mat/modelcheck-bsp.pdf>)



# Fazit

- Qualitätssicherung **wichtige Aufgabe** - Fehler in der Anforderungsspezifikation sind sehr teuer
- **Reviews**  
eignen sich für alle Arten von Spezifikationen  
effektiv, wenn gut vorbereitet  
weniger geeignet für subtile Fehler
- **Prototypen**  
gut geeignet für Prüfen der Adäquatheit  
gut für Feedback künftiger Benutzer
- **Animationen**  
findet auch subtile Probleme, aber nur partiell  
setzt ausführbares Modell voraus
- **Model Checking**  
für subtile Fehler viel besser als Testen  
aber aufwändig