

# Testbare Anforderungen modellbasiert spezifizieren

27. Nov. 2014 FG-Treffen

Michael Felderer<sup>1</sup>, Andrea Herrmann<sup>2</sup>

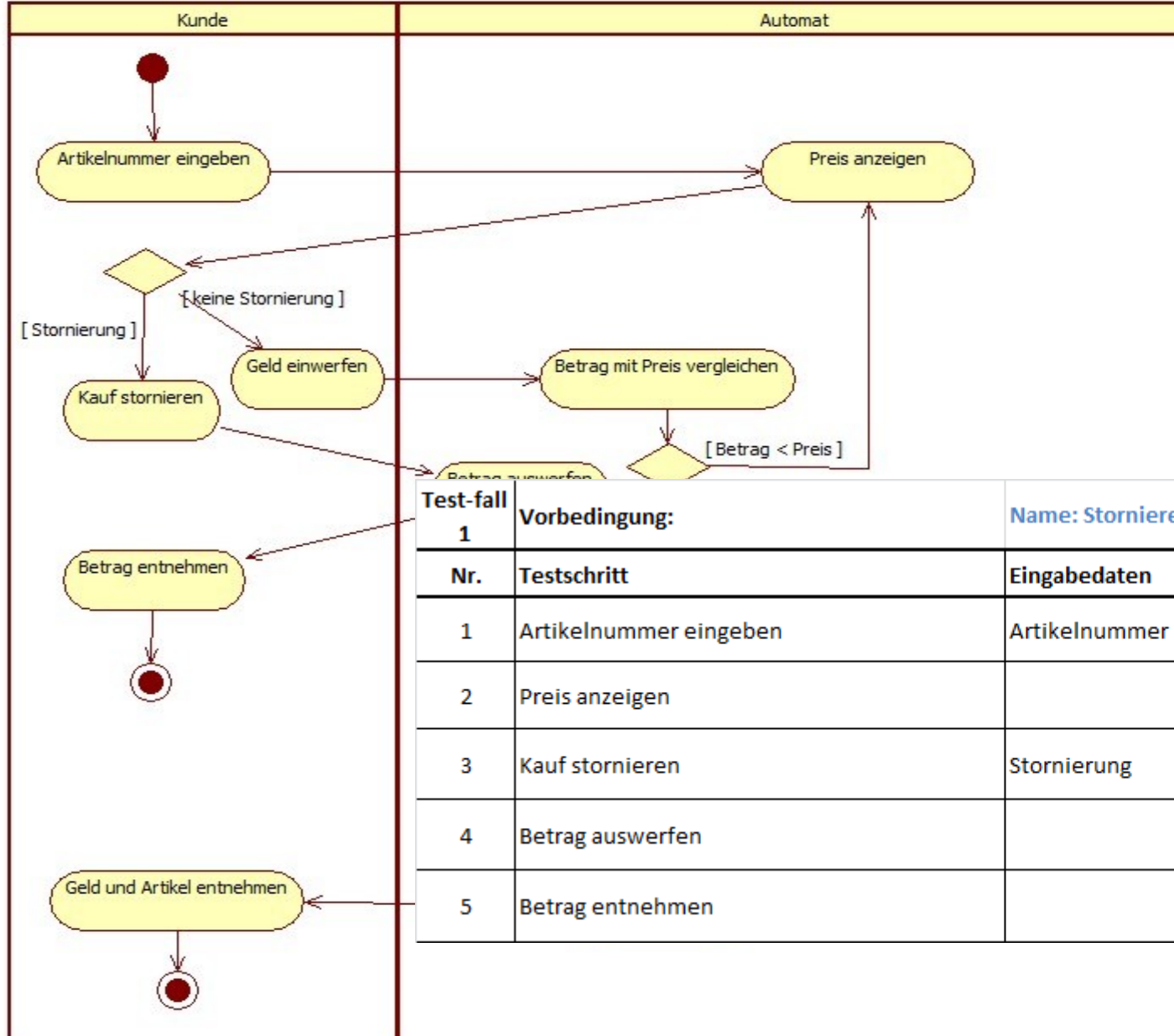
<sup>1</sup>Institut für Informatik, Universität Innsbruck  
A-6020 Innsbruck, michael.felderer@uibk.ac.at

<sup>2</sup>Freiberufliche Trainerin für Software Engineering,  
D-70372 Stuttgart, herrmann@herrmann-ehrlich.de

# Überblick des Vortrags

- Motivation
- Testbarkeit: Was ist das?
- Empirische Studie: Was braucht ein Testdesigner?
- Schlussfolgerungen für Testbarkeit von UML-Modellen

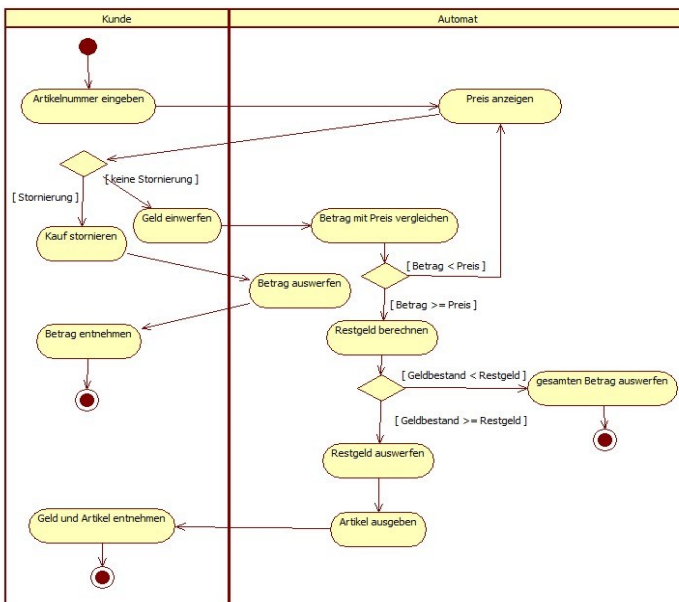
# Motivation



Test-fall 1		Vorbedingung:		Name: Stornieren	
Nr.	Testschritt	Eingabedaten	Soll-Ergebnis		
1	Artikelnummer eingeben	Artikelnummer			
2	Preis anzeigen		Preisanzeige		
3	Kauf stornieren	Stornierung			
4	Betrag auswerfen		Betrag ausgeworfen		
5	Betrag entnehmen				

# Motivation für modellbasiertes Testen

- Anforderungen = Grundlage für Entwicklung und Test:
  - nicht-testbare Anforderungen als Risiko
- Qualitätssicherung der Anforderungen:
  - Testbarkeit ist Qualitätskriterium für Anforderungen laut IEEE Standard 830-1998 und IREB
- -> frühe TestfallAbleitung sinnvoll




Test-fall	Vorbedingung:	Name: Stornieren	
Nr.	Testschritt	Eingabedaten	Soll-Ergebnis
1	Artikelnummer eingeben	Artikelnummer	
2	Preis anzeigen		Preisanzeige
3	Kauf stornieren	Stornierung	
4	Betrag auswerfen		Betrag ausgeworfen
5	Betrag entnehmen		

# Fragen

1. Wie definiert man Testbarkeit eines UML-Modells?
2. Wie stellt man sie sicher?

# Überblick des Vortrags

- 
- Motivation
  - Testbarkeit: Was ist das?
  - Empirische Studie: Was braucht ein Testdesigner?
  - Schlussfolgerungen für Testbarkeit von UML-Modellen

# Definition: Testbarkeit

**Testbarkeit** (eines UML-Modells): Es sind alle Informationen enthalten, die für die Ableitung des Testfalls nötig sind.

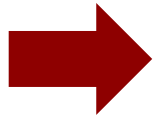
Test-fall 1	Vorbedingung:	Name: Stornieren	
Nr.	Testschritt	Eingabedaten	Soll-Ergebnis
1	Artikelnummer eingeben	Artikelnummer	
2	Preis anzeigen		Preisanzeige
3	Kauf stornieren	Stornierung	
4	Betrag auswerfen		Betrag ausgeworfen
5	Betrag entnehmen		

- **Inhalte:**
  - Vorbedingung, Aktionen, Eingabedaten und erwartetes Ergebnis
- **Detailtiefe:**
  - dieselbe Detailtiefe von Modell und Testfälle, z.B. Aktivität = Testschritt
- **Kontrollfluss:**
  - u.a. alle Fehler- und Sonderfälle



# Überblick des Vortrags

- Motivation
- Testbarkeit: Was ist das?
- Empirische Studie: Was braucht ein Testdesigner?
- Schlussfolgerungen für Testbarkeit von UML-Modellen



# Studentenexperiment

- 86 Teilnehmer/innen in 3 Gruppen an 2 Hochschulen
- Vorkenntnisse:
  - Domänenwissen (Getränkeautomat und Geldautomat)
  - UML aus vorigem Kurs
  - Testen: Einführung, Übungsbeispiel mit Musterlösung
- Testbarkeit der UML-Modelle:
  - Detailtiefe: wie Systemtests
  - Kontrollfluss: vollständig, alle Sonderfälle
  - Inhalte: Vorbedingungen, Eingabedaten und erwartete Ergebnisse nur teilweise explizit im Modell

# Testfall-Ableitung im Experiment

Aufgabe: manuelle Ableitung von Systemtests aus UML-Modellen

	Karlsruhe Group 1	Karlsruhe Group 2	Innsbruck
Treatment A	DVM-AD ATM-SM	DVM-AD ATM-SM	DVM-AD ATM-SM
Treatment B	DVM-SM ATM-AD	DVM-SM ATM-AD	DVM-SM ATM-AD

DVM = Getränkeautomat

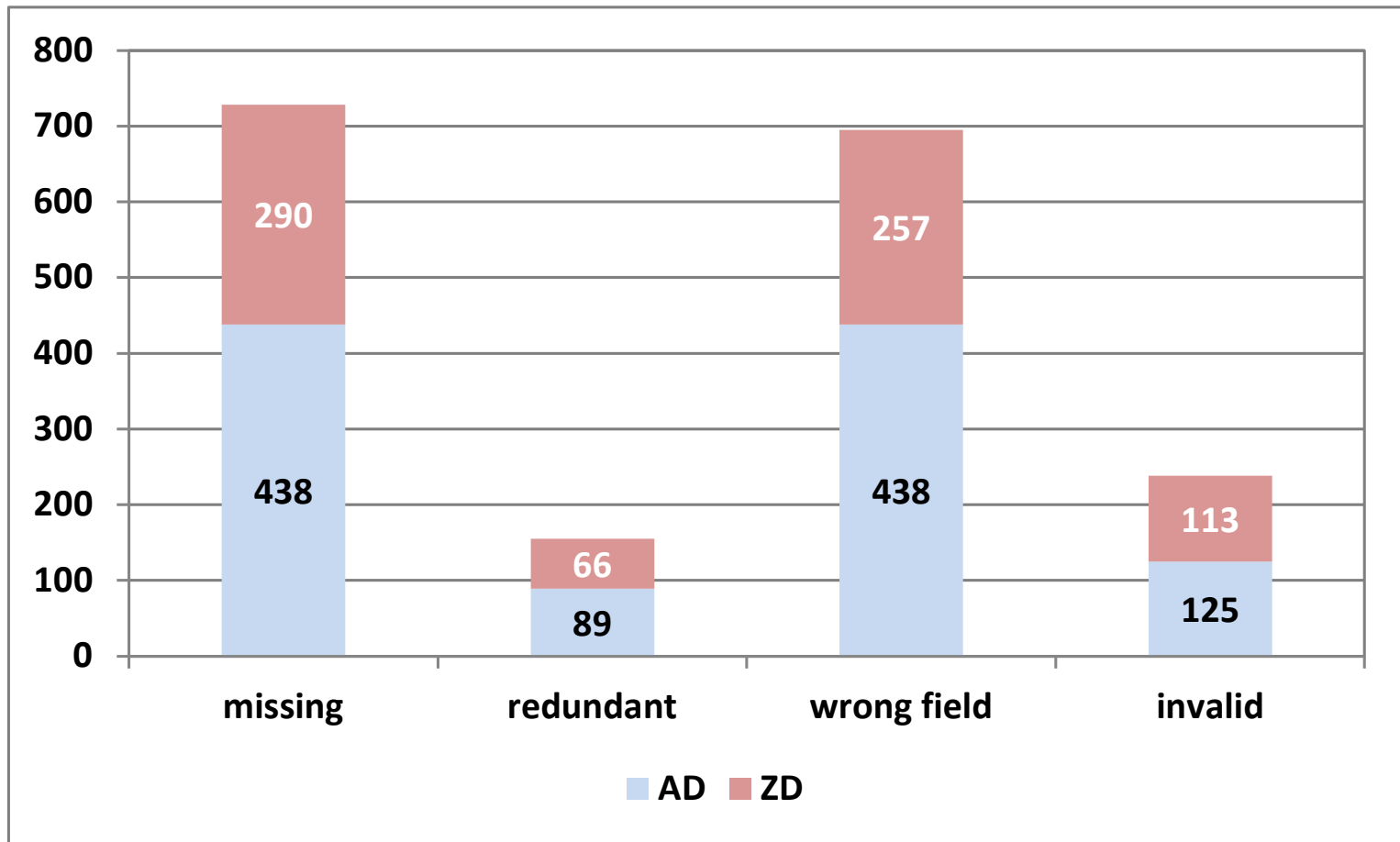
ATM = Geldautomat

AD = Aktivitätsdiagramm

SM = State Machine

# Ergebnisse: Wie viele und welche Fehler?

- 150 Sätze von Testfällen, ca. 340 Testfälle
- 1816 Fehler:

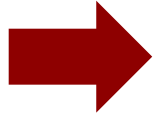


Aktivitätsdiagramm (vgl. Zustandsdiagramm):

- mehr Fehler
- Als verständlicher empfunden
- Besser verstanden (=bessere Antworten auf Verständnisfragen)

# Überblick des Vortrags

- Motivation
- Testbarkeit: Was ist das?
- Empirische Studie: Was braucht ein Testdesigner?
- Schlussfolgerungen für Testbarkeit von UML-Modellen



# Schlussfolgerungen

- UML-Modelle müssen vollständig sein, auf „intuitive“ Ergänzung sollte man sich nicht verlassen
- Vermutlich hilfreich: konkrete Ableitungsregeln wie „Eine Aktivität entspricht einem Testschritt.“
- Wahl des Diagramms:
  - Aktivitätsdiagramm besser geeignet für Anforderungskommunikation
  - Zustandsdiagramm für testbare Anforderungen

**Vielen Dank für Ihre Aufmerksamkeit!**

Fragen?