

---

# Testfallerzeugung aus Use-Case-Beschreibungen

---



**Fraunhofer** Institut  
Rechnerarchitektur  
und Softwaretechnik

---

---

# Testfallerzeugung aus Use-Case-Beschreibungen

---

Mario Friske

FhG FIRST

Synthese, Validierung und Test (SVT)

[mario.friske@first.fhg.de](mailto:mario.friske@first.fhg.de)

17./18. Juni 2004

---

---

## Testfallerzeugung aus Use-Case-Beschreibungen – Inhalt

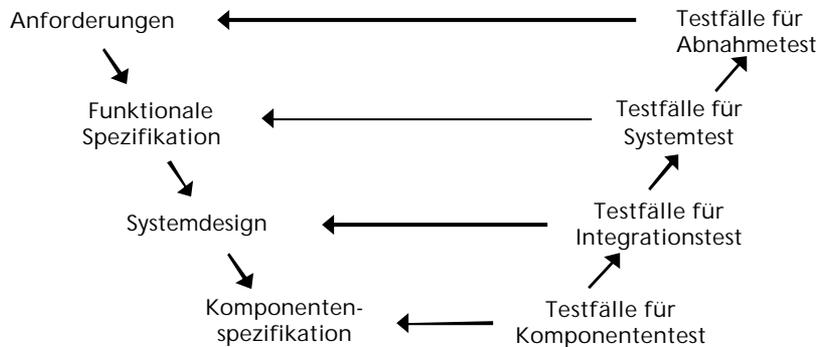
- Manuelle Testfallerstellung
- Model-Driven Testing
- MDA-basierte Werkzeugunterstützung

---

Seite 3

---

## V-Modell



---

Seite 4

---

## Beispiel einer Anwendungsfallbeschreibung [Schneider/Winters]

*Use Case 1 – Place Order*

### Brief Description

This use case describes the process by which orders are entered into the order processing system

### Actors

Customer

### Precondition

A valid user has logged into the system.

...

Seite 5

---

## Beispiel einer Anwendungsfallbeschreibung (Forts.)

### Basic Path

1. The use case starts when the customer selects Place Order.
2. The system displays the Place Order screen.
3. The customer enters his or her name and address.
4. The customer enters product codes for products to be ordered.

...

### Postcondition

The order has been saved in the system and marked confirmed.

Seite 6

---

## Beispiel eines logischen Testfalls

### *Logical Test Case 1 – Place Order*

#### Brief Description

This test case is used to test the process by which orders are entered into the order processing system

#### Traces

UC Place Order – Basic Path

#### Precondition

A valid user has logged into the system.

...

---

Seite 7

---

## Beispiel eines logischen Testfalls (Forts.)

#### Test Steps

<b><i>Nr.</i></b>	<b><i>Step</i></b>	<b><i>Expected Result</i></b>
1.	customer selects „Place Order“	system displays place order screen
2.	customer enters name and address	
3.	customers enters product codes	

---

Seite 8

---

## Zuordnung von System- und GUI-Funktionen

Systemfunktion	GUI Funktion
select place order screen	<ul style="list-style-type: none"><li>• Menü "Orders" -&gt; "Place Order"</li><li>• &lt;Alt+O&gt; - &lt;P&gt;</li><li>• Button "Place Order" in Toolbar</li></ul>
enter name and address	<ul style="list-style-type: none"><li>• Textfelder "Name" und "Address"</li></ul>
enter product codes	<ul style="list-style-type: none"><li>• Textfeld "Product Codes"</li><li>• Produktsuchwerkzeug</li></ul>

---

Seite 9

---

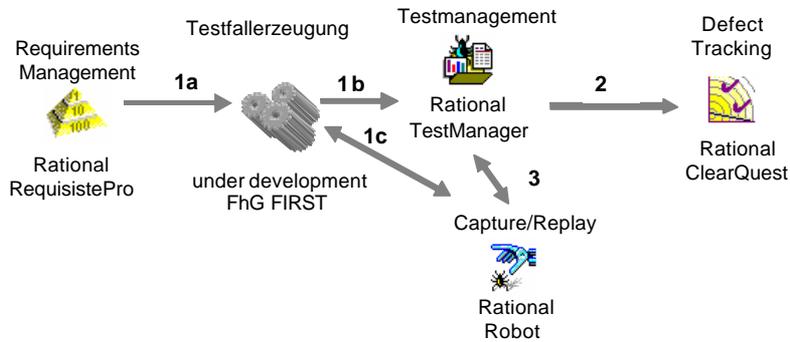
## Beispiel eines konkreten Testfalls

<i>Nr.</i>	<i>Step</i>	<i>Expected Result</i>
1.	customer selects menu "Orders" -> "Place Order"	system displays dialog "Place Order", as defined in ...
2.	customer writes 'John Doe' in the text field "Name" and 'Test Road 2' in the text field "Address"	
3.	customers writes '00123' in the text field "Product Codes"	

---

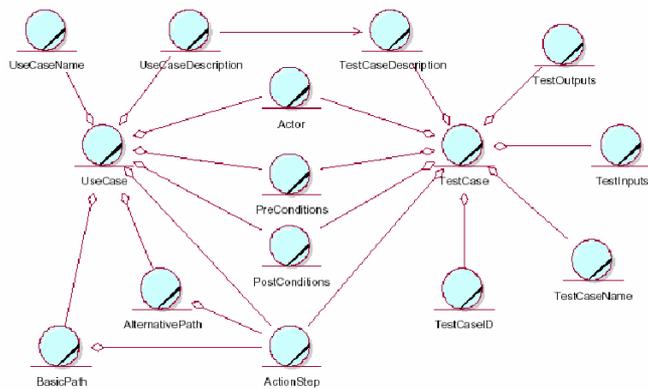
Seite 10

## Softwareentwicklungswerkzeugkette – inklusive werkzeugunterstützter Testfallerzeugung



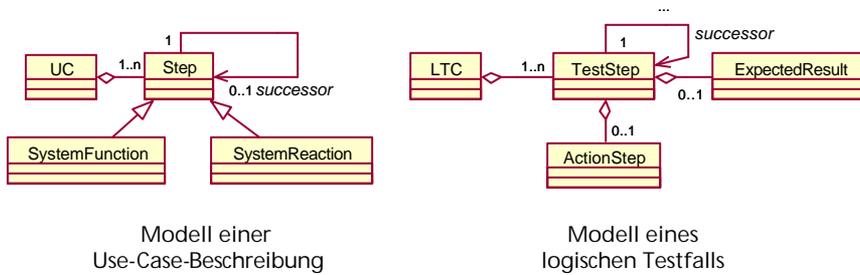
Seite 11

## Korrespondenzen in Use-Case- und Testfallbeschreibungen



Seite 12

## Umsetzung in Metamodelle



Seite 13

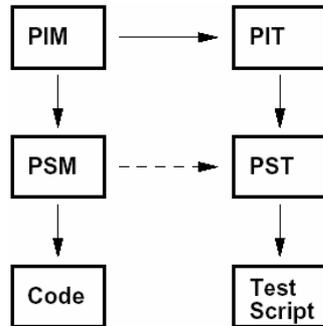
## Einordnung in die Modellebenen

Modellebene	Transformationsquelle	Transformationsziel
M3	MOF	MOF
M2	Modell der Use-Case-Beschreibung	Modell der Testfallbeschreibung
M1	Use-Case-Beschreibung	logischer Testfall
M0	Szenario	konkreter Testfall

Seite 14

---

## Model-Driven Testing



PIM – Platform Independent Model

PSM – Platform Specific Model

PIT – Platform Independent Test Case

PST –Platform Specific Test Case

---

Seite 15

---

## MDA-basierte Vorgehensweise

1. Metamodelle der Use-Case- und Testfallbeschreibungen erstellen
2. Systemfunktionen und Systemreaktionen identifizieren, Zuordnung von Systemfunktionen und GUI-Funktionen bestimmen
3. Use-Case-Beschreibungen durch interaktives Mapping mit Metadaten anreichern
4. Testfallerzeugungsalgorithmen in Form von (ggf. parametrisierbaren, alternativ selektierbaren) Transformationsregeln definieren
5. Gemäß gewünschter Strategie und Überdeckungskriterien Transformationsregeln selektieren und Transformationsparameter festlegen
6. Testfälle durch automatische Transformation erzeugen

---

Seite 16

---

## Anreicherung der Spezifikation durch interaktives Mapping

### Basic Path

1. The use case starts when the **customer** selects **Place Order**.
  2. The system **displays the Place Order screen**.
  3. The **customer** enters his or her name and address.
  4. The **customer** enters product codes for products to be ordered.
- ...

Step = (**Actor** + **System Function**) oder **System Reaction**

Die Anzahl der Systemfunktionen und -reaktionen ist begrenzt.

---

Seite 17

---

## Transformationsregeln - Beispiel

- Für jeden Schritt vom Typ *SystemFunction* im PIM wird ein *ActionStep* im PIT erzeugt.
- Für jeden Schritt vom Typ *SystemReaction* im PIM wird ein *ExpectedResult* im PIT erzeugt.
- Falls der Nachfolger eines Schrittes vom Typ *SystemFunction* ein Schritt vom Typ *SystemReaction* ist, dann werden beide dem gleichen Testschritt zugeordnet, ansonsten werden sie zwei aufeinanderfolgenden Testschritten zugeordnet.

---

Seite 18

---

## Zusammenfassung

- Testfallerstellung durch mehrstufige Transformationen (Use-Case-Beschreibung -> logischer Testfall -> konkreter Testfall -> Testskript)
- Nutzung der Verfahren und Werkzeuge der modellbasierten Entwicklung zur Testfallerzeugung
- Umsetzung von Testfallerzeugungsalgorithmen und Abdeckungskriterien in Transformationen
- flexibler Umgang mit unterschiedlichen Use-Case- und Testfallformaten durch anpassbare Metamodelle und Transformationen

---

Seite 19

---

## Status und Ausblick

Abgeschlossen

- Prozessmodellierung
- Anforderungsermittlung, Use-Case-Modell und Architekturmodell für Werkzeugunterstützung

Laufend

- Integration von Capture/Replay-Tools
- Umsetzung der MDA-basierten Transformationen

Geplant

- Integration in kommerzielle Werkzeugketten und Evaluation

---

Seite 20