

# Softwareanforderungsanalyse

## Akteure und ihre Verantwortlichkeiten

Burkhardt Renz

Institut für SoftwareArchitektur der Technischen Hochschule Mittelhessen

Wintersemester 2015/16

# Übersicht

- **Akteure**
  - Was sind Akteure?
  - Charakterisierung von Akteuren
- Darstellung des Modells der Akteure
- Heuristiken für das Entwickeln von Akteurmodellen

# Akteure und ihre Verantwortlichkeiten

Modellierung von Akteuren und ihrer Verantwortlichkeit ist die Dimension **Wer** in der Anforderungsanalyse:

- Akteure im System
- ihre Fähigkeiten und Eigenschaften
- wie diese zur Zielerreichung beitragen
- die Beziehungen und Abhängigkeiten von Akteuren untereinander

# Akteure

Was sind Akteure?

- Aktive Objekte, die Zustand und Verhalten des Systems beeinflussen, steuern
- als solche verantwortlich für das Erreichen von Zielen
- dazu müssen sie die entsprechenden Fähigkeiten haben
- man betrachtet eher Rollen statt individuelle Akteure
- Akteure handeln i.a. nebenläufig

# Kategorien von Akteuren

- Neu zu entwickelnde Software-Akteure  
z.B. Controller-Komponenten, Web-Services u.ä.
- Eingesetzte vorhandene Software-Akteure  
z.B. Datenbankmanagementsysteme, ERP-Systeme u.ä.
- Geräte  
z.B. Sensoren, Aktoren, Infrastruktur wie Router. . .
- Menschliche Akteure  
z.B. Anwender, Verantwortliche für den Betrieb,  
Organisationseinheiten. . .

# Eigenschaften von Akteuren

Wie jedes Objekt werden Akteure im Modell mit Annotationen versehen:

- Name = Name des Akteurs
- Def = Präzise Definition des Akteurs
- Category = Kategorie des Akteurs (s.o.)

# Fähigkeiten von Akteuren

- Akteur **beobachtet** (*monitors*) Attribute eines Objekts, wenn er die Werte der Attribute des Objekts lesen kann.
- Akteur **steuert** (*controls*) Attribute eines Objekts, wenn er die Werte der Attribute des Objekts ändern kann.
- Akteure haben untereinander **Schnittstellen**, wenn ein Akteur ein Attribut beobachtet, das ein anderer steuert.
- Ein Wert eines Attributs wird von höchstens **einem** Akteur (einer Rolle) gesteuert

# Verantwortlichkeiten von Akteuren

- Ein Akteur ist **verantwortlich** für ein Ziel, wenn seine Instanzen die einzigen sind, die das Verhalten des Systems so beeinflussen können, dass das Ziel erfüllt wird
- Das bedeutet, dass der Akteur durch die Attribute, die er beobachtet und/oder steuert eine **Folge von Zustandsänderungen** erreichen kann, die zu einem Verhalten des Systems führen, das dem Ziel entspricht.
- Ein Akteur kann **Funktionen** ausführen, die es ihm erlauben, Werte von Variablen zu beeinflussen.





# Wünsche und Annahmen eines Akteurs

- Ein Akteur **wünscht**, dass ein Ziel erreicht wird, dies kann im Modell durch eine **Wunsch-Beziehung** dargestellt werden.
- Ein Akteur hat **Annahmen** über seine Umgebung und deren aktuellen Zustand.

Dies wird modelliert, indem man dem Akteur einen **lokalen Speicher** gibt.

Man sagt dann, dass der Akteur an die Fakten *glaubt*, die er in seinem lokalen Speicher hat.

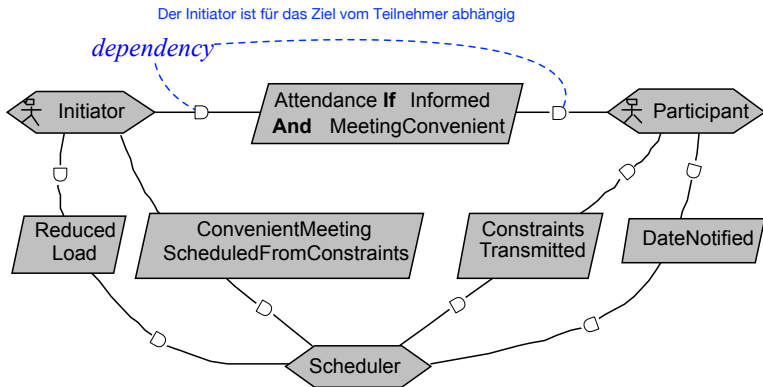
Spezielle Formen für die Risikoanalyse sind

- **Konfusion** = der Akteur irrt sich über die Identität eines Objekts
- **Fehlglaube** = die Werte im lokalen Speicher des Akteurs stimmen nicht mit den wirklichen Fakten überein.

# Abhängigkeiten von Akteuren

Ein Akteur hängt in der Verantwortlichkeit für ein Ziel von einem anderen Akteur ab, wenn dessen Versagen dazu führen kann, dass er sein Ziel nicht erreicht.

Darstellung im Akteursdiagramm:



# Übersicht

- Akteure
- Darstellung des Modells der Akteure
  - Akteursdiagramme
  - Kontextdiagramme
  - Abhängigkeitsdiagramme
- Heuristiken für das Entwickeln von Akteurmodellen

# Diagramme des Akteursmodells

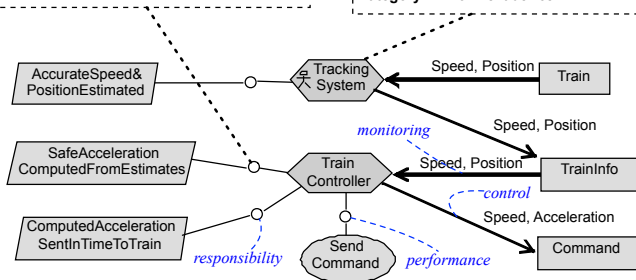
Im Akteursmodell werden 3 Arten von Diagrammen verwendet

- **Akteursdiagramm** – zeigt Akteure und ihre Fähigkeiten, Verantwortlichkeiten und Funktionen
- **Kontextdiagramm** – zeigt Akteure und ihre Schnittstellen
- **Abhängigkeitsdiagramm** – zeigt Akteure und ihre Abhängigkeiten

# Akteursdiagramme

**InstanceResponsibility** A train controller at a station is responsible for computing safe accelerations of all trains between this station and the next one

**Def** A set of sensors distributed along blocks to measure the physical speed and position of trains

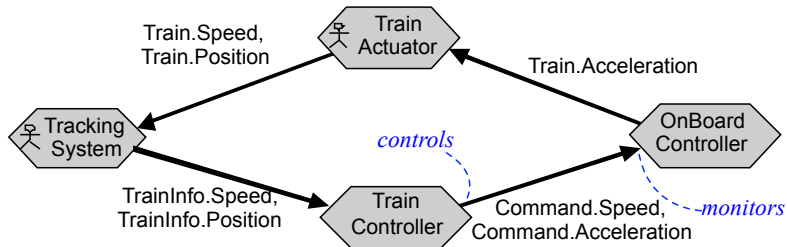
**Category** Environment device

Quelle: Lamsweerde S.406

# Kontextdiagramme

- **Akteure** sind die Knoten des Graphen
- Eine gerichtete Kante beschreibt wie ein Akteur einen anderen **beeinflusst**
- Die Kanten werden beschriftet mit den beeinflussten Attributen aus dem Objektmodell
- Das Kontextdiagramm kann aus dem Akteursdiagramm generiert werden

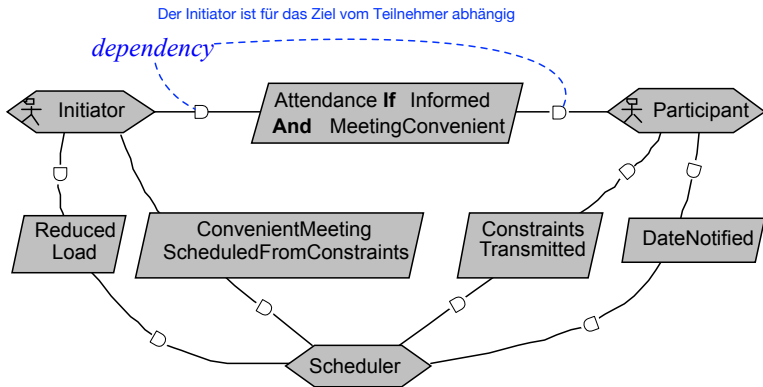
# Kontextdiagramm – Beispiel



Quelle: Lamsweerde S.407



# Abhängigkeitsdiagramme



Quelle: Lamsweerde S.408

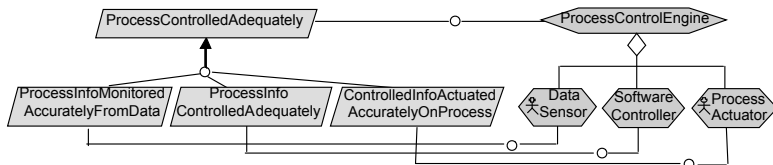
# Übersicht

- Akteure
- Darstellung des Modells der Akteure
- Heuristiken für das Entwickeln von Akteurmodellen
  - Abstrakte Akteure und ihre Verfeinerung
  - Von Zielen zu Akteuren
  - Von Zielen zu Verantwortlichkeiten
  - Kontextdiagramme aus dem Zielemodell generieren

# Abstrakte Akteure und ihre Verfeinerung

- Oft kann man nicht im ersten Schritt die Akteure so genau bestimmen, dass sie für feingranulare Ziele verantwortlich sind
- Man verwendet dann **abstrakte Akteure**, die man dann verfeinert
- Dabei kann man oft das Muster der **parallelen Verfeinerung** von Zielen und Akteuren, die für sie verantwortlich sind, anwenden

## Beispiel



# Ableitung von Akteuren aus Zielen

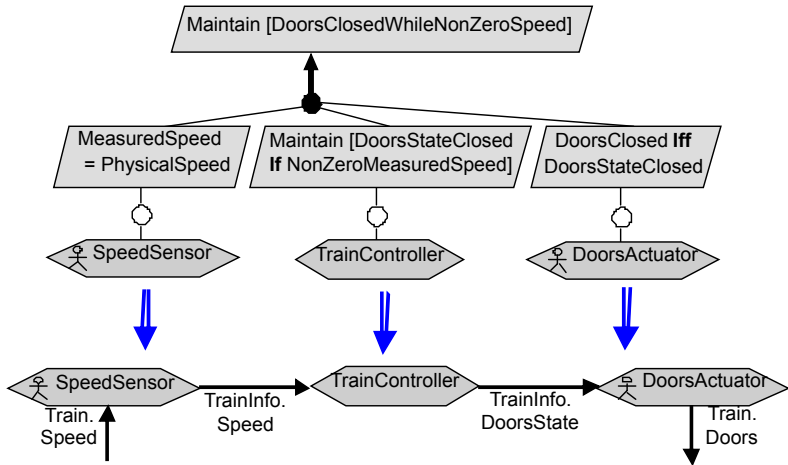
- Analyse aller Objekte in der Formulierung eines Ziels – Herausfinden der **aktiven** Objekte
- Zuordnen von Akteuren zu Zielen, in dem man untersucht, wer für das Erreichen eines Ziels **verantwortlich** sein kann
- Untersuchen welcher Akteur sich ein Ziel **wünscht**
- Analysieren, wer Attribute/Objekte beobachtet und/oder steuert

# Zuordnung von Verantwortlichkeiten

- Untersuchen, wer **fähig** ist, ein Ziel zu erreichen
- Verfeinerung von Zielen mit **paralleler Verfeinerung** von Akteuren (s.o.)
- Kann man sich Softwareakteure vorstellen, die menschliche Akteure **ersetzen**?
- Ist der Akteur, der ein Ziel **wünscht** auch für es verantwortlich?

# Kontextdiagramme aus dem Zielemodell generieren

Es ist möglich, Kontextdiagramme aus dem Ziele- bzw. Akteursmodell herzuleiten – Beispiel zur Zugsteuerung:



# Kontextdiagramme aus dem Zielemodell generieren

Es ist möglich, Kontextdiagramme aus dem Ziele- bzw. Akteursmodell herzuleiten – Beispiel zur Bibliotheksverwaltung:

