

Softwareanforderungsanalyse

Ziele und das Zielemodell

Burkhardt Renz

THM, Fachbereich MNI

Wintersemester 2018/19

Übersicht

- Ziele
 - Ziele und Anforderungen
 - Klassifikationen von Zielen
- Zielorientierter Ansatz zur Anforderungsanalyse
- Zielemodell
- Exkurs zu Zielemodell & Problem Frames

Was sind Ziele?

Definition (Akteur *agent*)

Ein **Akteur** ist eine aktive Komponente eines Systems, z.B. Softwarekomponenten, Geräte (Sensoren, Aktoren), Anwender

Definition (Ziel *goal*)

Ein **Ziel** ist eine präskriptive Aussage, welche Effekte ein System durch die Kooperation seiner Akteure im Anwendungsgebiet erreichen soll.

Beispiele von Zielen

Beispiel (Bibliotheksverwaltung)

Studenten sollen Bücher des eigenen Fachbereichs und von anderen Bibliotheken bestellen können.

Beispiel (Zugsteuerung)

Das System soll sicheren Transport der Passagiere garantieren.

Beispiel (Terminvereinbarungen)

Terminvereinbarungen sollen so gelegt werden, dass möglichst viele Teilnehmer anwesend sein können.

Ziele und Eigenschaften des Anwendungsgebiets

- Ziel ist präskriptiv
- Eigenschaft ist deskriptiv, also im Indikativ

Unterschied ist wesentlich:

- Ziele kann man diskutieren, verhandeln, verändern
- Eigenschaften sind i.a. **gegeben**, etwa durch physikalische Gesetze, Regeln einer Organisation etc
- In der Anforderungsanalyse werden beide Arten von Aussagen benötigt, wichtig sie zu **unterscheiden**

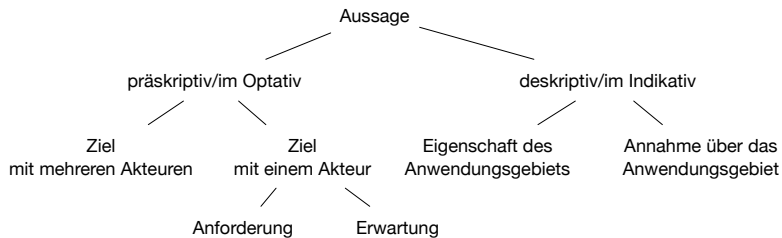
Granularität von Zielen

- Höhere Ebene – strategische Ziele
z.B. „50% Steigerung der Transportkapazität der Züge“
- Niedrigere Ebene – detaillierte, technischer
z.B. „Wenn das Signal auf ‚frei‘ wechselt, soll der Zug möglichst schnell in das nächste Segment fahren“
- Je niedriger die Ebene ist, desto weniger Akteure sind an der Erreichung des Ziels beteiligt

Ziele, Anforderungen, Erwartungen, Eigenschaften

- **Ziel** = gewünschtes/gefordertes Verhalten für dessen Erreichung mehrere Akteure verantwortlich sind
- **(Software)anforderung** = Ziel, für das ein einzelner Akteur in der zu konstruierenden Software verantwortlich ist
„doorState = 'closed' solange measuredSpeed > 0“ – verantwortlich die Zugsteuerung
- **Erwartung** = Ziel, für das ein einzelner Akteur im Anwendungsgebiet verantwortlich ist
„Passagiere steigen aus, wenn der Zug sein Ziel erreicht hat“ – verantwortlich die Passagiere
- **Eigenschaft** = deskriptive Aussage über das Anwendungsgebiet, die unabhängig von dem zu konstruierenden System gilt
„Ein Zug bewegt sich, wenn seine Geschwindigkeit $v \neq 0$ ist“ – physikalische Tatsache

Arten von Aussagen



Typen von Zielen

- **Weiches Ziel** (*soft goal*) = Präferenz unter Alternativen des Verhaltens des Systems
z.B. „Für die Vereinbarung der Termine sollen möglichst wenig Interaktionsschritte notwendig sein“
 - **Improve**
 - **Increase, Reduce**
 - **Maximize, Minimize**
- **Verhaltensziel** (*behavioural goal*) = deklarative Beschreibung des Verhaltens des Systems
z.B. „Achieve[BookRequestSatisfied]: Wenn ein Buch bestellt wird, wird ein Exemplar früher oder später für die Ausleihe verfügbar sein“
 - **Achieve** bzw. **Cease**
wird irgendwann erreicht oder (dual) nicht mehr gültig sein
 - **Maintain** bzw. **Avoid**
ist immer so oder (dual) niemals so

Kategorien von Zielen

- funktionale Ziele
intendierte Dienste, die das System erfüllen soll
- nicht-funktionale Ziele, die die Anwender betreffen
Qualitätsmerkmale der zur Verfügung gestellten Funktionen
- nicht-funktionale Ziele, die die Entwicklung betreffen
Qualitätsmerkmale der Entwicklung

Übersicht

- Ziele
- Zielorientierter Ansatz zur Anforderungsanalyse
- Zielemodell
- Exkurs zu Zielemodell & Problem Frames

Vorgehen

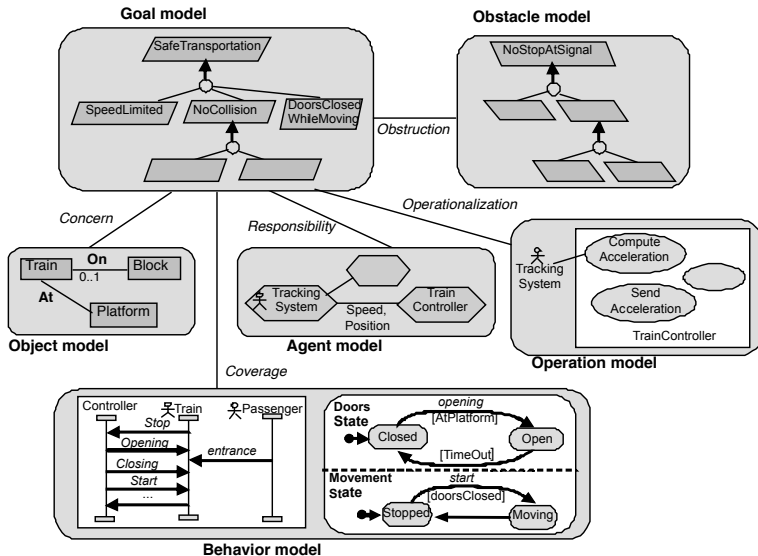
- **zielorientiert** = die Warum-Dimension!
- **modellbasiert** = abstrakte Repräsentation des (zu konstruierenden) Systems
- **mehrere Sichten**
- **KAOS** – Beispiel für das Vorgehen
KAOS = „Knowledge Acquisition in autOmated Specification“
KAOS = „Keep All Objectives Satisfied“

Sichten/Modelle

Sicht auf ...

- **Intentionen**: Zielemodell und Risikoanalyse im Hindernismodell
- **Struktur**: Objektmodell
- **Verantwortlichkeit**: Akteure und ihre Verantwortlichkeiten
- **Funktionalität**: Modell der Operationen
- **Verhalten**: Modell des Verhaltens

Übersicht über die Modelle und ihre Beziehungen



Übersicht

- Ziele
- Zielorientierter Ansatz zur Anforderungsanalyse
- **Zielemodell**
 - Spezifikation von Zielen
 - Verfeinerung von Zielen
 - Alternative Möglichkeiten
 - Regeln für die Entwicklung von Zielemodellen
 - Muster der Verfeinerung von Zielen
- Exkurs zu Zielemodell & Problem Frames

Ziele und Zielemodell

Definition (Zielemodell)

Das **Zielemodell** zeigt, wie die funktionalen und nicht-funktionalen Ziele durch Verfeinerung bis hin zu Softwareanforderungen und Erwartungen an Akteure der Umgebung gegliedert sind.

Eigenschaften des Zielemodells

- Spezifikation von Ziele
- Zielediagramm
- Erfüllbarkeitsargument

Spezifikation von Zielen

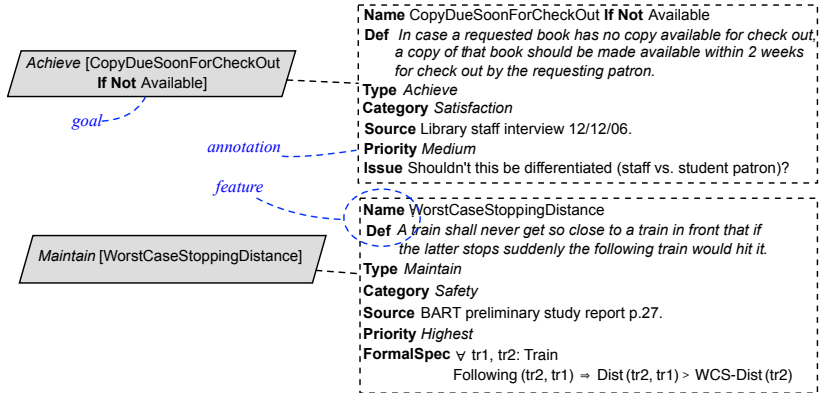
Jedes Ziel wird im Modell durch Eigenschaften annotiert:

- Name – eindeutige Bezeichnung
- Def – präzise textuelle Beschreibung

optional:

- Type – z.B. Achieve ...
- Category – z.B. Safety ...
- Source – Herkunft des Ziels
- Priority
- Stability
- FitCriterion – wie stellt man fest, ob erfüllt?
- FormalSpec – formale Spezifikation
- Issue – Fragen, Probleme

Beispiele

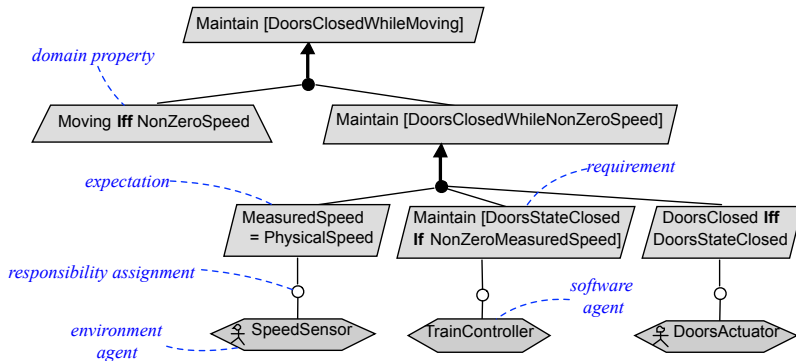


Quelle: Lamsweerde S.295

AND-Verfeinerung

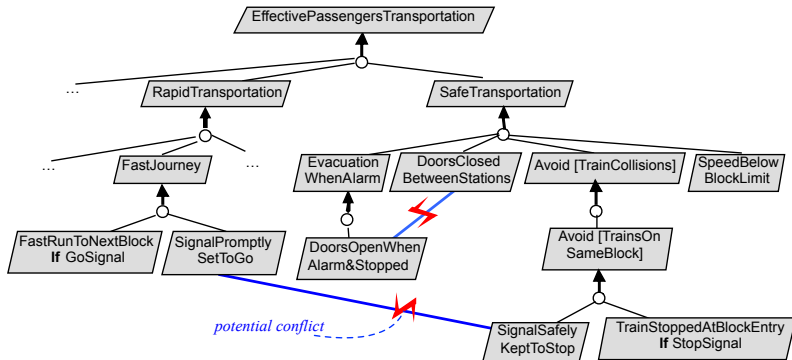
- Ziele werden in Subziele aufgegliedert
- Die Subziele verfeinern ein Ziel
- Ein Ziel wird erreicht, wenn alle Subziele erreicht werden (Konjunktion)
- AND-Verfeinerung sollten sein
 - vollständig
 - konsistent
 - minimal
- Blätter sind Anforderungen oder Erwartungen und können einem Akteur zugeordnet werden

Beispiel AND-Verfeinerung



Quelle: Lamsweerde S.299

Beispiel mit potentiellen Konflikten

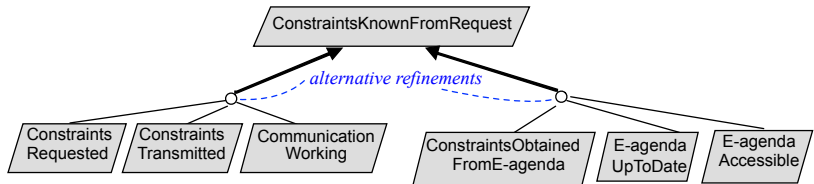


Quelle: Lamsweerde S.302

Alternative Möglichkeiten (OR)

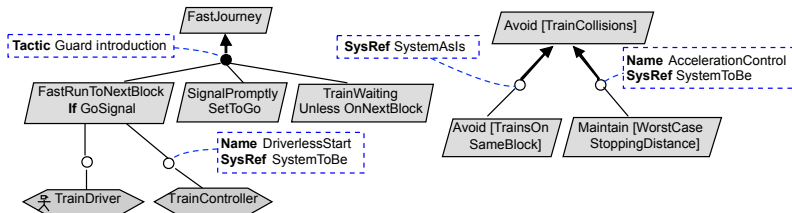
- Ziele können mehrere Verfeinerungen haben
- Das Ziel kann dann durch alle Subziele jeweils einer der Alternativen erreicht werden
- Mehrfache Verfeinerungen bedeuten **verschiedene** Systeme, die dasselbe Ziel erreichen
- Folglich können auch jeweils unterschiedliche Akteure verantwortlich sein

Beispiel (Terminplaner)



Quelle: Lamsweerde S.8.8

Annotationen in Zielediagrammen



Quelle: Lamsweerde S. 308

Heuristische Regeln für Zielemodelle, 1

- Ziele und Probleme vorhandener Systeme untersuchen
- Begriffe, die auf Ziele hindeuten in Dokumenten suchen
- Kategorien von Zielen als Basis der Ermittlung nehmen
- Wie-Fragen → Verfeinerung von Zielen
Warum-Fragen → Allgemeinere Ziele finden
- Verantwortlichkeiten aufteilen → Verfeinerung von Zielen
- Weiche Ziele finden durch die Analyse von Pro und Contra bei Alternativen

Heuristische Regeln für Zielemodelle, 2

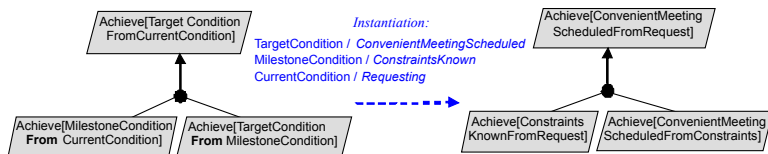
- Wünsche von Akteuren ermitteln
- Umkehrung eines Achieve-Ziels prüfen
- Komplementären Fall eines bedingten Achieve-Ziels prüfen
- Verfeinerung bis die Zuordnung zu einem Akteur möglich ist
- Umgekehrt: Verallgemeinerung von Zielen bis zur Systemgrenze
- Ziele nicht mit Operationen/Funktionen verwechseln
- AND- und OR-Verfeinerungen nicht verwechseln

Verfeinerung durch Teilschritte

Idee

Zerlege ein Ziel in Teilziele, die sukzessive zum Erreichen des Ziels führen

Beispiel



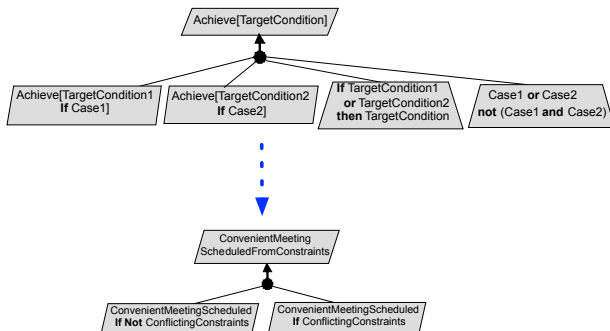
Quelle: Lamsweerde S.320

Verfeinerung durch Fallunterscheidung

Idee

Zerlege ein Ziel in verschiedene Fälle, die zusammen zum Erreichen des Ziels führen

Beispiel



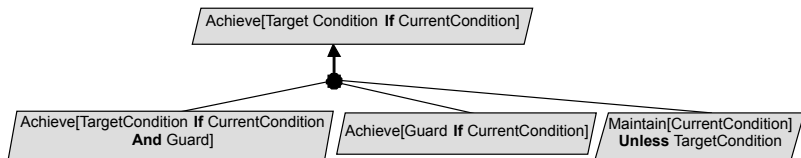
Quelle: Lamsweerde S.323

Verfeinerung durch Einführen einer Wächterbedingung

Idee

Führe zu einem Ziel vom Typ Achieve eine Wächterbedingung ein, deren Erreichen für das Erreichen des Ziels erforderlich ist

Beispiel



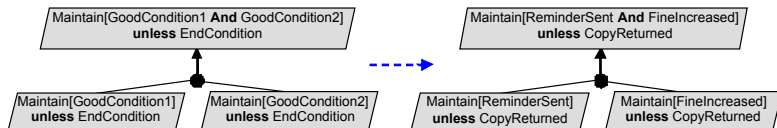
Quelle: Lamsweerde S. 324

Verfeinerung durch *Teile und herrsche*

Idee

Ein Ziel vom Typ Maintain wird erreicht, indem man Teilziele erreicht, die zusammen das Ziel ergeben

Beispiel



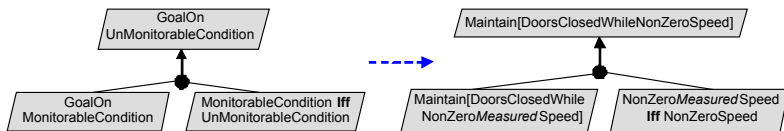
Quelle: Lamsweerde S. 324

Verfeinerung durch Einführen einer beobachtbaren Bedingung

Idee

Ist ein Ziel abhängig von einer Bedingung, die ein Akteur nicht beobachten kann, führt man eine beobachtbare Bedingung ein, die indirekt zum Erreichen des Ziels führt

Beispiel



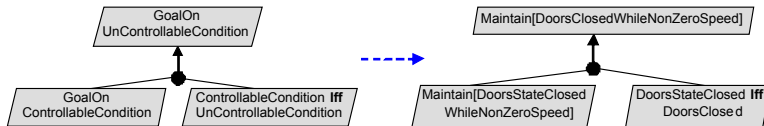
Quelle: Lamsweerde S. 325

Verfeinerung durch Einführen einer steuerbaren Bedingung

Idee

Ist ein Ziel abhängig von einer Bedingung, die ein Akteur nicht kontrollieren kann, führt man eine steuerbare Bedingung ein, die indirekt zum Erreichen des Ziels führt

Beispiel



Quelle: Lamsweerde S. 326

Übersicht

- Ziele
- Zielorientierter Ansatz zur Anforderungsanalyse
- Zielemodell
- Exkurs zu Zielemodell & Problem Frames

Das Problem Frame: Anzeige von Information (*Information Display*)

Ein Problem Frame aus dem Buch von Michael Jackson *Problem Frames: Analysing and Structuring Software Development Problems*

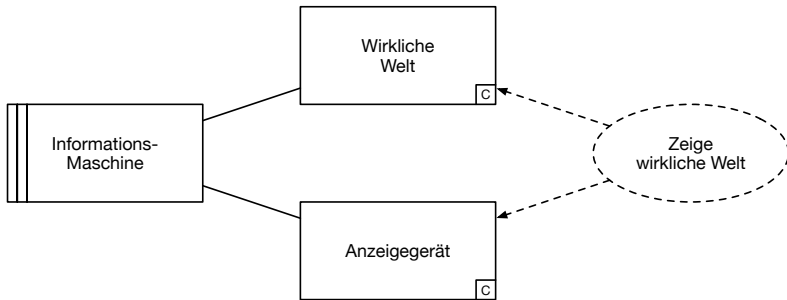
Problemstellung

Es gibt einen Teil der wirklichen Welt, über dessen Zustände und Verhalten fortlaufend gewisse Informationen gebraucht werden. Das Problem ist, eine Maschine zu bauen, die diese Informationen ermittelt und am verlangten Ort und in der verlangten Form darstellt.

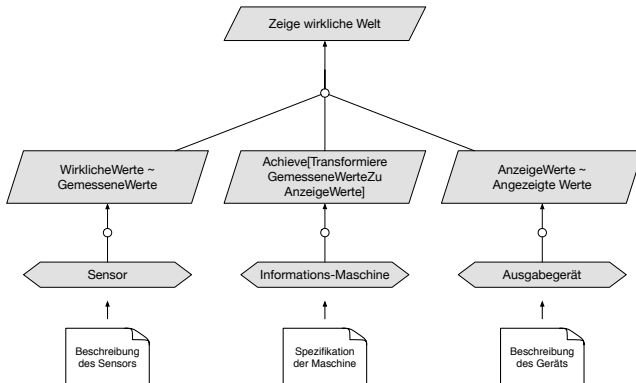
Beispiel

- Tachometer und Kilometerzähler

Problem Frame Anzeige von Information



Zielemodell zu diesem (einfachen) Problem



Argumentation bei der „Informationsanzeige“

