

## 2 Systeme und ihr Kontext

### 2.1 Definitionen

**System (system).** 1. Ein Ordnungs- und Gliederungsprinzip. 2. Eine zusammengehörige, von ihrer Umgebung abgrenzbare Menge von Komponenten und Wechselwirkungen, die beobachtete oder postulierte Phänomene der Realität beschreiben. 3. Eine zusammengehörige, von ihrer Umgebung abgrenzbare Menge von Komponenten, die durch koordiniertes Zusammenwirken Leistungen erbringen.

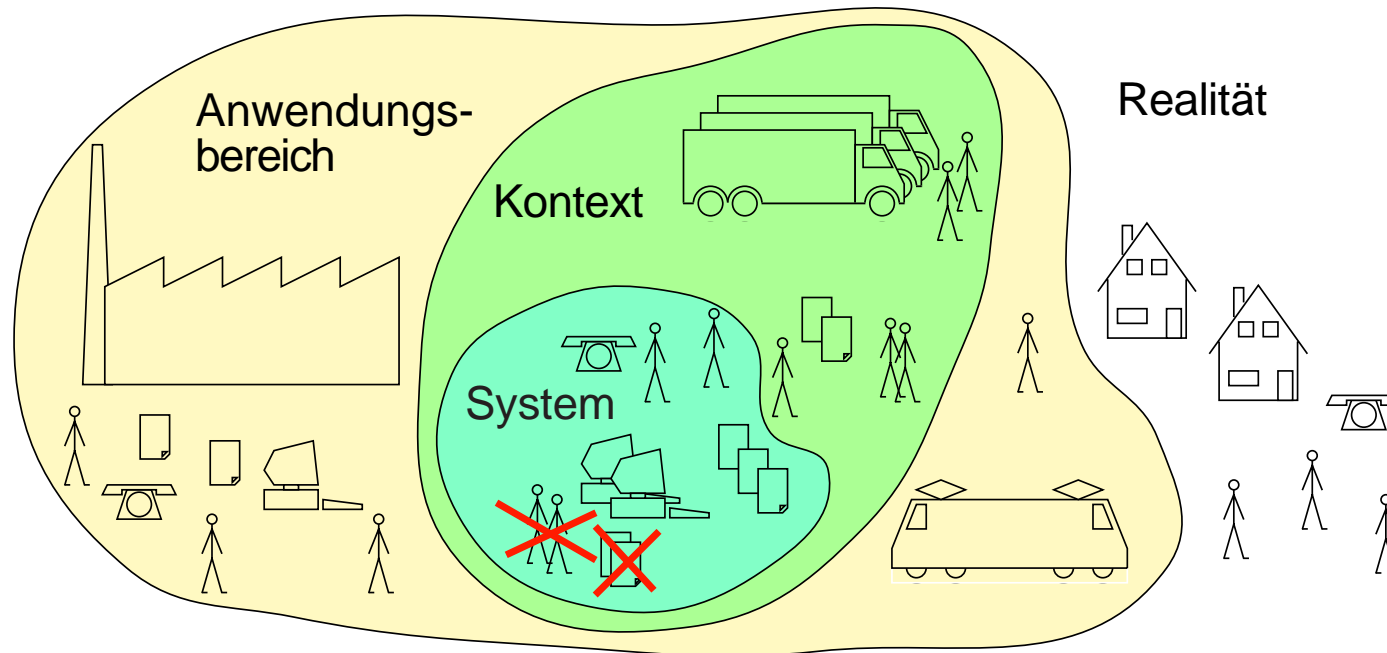
Nicht näher bezeichnete Systeme in der Informatik sind vom Typ 3., wobei die Komponenten aus Hard- und Software bestehen.

**Kontext (context).** 1. Der umgebende Text einer sprachlichen Einheit. 2. Der Gedanken- und Sinnzusammenhang, in dem ein Phänomen oder eine Äußerung verstanden werden muss. 3. (in der Informatik). Diejenigen Komponenten des Anwendungsbereichs, welche mit einem System interagieren, aber selbst nicht Bestandteil des Systems sind.

**Anwendungsbereich (application domain).** Derjenige Ausschnitt der Realität, welcher im Zusammenhang mit einem in diesem Bereich eingesetzten System relevant ist.

**Realität.** Die Wirklichkeit: Alle beobachtbaren oder sonstwie gedanklich fassbaren Dinge (Gegenstände, Personen, Phänomene, Handlungen, Ereignisse, Gedanken...) dieser Welt.

## 2.2 Einbettung und Abgrenzung



- Requirements Engineering bildet Modelle eines **Ausschnitts der Realität**
- Sobald ein System ein Problem der Realität bearbeitet, ist es **eingebettet** in die Realität und besitzt einen Kontext
- Requirements Engineering legt die **Grenzen** eines geplanten Systems fest
- Systeme **verändern** durch ihren Einsatz **die Realität**

## 2.3 Betrachtungsebenen

- ☆ Es gibt mindestens drei **Betrachtungsebenen** mit unterschiedlichen Systemgrenzen:
  - **Geschäft** «Auf dem bestehenden Schienennetz sollen mehr Leute transportiert werden»
  - **System** «Die Minimaldistanz zwischen zwei Zügen ist immer größer als der maximale Bremsweg des nachfolgenden Zuges.»
  - **Software** «Der maximale Bremsweg muss alle 100 ms neu berechnet werden.»
- ☆ Faktisch oft n Ebenen
- ☆ Kontextbestimmung und -abgrenzung erfordert Klarheit über die aktuelle Betrachtungsebene
- ☆ **Verzahnung von Anforderungen und Lösungen ist unausweichlich**
- ⇒ Problem der **Abgrenzung** von Anforderungen und Entwurf

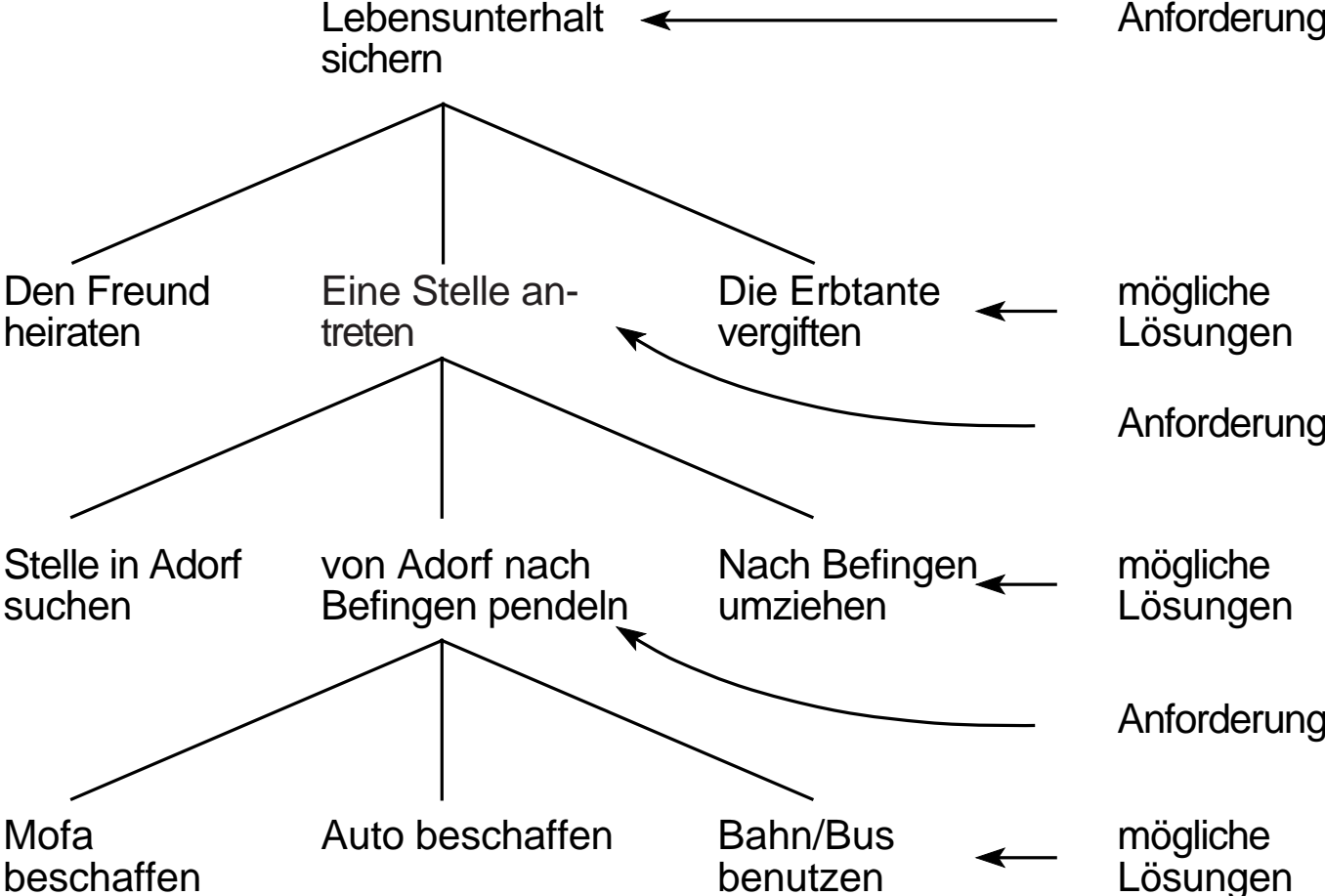
## 2.4 WAS versus WIE im Requirements Engineering

Volksweisheit: WAS = Spezifikation, WIE = Entwurf

- ☆ Aber: ist folgender Satz eine Anforderung oder eine Entwurfsentscheidung?  
„Das System druckt eine wahlweise nach Namen oder Land alphabetisch sortierte Liste von Teilnehmern mit Nummer, Name, Vorname, Affiliation und Land. Auf jeder Seite sind unten links das Erstellungsdatum und unten rechts die Seitenzahl aufgedruckt.“
- WAS vs. WIE ist **kontextabhängig** und liefert **keine brauchbare Abgrenzung** zwischen Anforderungen und Entwurfsentscheidungen. Die gleiche Sache kann je nach Kontext beides sein.
- ☆ Probleme (beschrieben als Anforderungen) und Lösungen (das Ergebnis von Entwurfsentscheidungen) sind hierarchisch miteinander verzahnt
- WAS vs. WIE ist **stufenabhängig**: Eine Entwurfsentscheidung auf Stufe n wird zur Aufgabe (=Anforderung) auf Stufe n+1

# Hierarchische Verzahnung von Problem und Lösung

Problem: Sonja Müller hat ihr Studium abgeschlossen und erhält keine Unterstützung von ihren Eltern mehr. Sie ist daher mit der Anforderung konfrontiert, ihren Lebensunterhalt zu sichern. Sie wohnt in Adorf und hat ein Stellenangebot bei einer Firma in Befingen. Ferner hat sie einen reichen Freund und eine ebenso reiche Erbtante.



# Sind Spezifikation von Anforderungen und Entwerfen von Lösungen voneinander trennbar?

Ja, mit *operationaler* Abgrenzung:

- Änderungen der **Anforderungen** brauchen die Zustimmung des Auftraggebers/Kunden
  - Änderungen im **Entwurf** kann der Auftragnehmer/Lieferant autonom vornehmen
- ⇒ Also: **Braucht eine Veränderung eines Satzes, eines Modellelements, eines Konstrukts, etc. die Zustimmung des Auftraggebers/Kunden?**

ja → Anforderung

nein → Entwurfsentscheidung

# Ist die Trennung notwendig?

Ja.

- Die Kompetenz zur Festlegung von Anforderungen liegt fast immer bei anderen Personen als die Kompetenz zum Treffen von Entwurfsentscheidungen
- Anforderungen und Entwürfe sind getrennt zu dokumentieren



# Ist eine strenge Trennung im Entwicklungsprozess möglich?

Nein.

- Hierarchische Verzahnung: Übergeordnete Entwurfsentscheidungen beeinflussen untergeordnete Anforderungen
- Nicht realisierbare Anforderungen sind sinnlos → Technische Machbarkeit (d.h. Lösungen) beeinflusst die Anforderungen
- Machbarkeitsstudien, die häufig am Anfang von Projekten durchgeführt werden, müssen auf der Grundlage der Kernanforderungen die Realisierbarkeit und Wirtschaftlichkeit möglicher Lösungen abschätzen
- Validierung: Anforderungen sind oft nur mit Hilfe von Lösungen (z.B. Prototypen) beurteilbar und validierbar
- Erkenntnisse und Schwierigkeiten bei der Lösung können Änderungen in den Anforderungen bewirken

## 2.5 Kontextmodelle

Modell eines Systems in seinem **Kontext**

- **Betrachtungsebene** festlegen
- **Keine Systeminterna** (→ System als schwarzer Kasten)
- **Kontextelemente** bestimmen (Kontextelement = Subjekt oder Objekt der Systemumgebung, das mit dem System interagiert)
- **Interaktion** zwischen **System** und **Kontextelementen** modellieren
- **Interaktion** von **Kontextelementen untereinander** modellieren
- Ergebnis ggf. grafisch darstellen

# Beispiel: Kontexteinbettung eines Fuhrpark-Dispositionssystems

