

3 Modellierungssprachen

3.1 Grundidee

Anforderungen und Entwürfe nicht als eine Sammlung von Sätzen in natürlicher Sprache beschreiben,

sondern

- ☆ zur **Spezifikation von Anforderungen** ein **anwendungsorientiertes Modell** der **Aufgabenstellung** erstellen
- ☆ im **Entwurf** konzeptionelle und/oder physische **Modelle** der geplanten **Lösung** erstellen

- formale oder teilformale Modelle möglich
- meistens (zumindest teilweise) grafisch repräsentierte Modelle
- Meistens nur für funktionale Anforderungen / Architekturen

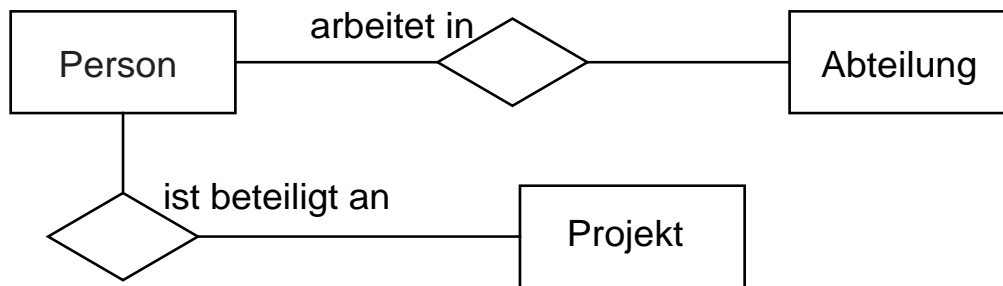
3.2 Modellierungsaspekte

- **Statische Struktur**
 - Datenmodelle (Gegenstands-Beziehungs-/ Entity-Relationshipmodelle)
 - Klassen- und Objektmodelle
 - Komponenten-Konnektorenmodelle
 - Paket- und Verteilungsmodelle
- **Interaktion**
 - Anwendungsfall- / Szenarienmodelle
 - Sequenz- und Zusammenarbeitsmodelle
- **Verhalten**
 - Zustandsmodelle
 - Aktivitätsmodelle
- **Funktionen, Flüsse und Abläufe**
 - Funktionsmodelle
 - Datenfluss-, Informationsfluss- und Arbeitsflussmodelle
 - Ablaufmodelle

3.3 Ausgewählte Beispiele

3.3.1 Datenmodellierung

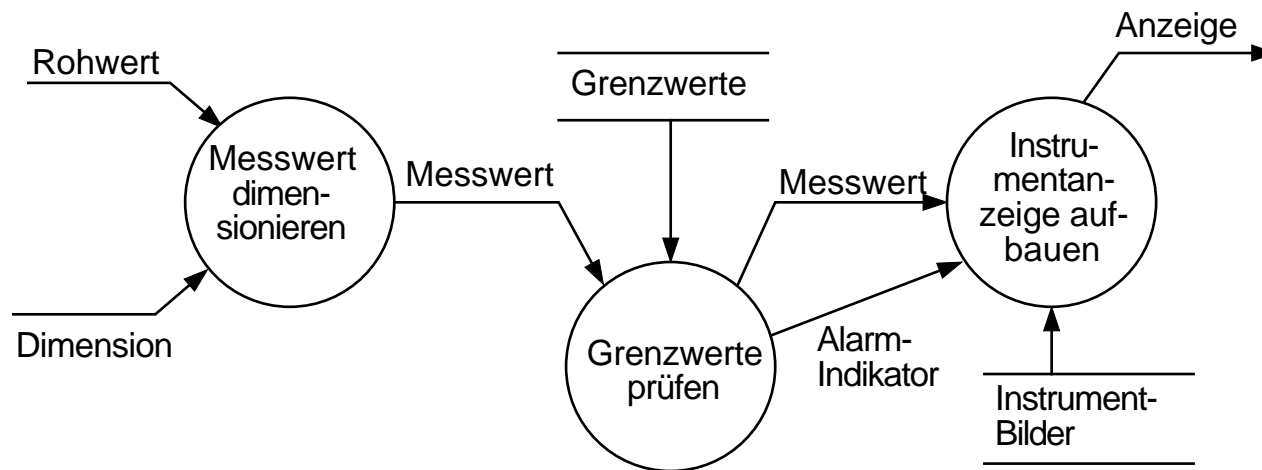
- ❑ Grundlage: **Entity-Relationship Ansatz**
- ❑ modelliert einen Ausschnitt der Realität mit Hilfe von **Gegenstandstypen** (entity types), **Beziehungstypen** (relationship types) und **Attributen** (attributes)



- + Einfach und klar
- + Leicht auf Datenbank-Realisierungen abbildbar
- Ignoriert Funktionalität und Verhalten der Systeme
- Keine Mittel zur Systemdekomposition
- Keine Lokalität oder Einkapselung von Daten

3.3.2 Strukturierte Analyse (Datenflussmodell)

- ❑ Modelliert die Funktionalität eines Systems mit Hilfe von **Datenflussdiagrammen**
- ❑ Modell ist ein Netzwerk von **Datenflüssen**, **Aktivitäten**, **Speichern** und **Endknoten**



- + Sehr anschaulich
- + Unterstützt Systemdekomposition
- Strukturbruch zwischen Spezifikation und Implementierung
- Keine Lokalität von Daten, Einkapselung nur begrenzt möglich

3.3.3 Objektorientierte Spezifikation

- ☆ Modelliert die statische Struktur eines Systems mit Hilfe von **Objekt-** oder **Klassendiagrammen**
- ☆ Näheres im [Kapitel 10](#)

3.3.4 Szenarien und Anwendungsfälle

- ☆ Modellieren die **Interaktion** zwischen systemexternen **Akteuren** und dem **System**. Jede Interaktionssequenz wird durch ein Szenario (einen Anwendungsfall) beschrieben.
- ☆ Näheres im [Kapitel 11](#)

3.3.5 UML

- ☆ Lose **Sammlung** von **Modellierungssprachen** mit einem **Klassenmodell** im Zentrum
- ☆ Näheres im **Kapitel 4**