

Angewandte Informatik I

Wintersemester 2006/2007

Prof. Dr. A. Oberweis

Prof. Dr. R. Studer

Dr. P. Hitzler

Dipl. Wirtsch. Inf. V. Pankratius

Sebastian Rudolph

Vorlesung

- **Vorlesungstermin:**
 - Dienstag 15:45 - 17:15, Tulla HS
 - ggf. Montag 15:45 - 17:15, Tulla HS

- **Aktuelle Informationen, Literaturhinweise und Unterlagen:**
 - <http://www.aifb.uni-karlsruhe.de/Lehre/Winter2006-07/AI1/>

Übungsbetrieb

- **Übungen:**
 - Mittwoch 09:45-11:15, Raum -116, Geb. 11.40
 - Mittwoch 11:30-13:00, Raum -116, Geb. 11.40
 - Donnerstag 09:45-11:15, Raum -116, Geb. 11.40
 - Freitag 08:00-09:30, Raum -116, Geb. 11.40
Beginn: nächste Woche
- **Übungsleiter:**
 - Sebastian Rudolph
(sru@aifb.uni-karlsruhe.de)
 - Victor Pankratius
(pankratius@aifb.uni-karlsruhe.de)
- **Details** unter: <http://www.aifb.uni-karlsruhe.de/Lehre/Winter2006-07/AI1/>

Inhaltsverzeichnis

Modellierung	Studer
UML-Vertiefung	Studer
Entity-Relationship Modell	Studer
Beschreibungslogik	Hitzler/Studer
Relationales Modell	Oberweis
DB-Entwurf	Oberweis
Petri-Netze	Oberweis
Ereignisgesteuerte Prozessketten	Oberweis

Literatur - Einführung und UML

- Guus Schreiber, Robert de Hoog, Hans Akkermans, Anjo Anjewierden, Nigel Shadbolt, Walter Van de Velde: **Knowledge Engineering and Management**, The MIT Press, 2000.
- Helmut Balzert: **Lehrbuch Grundlagen der Informatik**, Spektrum Akademischer Verlag, 1999
- Rainer Burkhardt. **UML. Unified Modeling Language**. Objektorientierte Modellierung für die Praxis. Gebundene Ausgabe (1999) Addison-Wesley, München; ISBN: 3827314070
- Martin Fowler, Kendall Scott: **UML konzentriert - Eine strukturierte Einführung in die Standard-Objektmodellierungssprache**, Addison-Wesley, 2000.
- Bernd Oestereich: **Objektorientierte Softwareentwicklung Analyse und Design mit der UML**. Verlag R. Oldenbourg, München 2001
- I. Sommerville, **Software Engineering**, 7th Edition, Pearson Addison Wesley, 2004, ISBN 0321210263
- Wolfgang Zuser, Thomas Grechenig, Monika Köhle, **Software Engineering**, Pearson Studium, 2004, ISBN 3827370906

Literatur – ER und DB-Theorie

- G. Schlageter, W. Stucky: **Datenbanksysteme: Konzepte und Modelle**. B. G. Teubner, Stuttgart 1983 (2. Auflage)
- R. Elmasri, S. B. Navathe: **Fundamentals of Database Systems**. Pearson Education, 4. Aufl., 2004, ISBN 0321204484;
(*Deutsche Ausgabe: R. Elmasri, S. B. Navathe: Grundlagen von Datenbanksystemen. Pearson Studium, 3., überarb. Aufl., 2002, ISBN 3827370213*)
- H. Garcia-Molina, J.D. Ullman, J. Widom, **Database Systems. The Complete Book**, Prentice Hall, 2002, ISBN 0130319953
- Hector Garcia-Molina, Jeff Ullman and Jennifer Widom: **Database System Implementation**, Prentice-Hall, 2000
- Gottfried Vossen. **Datenbankmodelle, Datenbanksprachen und Datenbankmanagementsysteme**. Oldenbourg, München, 2000, ISBN: 3486253395.
- Stefan M. Lang ; Peter C. Lockemann: **Datenbankeinsatz**, Springer Verlag, 1995, ISBN 9783540585589

Literatur - Beschreibungslogik

- Günther Görz, C.-R. Rollinger and J. Schneeberger (Hrsg): **Handbuch der Künstlichen Intelligenz**. Oldenbourg, München, 3. Auflage, 2000.
- U. Schöning, **Logik für Informatiker**, Reihe Informatik, Band 56, BI Wissenschaftsverlag, Mannheim 1989.
- John Sowa, David Deitz: **Knowledge Representation: Logical, Philosophical, and Computational Foundations**, Brooks/Cole Pub Co, 1999
- Staab, Studer, **Handbook on Ontologies**, Springer 2003.
- Dieter Fensel, **Ontologies: A Silver Bullet for Knowledge Management and Electronic Commerce**, Springer, 2001
- Franz Baader, Diego Calvanese, Deborah McGuinness, Daniele Nardi, Peter Patel-Schneider **The Description Logic Handbook – Theory, Implementation and Applications**, Cambridge 2003

Literatur - Petrinetze

- James L. Peterson: **Petri Nets**. ACM Computing Surveys (CSUR), Volume 9, Issue 3, September 1977
- T. Murata: **Petri nets: Properties, analysis and applications**. Proceedings of the IEEE, Volume 77, Issue 4, April 1989, pp. 541 – 580
- Wolfgang Reisig: **A Primer in Petri Net Design**. Springer Verlag, 1992
- J. Desel, J. Esparza: **Free Choice Petri Nets**. Cambridge University Press, 1995
- Bernd Baumgarten: **Petri-Netze. Grundlagen und Anwendungen**. Spektrum Akademischer Verlag, 1996
- W. van der Aalst, J. Desel, A. Oberweis (Eds.): **Business Process Management: Models, Techniques, and Empirical Studies**. Lecture Notes in Computer Science 1806, Springer Verlag Heidelberg, 2000

Was ist Angewandte Informatik?

- Angewandte Informatik im Sinne des AIFB
 - Vermittlung von **Grundlagen** und **Methoden** der Informatik im Kontext ihrer **Anwendungsmöglichkeiten** in der Praxis
 - Softwaresysteme und Informationsprodukte durch den systematischen Einsatz geeigneter **Analyse- und Modellierungsmethoden** so auslegen, dass sie durchgängig zu **verstehen**, dem Einsatzzweck entsprechend **optimal zu nutzen** und **dauerhaft beherrschbar** sind
 - grundlegende Qualifikation für das „**Life-Long-Learning**“

Beispiele für konkrete Aufgabenstellungen der angew. Informatik

- systematische Einführung von Wissensmanagementlösungen, z.B.
 - Skill-Management
- intelligente Web-Anwendungen, z.B.
 - Community-Portale
 - Informations-Broker
- Analyse und Umstrukturierung von Geschäftsprozessen
- Umsetzung von Geschäftsprozessen in Workflowsysteme
- Analyse und Modellierung von e-Business Anwendungen
- strategische Informatikplanung

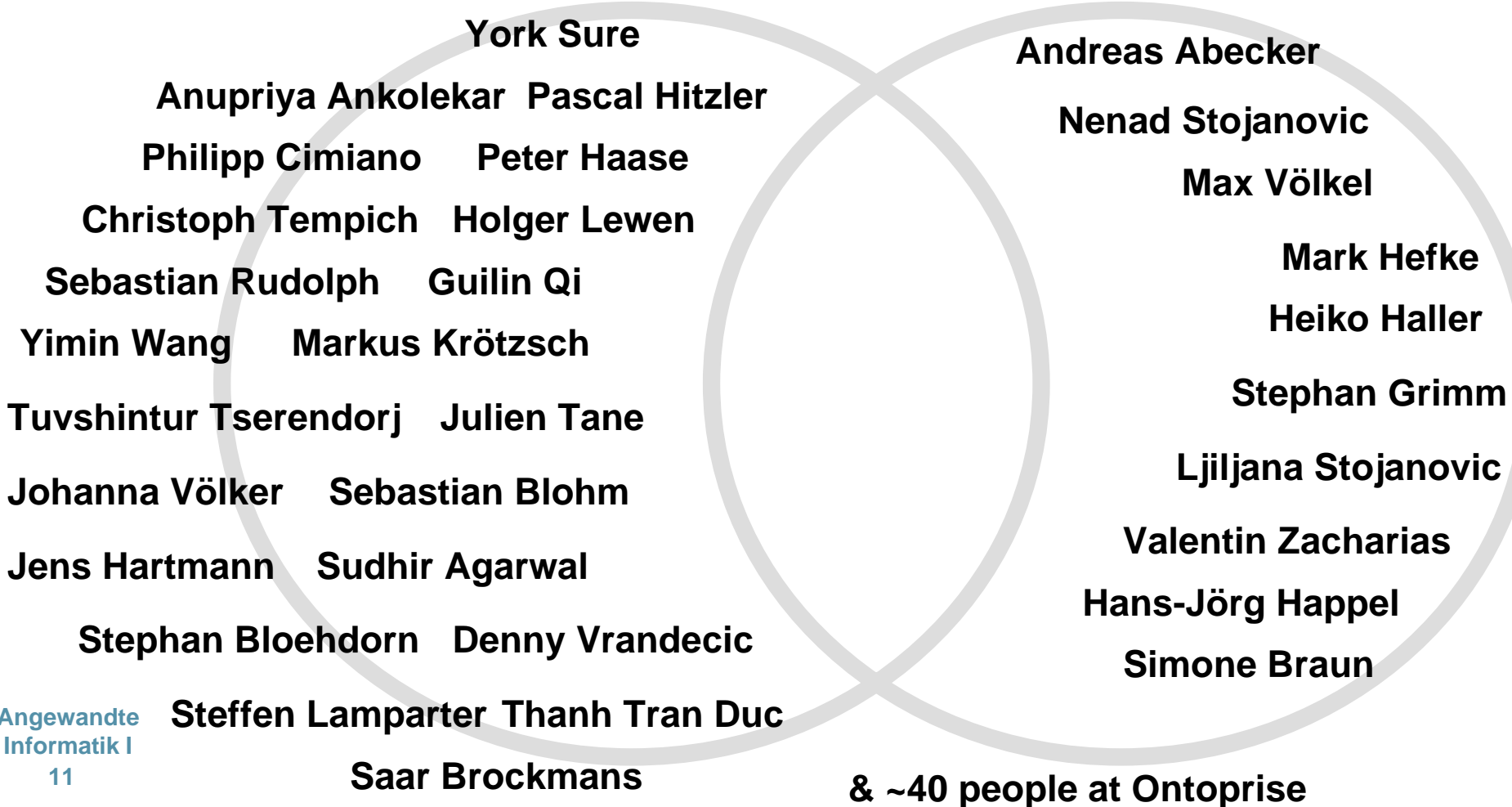
Wer sind wir?

Die Wissensmanagement-Gruppe

AIFB

Rudi Studer

FZI



Wer sind wir?

Die Wissensmanagement-Gruppe

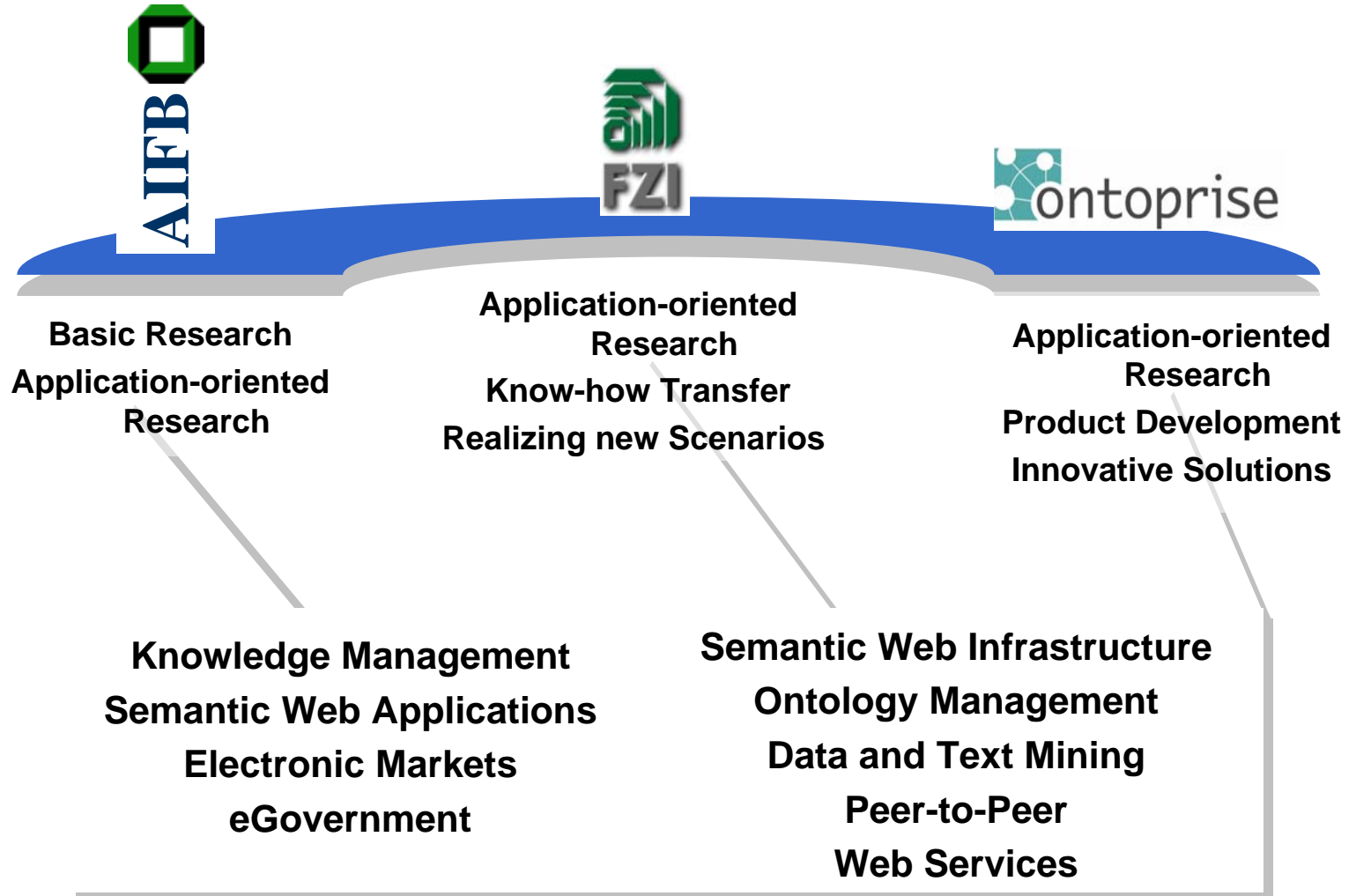
AIFB

Rudi Studer

FZI



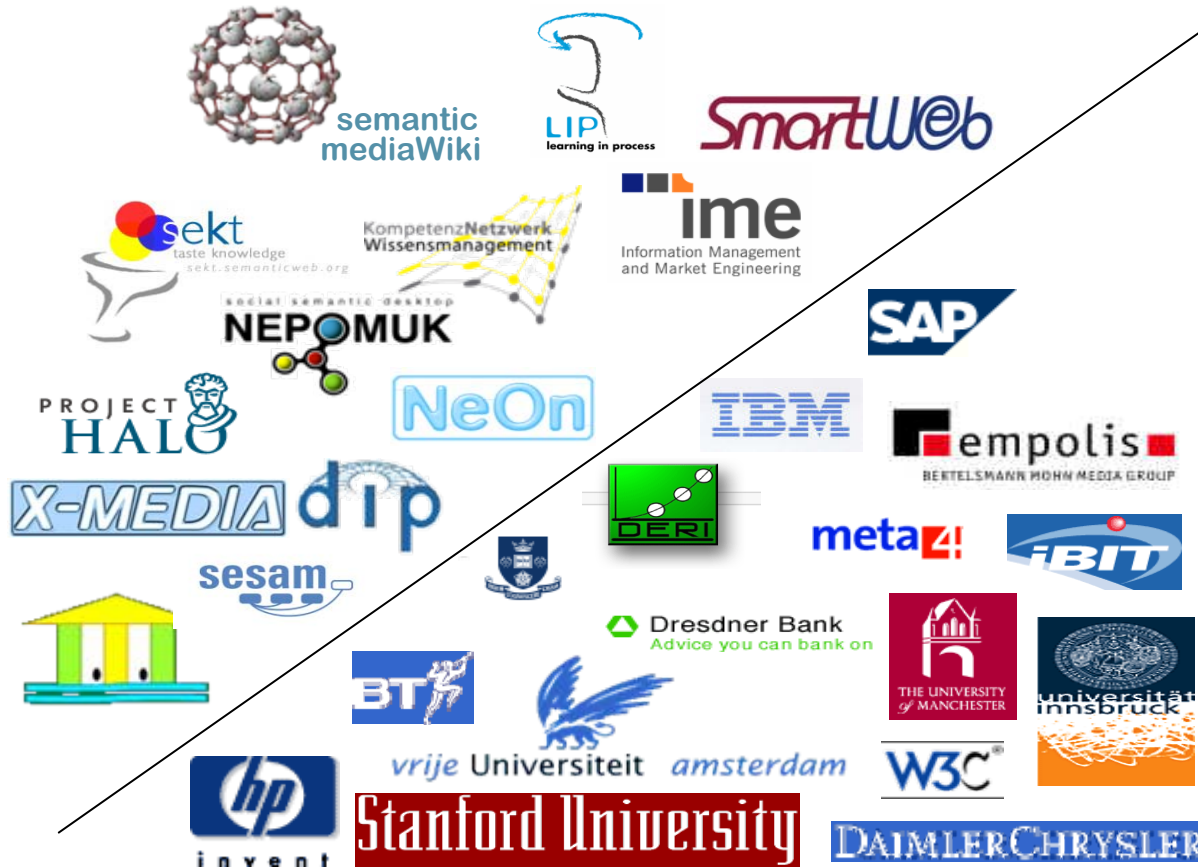
Karlsruhe: Location for Semantic Technologies and Applications



Wir bieten interessante, praxisrelevante Themen

Vorlesungen - Seminare - Praktika - Diplomarbeiten

Projekte und Partner:



Baden-Württemberg



Information Society Technologies



Interesse an Lehrveranstaltungen?

- Lehrveranstaltungen im HD für Wirtschaftsingenieure
 - Wissensmanagement
 - Knowledge Discovery
 - Intelligente Systeme im WWW
 - Dynamische und interoperable Systeme im Wissensmanagement*

- Lehrveranstaltungen für Informationswirte
 - Pflichtveranstaltung:
 - Informations- und Wissensmanagement (mit IPD)
 - Wahlfach Informations- und Wissenssysteme:
 - Knowledge Discovery
 - Wahlfach Informationsdienstleistungen in Netzen
 - Intelligente Systeme im WWW

Interesse an Mitarbeit?

- Mitarbeit an praxisrelevanten internationalen Forschungsprojekten im Rahmen von
 - HiWi-Tätigkeiten
 - Seminaren / Praktika
 - Studien- / Diplomarbeiten

Themen

Wissensmanagement	Ontology Management
Semantic Web	Data/Text Mining
Intelligente WWW-Applikationen	Peer-to-Peer
Business Intelligence	Web Services

<http://www.aifb.uni-karlsruhe.de/WBS>

Unsere Forschungsschwerpunkte

“People can` t share knowledge
if they do not speak a common language.“

“Modellieren statt Programmieren!“

Betriebliche Informations- und Kommunikationssysteme – Prof. Stucky / Prof. Oberweis

- Lehrveranstaltungen
 - Datenbanksysteme
 - Workflow Management
 - Verteilte Datenbanksysteme
 - Programmierung Kommerzieller Systeme – Einsatz betrieblicher Standardsoftware

1 Modellierung - Inhaltsverzeichnis

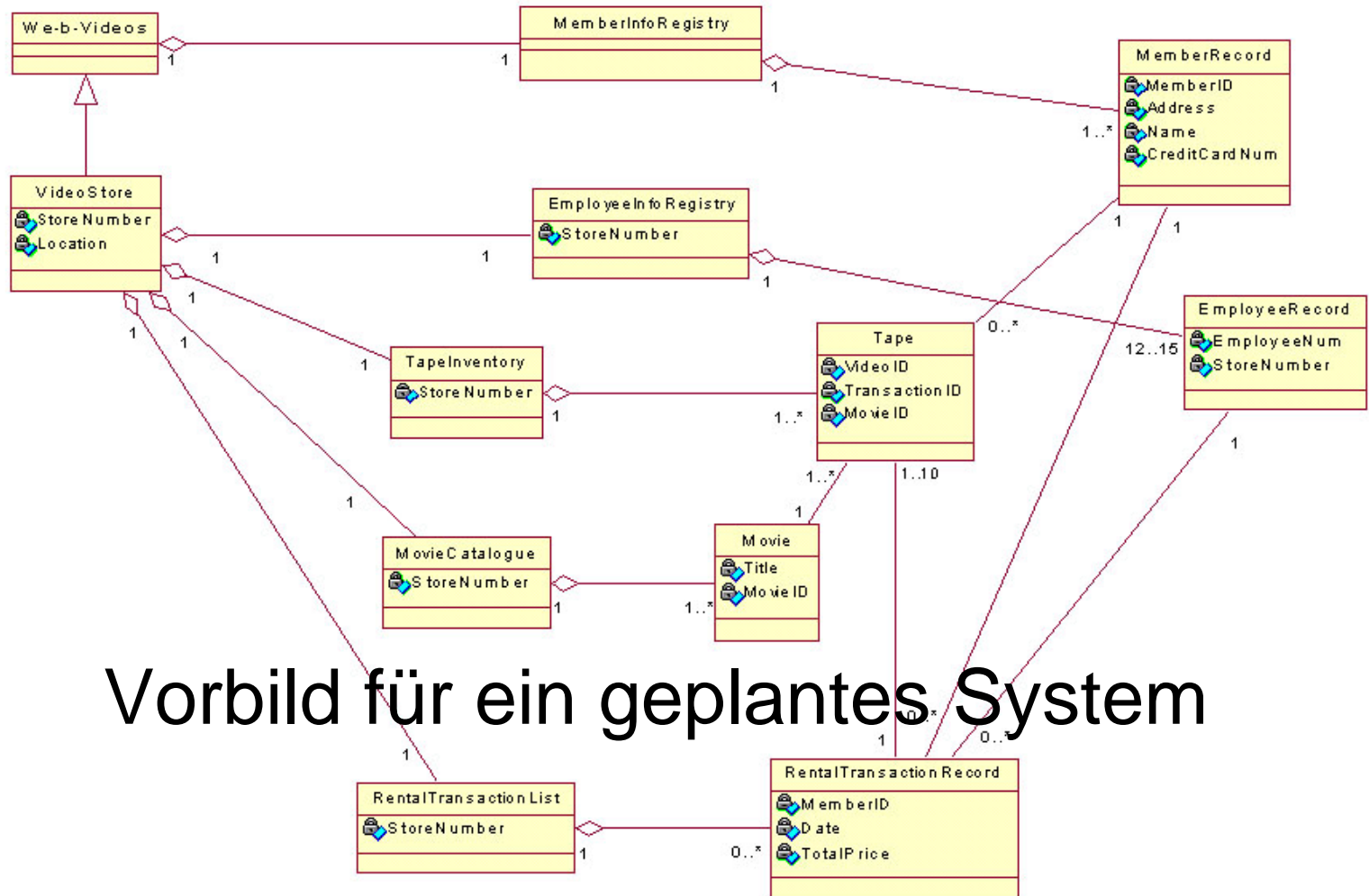
- 1.1 Was ist Modellierung?
- 1.2 Modelle in der Informatik
- 1.3 Wozu Modellierung?
- 1.4 Wer modelliert für wen?
- 1.5 Was muss modelliert werden?
- 1.6 Wie kann modelliert werden?
 - 1.6.1 Anforderungen an Modellierungssprachen
- 1.7 Ein begleitendes Beispiel

1.1 Modelleisenbahn

Abbild eines realen Systems



1.1 Modell eines Softwaresystems



Vorbild für ein geplantes System

1.1 Modell / Modellierung

- Ein Modell ist ein konkretes oder gedankliches
 - **Abbild** eines vorhandenen Systems
 - **Vorbild** für ein zu schaffendes System
- Modelle sind durch die **Wahrnehmung** der modellierenden Person geprägt und sind auf einen bestimmten **Verwendungszweck** ausgerichtet
- Modellierung ist der **Prozess** der Erstellung eines Modells

1.2 Modelle in der Informatik

Analogie zu Eisenbahn oder Bauwerken

- **Abbilder** von bestehenden Informatiksystemen
- **Vorbilder** für zu konstruierende Informatiksysteme

Aber

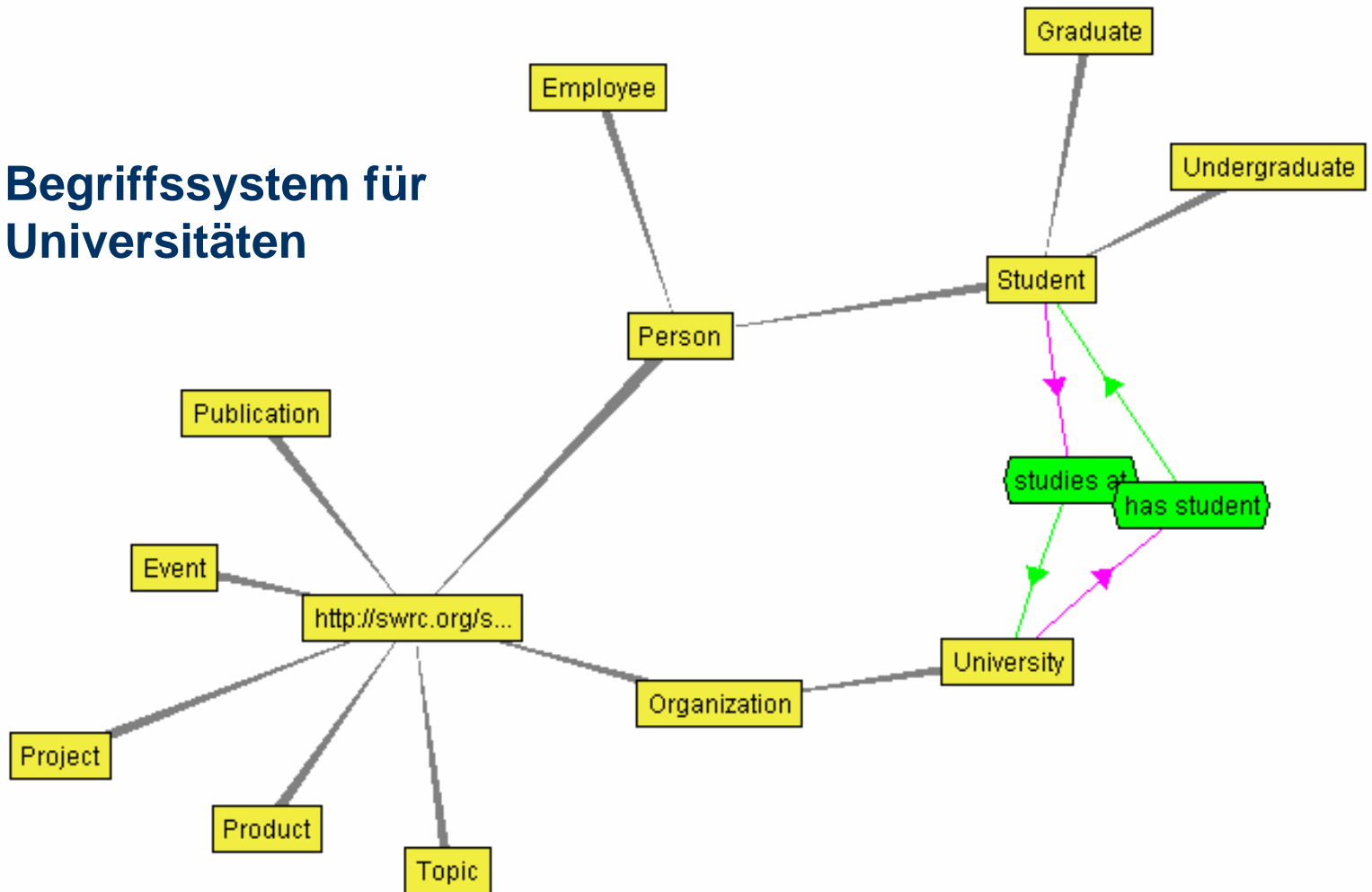
Unser Fokus in dieser Vorlesung



- Informatikmodelle sind wesentlich abstrakter
 - Software/Modelle kann man nicht anfassen

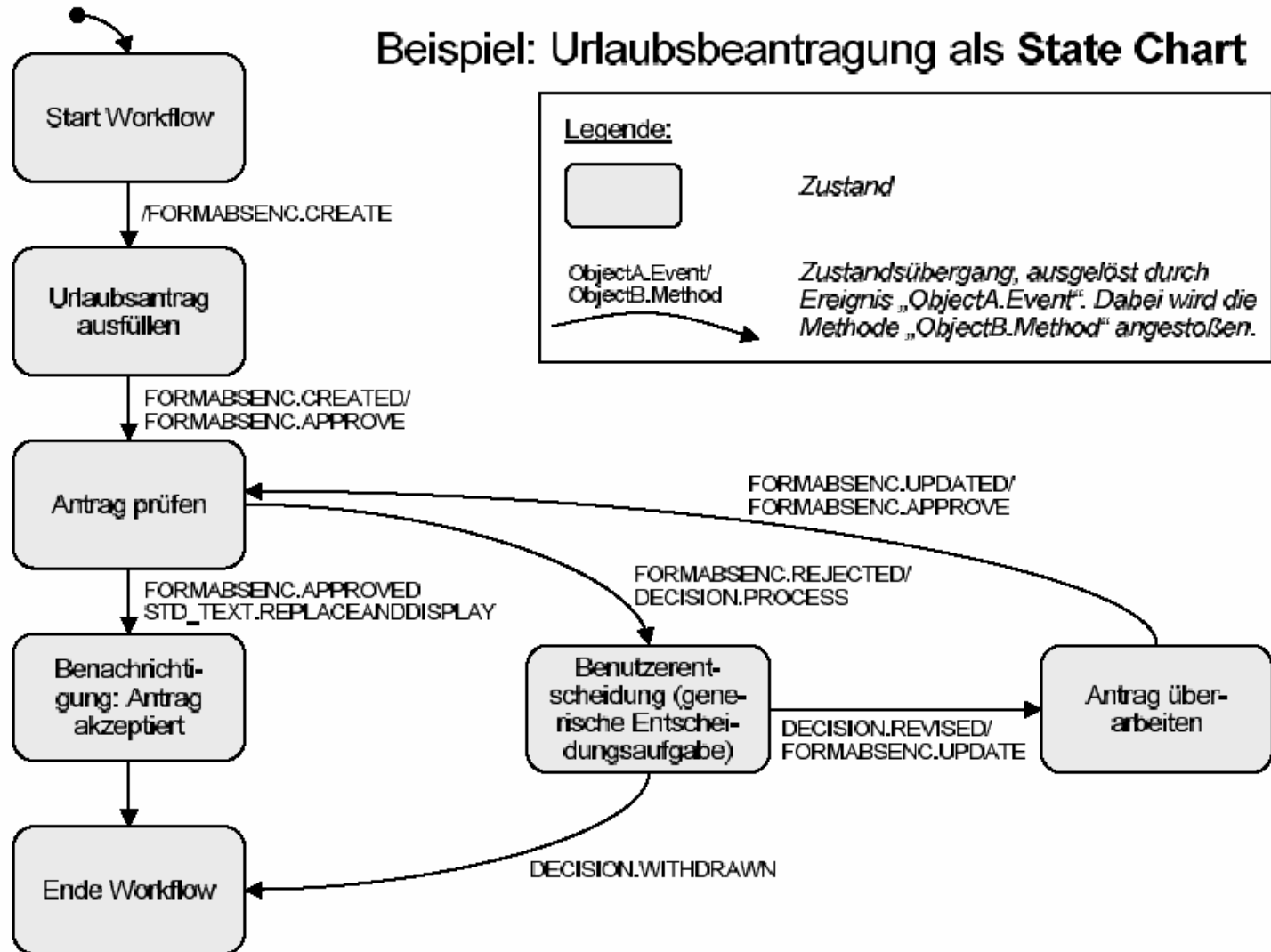
1.2 Modelle in der Informatik

Begriffssystem für Universitäten



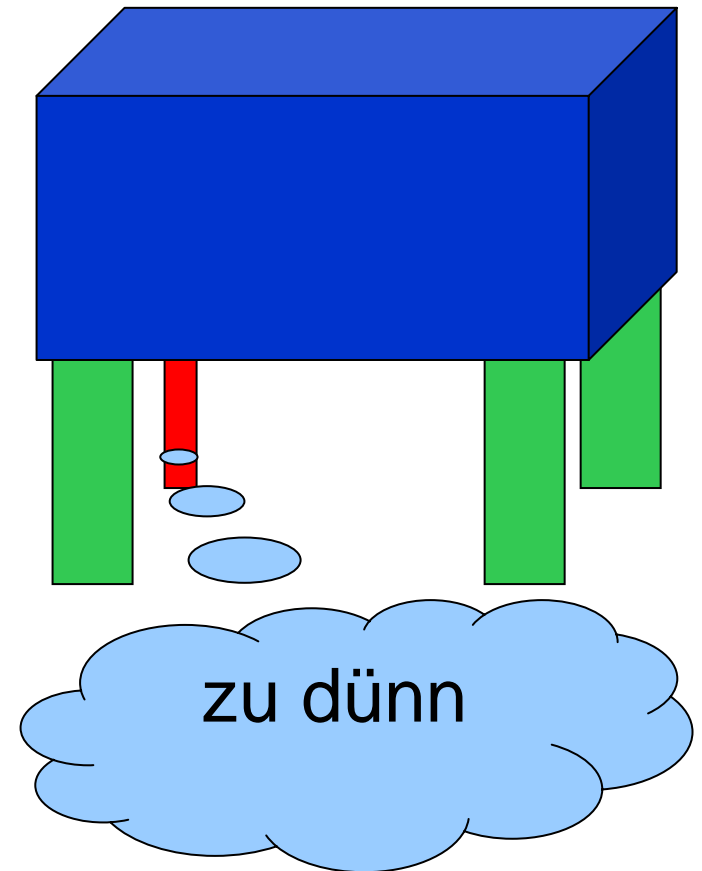
1.2 Modelle in der Informatik

Beispiel: Urlaubsbeantragung als State Chart



1.3 Wozu Modellierung?

zum Verstehen eines Systems oder Problems



1.3 Wozu Modellierung?

zur Kommunikation über das (geplante) System



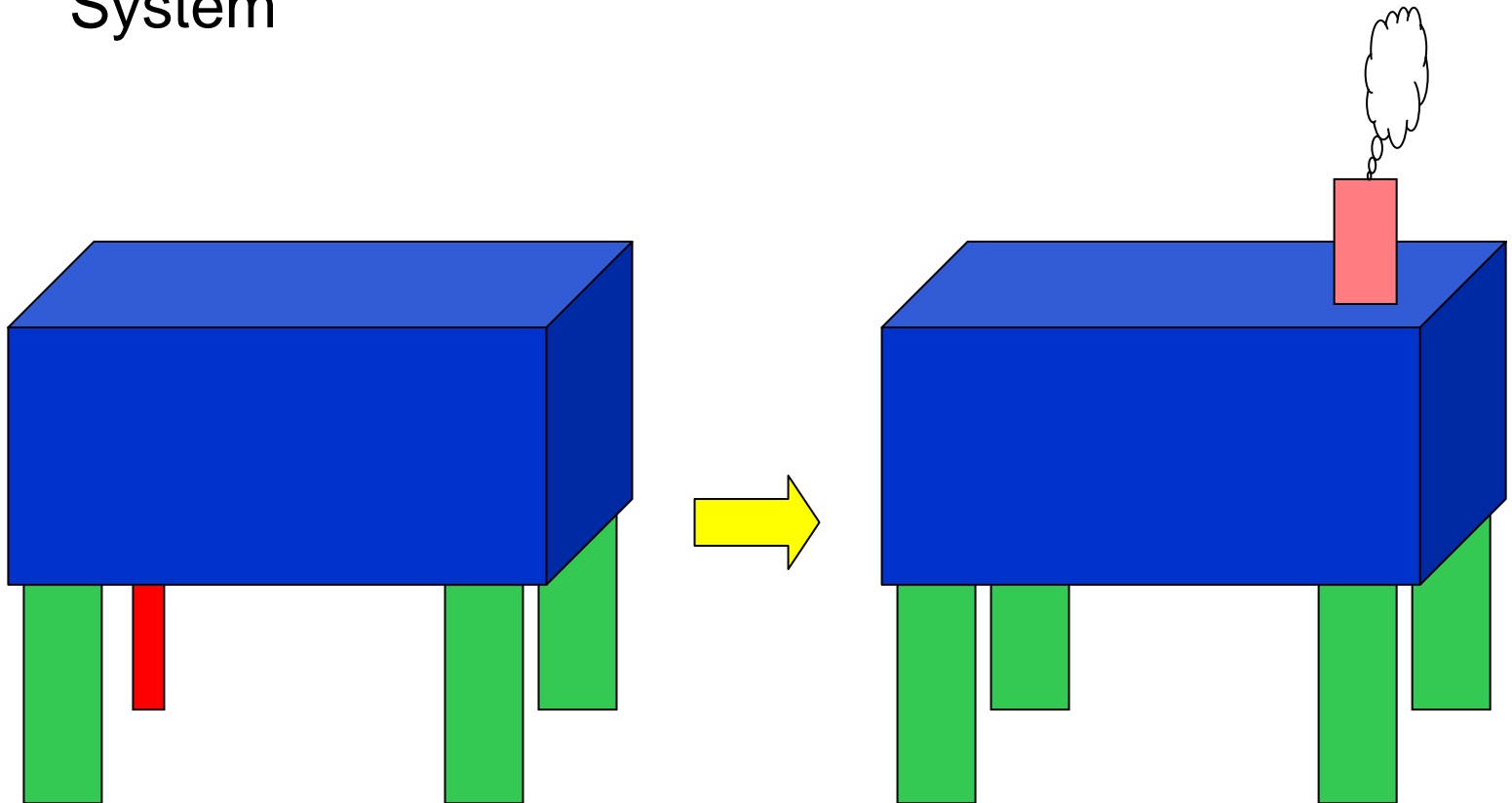
1.3 Wozu Modellierung?

zum Bewerten und Kritisieren eines (geplanten)
Systems



1.3 Wozu Modellierung?

Zur Spezifikation von Anforderungen an ein geplantes System



1.2 Wozu Modellierung?

zur Durchführung von Experimenten, die am Original nicht durchgeführt werden können, sollen oder dürfen



1.3 Wozu Modellierung?

- Trennen von Anwendungswissen und Programmcode bei Softwaresystemen
 - deklaratives Wissen
 - ist nicht von Änderungen in der Software betroffen
 - kann unabhängig von der Software gewartet werden

1.4 Was muss modelliert werden?

- Zu modellierende Aspekte hängen stets von der Aufgabenstellung und Zielsetzung ab.

- **Statisch:**
 - Konzepte und Instanzen
 - Beziehungen
 - Ressourcen

- **Dynamisch:**
 - Aktivitäten und Ablaufstrukturen:
 - Sequenz, Alternative, Wiederholung, Parallelität, Unabhängigkeit
 - Zeitaspekte
 - Ausnahmesituationen

1.4 Was muss modelliert werden?

- **Organisatorisch:**
 - Rollen und Organisationsstrukturen:
 - Zuständigkeiten, Verantwortlichkeiten, Kompetenzen
 - Kommunikationsstrukturen
 - Geschäftsregeln:
 - allgemeingültige Regeln
 - Kostenaspekte
- **Qualitäts- und Sicherheitsaspekte**
- **Was nicht?**
 - Implementierungsabhängiges wie
 - (programmier-) technische Realisierung
 - plattformabhängige Zusammenhänge

1.4 Was muss modelliert werden?

- Modellierung ist stets abhängig von:
 - einer **Domäne/Anwendungsbereich** (z.B. PKW)
 - einer bestimmten **Aufgabe** (PKW-Herstellung vs. PKW-Verkauf)
- Daraus ergeben sich die Fragen:
 - **Was ist relevant** für die Modellbildung?
 - **Welche Zusammenhänge** gibt es zwischen den relevanten Gegenständen?
- **Wie fein** muss das resultierende Modell sein?
- **Merke: Es gibt nicht DAS richtige Modell!**
 - Abhängig zum einen von der Sichtweise, zum anderen vom Zweck sind verschiedene Abstraktionsgrade der Modellierung möglich

1.5 Wer modelliert für wen?

- **Drei Rollen**
 - **Wissensträger:** Personen, die über das Wissen des zu modellierenden Systems verfügen, auch Domänenexperten genannt.
 - **Modellierer:** Personen, welche ein Modell erstellen, also modellieren.
 - **Endanwender:** Personen, die an dem Modell interessiert sind, auch Kunden oder Auftraggeber genannt.
- Bei der Modellierung können mehrere Personen ein und dieselbe Rolle haben und eine Person kann mehrere Rollen haben.
- Bei der Modellierung von Softwaresystemen sind dies üblicherweise Systemanalytiker, externe Berater, Endanwender, Programmierer, Management.

1.6 Wie kann modelliert werden?

datenorientiert	Statische Struktur des Anwendungsbereichs, Struktur und Zusammenhänge der Daten eines Systems
funktionsorientiert	Funktionen und Datentransformationen, Funktionalitäten eines Systems
prozessorientiert	Dynamisches Systemverhalten
objektorientiert	Struktur und Verhalten eines Systems in seinem Aufgabenumfeld
interaktionsorientiert	Interaktion zwischen Mensch und System

1.6.1 Anforderungen an Modellierungssprachen

- **Ausdrucksmächtigkeit**
 - Alle relevanten Aspekte müssen modellierbar sein.
 - Adäquatheit/ Angemessenheit der Modellierungskonstrukte
 - Erweiterbarkeit
 - Ausführbarkeit / Simulierbarkeit
- **Einfachheit, Verständlichkeit**
- **Implementierbarkeit**
- **Formalisierungs- bzw. Präzisierungsgrad**
 - flexible Anpassbarkeit an das Ziel der Modellierung, die Zielgruppe des Modells

1.6.1 Anforderungen an Modellierungssprachen

- **Visualisierungsmöglichkeiten**
 - graphische Darstellung
 - unterschiedliche Sichten, Modularisierbarkeit, Detaillierungsgrad
- **Entwicklungsunterstützung**
 - methodische Unterstützung für die Modellierung
 - benötigt Richtlinien
 - Werkzeugunterstützung

1.6.1 Anforderungen an Modellierungssprachen

- **Analysier- und Validierbarkeit**
 - Prüfung syntaktischer Eigenschaften (isolierte Elemente, Zyklen, ...)
 - Analyse semantischer Aspekte (Erfüllbarkeit, ...)
 - Analyse anwendungsbezogener Aspekte (Durchlaufzeiten, Reaktionszeit, ...)
 - inhaltliche Richtigkeit (entspricht das Modell der Realität)

1.7 Ein begleitendes Modellierungsbeispiel

Ein **Unternehmen** möchte alle relevanten Daten mit Hilfe eines Informationssystems verwalten. Deshalb wurden die Struktur des Unternehmens und die Anforderungen an das Informationssystem in einem Text skizziert. Dieser Text dient als Grundlage für die Modellierung mittels verschiedener Modellierungsformalismen.

- Das Unternehmen ist hierarchisch gegliedert. Es besteht aus verschiedenen Unternehmensbereichen (Elektro, KFZ, ...), die auf Betriebe verteilt sind, die sich an verschiedenen Standorten befinden. An einem Standort befindet sich nur jeweils ein Betrieb. Zu jedem Betrieb soll der Betriebsleiter und das Personalbudget, sowie die zum Betrieb gehörenden Gebäude festgehalten werden.

1.7 Forts. Ein begleitendes Modellierungsbeispiel

- Zu jedem Standort sollten sowohl die Postanschrift, als auch die geografischen Koordinaten gespeichert werden. Letztere dienen der Tourenplanung mittels eines GPS (Global Positioning System).
- Die Abteilungen des Unternehmens sind jeweils geschlossen in einem Gebäude untergebracht, um kurze Wege zu gewährleisten.
- Zu den zur Verwaltung der Mitarbeiter notwendigen Informationen gehören neben Personal-Nummer, Name und Gehalt auch die Zugehörigkeit zu Unternehmensbereich, Betrieb und Abteilung, sowie das Gebäude, in dem sich der Arbeitsplatz befindet.
- Die Aufgaben der Firma sind stark projektbezogen. Deshalb sind für eine Projektverwaltung alle Projekte, Projektleiter und die zugehörigen Mitarbeiter zu speichern. Da manche Mitarbeiter parallel an verschiedenen Projekten arbeiten, ist auch der prozentuale Anteil der Arbeitszeit, mit dem an einem Projekt gearbeitet wird, von Bedeutung.