



Fakultät für  
**Mathematik und  
Informatik**

# Modellieren lehren Lehren modellieren

**Jörg Desel**

Fakultät für **Mathematik und Informatik**

Lehrgebiet Softwaretechnik und Theorie der Programmierung  
Prof. Dr. Jörg Desel

## Projekt FIBIDAS im Forschungsschwerpunkt D<sup>2</sup>L<sup>2</sup>

Vom Fachinformatiker zum  
Bachelor Informatik durch  
adaptierte Studiengangsgestaltung



# Modellierung in der Ausbildung zum Fachinformatiker (Anwendungsentwicklung)

## Modellierung im Universitätsstudium (FH etwas anders)

## Modellieren lehren

## Lehre modellieren

## Was lernen Fachinformatiker über Modellierung?

# Bundesgesetzblatt<sup>1741</sup>

Teil I

G 5702

1997

Ausgegeben zu Bonn am 15. Juli 1997

Nr. 48

Tag	Inhalt	Seite
10. 7. 97	Verordnung über die Berufsausbildung im Bereich der Informations- und Telekommunikationstechnik FNA: neu: 806-21-1-244	1741
11. 7. 97	Zweite Verordnung zur Änderung der Branntweinsteuerverordnung ..... FNA: 612-7-10	1800
2. 7. 97	Berichtigung der Bekanntmachung der Neufassung des Grunderwerbsteuergesetzes ..... FNA: 610-6-10	1804

## Was lernen Fachinformatiker über Modellierung?

Analyse und Design  
(§ 10 Abs. 1 Nr. 6.1)

- a) Vorgehensmodelle und -methoden sowie Entwicklungsumgebungen aufgabenbezogen auswählen und anwenden
- b) strukturierte und objektorientierte Analyse- und Designverfahren anwenden
- c) Programmspezifikationen festlegen, Strukturen aus fachlichen Anforderungen ableiten, Schnittstellen festlegen
- d) Methoden zur Strukturierung von Daten und Programmen anwenden
- e) Daten und Funktionen zu Objekten zusammenfassen, Klassen definieren und Hierarchiediagramme erstellen

## Was lernen Fachinformatiker über Modellierung?

Architekturen  
(§ 10 Abs. 2 Nr. 8.1)

- a) Rechnerarchitekturen beurteilen und einordnen
- b) Softwarearchitekturen aufgabenbezogen entwickeln
- c) Softwarearchitekturen an Betriebssysteme anpassen
- d) Softwarearchitekturen in Netze integrieren
- e) Betriebssysteme anpassen und konfigurieren

Datenbanken und Schnittstellen  
(§ 10 Abs. 2 Nr. 8.2)

- a) Datenbankprodukte aufgabengerecht auswählen
- b) Datenbankstrukturen, insbesondere logische Struktur der Daten, Objekte, Attribute, Relationen und Zugriffsmethoden, festlegen sowie Schlüssel definieren



Was k

**Gesellschaft für  
Informatik e.V. (GI)**

liums?

**Empfehlungen für  
Bachelor- und  
Masterprogramme im  
Studienfach Informatik  
an Hochschulen**

Stand: 01.07.2016

## Modellierung

Kognitive Prozessdimension	Stufe 1 Verstehen	Stufe 2 Anwenden	Stufe 3 Analysieren
Geringe Kontextualisierung und Komplexität	Formale Beschreibungen von Aufgaben und Systemeigenschaften in grundlegenden Kalkülen, wie z.B. strukturierten Mengen, Termen und Algebren, Aussagen- und Prädikatenlogik sowie Graphen, interpretieren und erläutern.  Von wichtigen spezialisierten Modellierungskalkülen, wie z. B. endliche Automaten, kontextfreie Grammatiken und Petri-Netze sowie den Sprachen der UML, die Beschreibungsformen und die Bedeutung interpretieren und erläutern.	Aufgaben und Systemeigenschaften auf ihren konzeptionellen Kern abstrahieren und durch ein Modell in informeller oder formaler Notation präzise und vollständig beschreiben.  Verschiedene Sichten auf ein System mit passenden Modellen darstellen.	Formale Beschreibungen mittelgroßer Aufgabenstellungen analysieren und die spezifizierten Eigenschaften ermitteln.  Überprüfen, ob ein Modellkandidat vorgegebene Anforderungen erfüllt.
		<b>Stufe 2a Übertragen</b>	<b>Stufe 3a Bewerten</b>
Starke Kontextualisierung und hohe Komplexität		Eine vorgegebene Aufgabenstellung größerer Komplexität mit passenden Kalkülen auf angemessenem Abstraktionsgrad zielgerichtet formal beschreiben. Dabei professionelle Werkzeuge einsetzen und Ergebnisse bewerten.  Allgemeine, abstrakte Fragestellungen und Zielsetzungen in konkrete Spezifikationen und Analysefragen umsetzen.	Praxisrelevante Modellierungskalküle und -werkzeuge auf ihre Eignung für eine konkrete Anwendung prüfen, beurteilen und auswählen.

Informatik

?

## Inhalte

**Modellierung:** Das Modellieren ist eine für das Fach Informatik typische Arbeitsmethode, die in allen Gebieten des Faches angewandt wird. Aufgaben, Probleme oder Strukturen werden untersucht und als Ganzes oder in Teilaspekten beschrieben, bevor sie durch den Entwurf von Software, Algorithmen, Daten und/oder Hardware gelöst bzw. implementiert werden. Mit der Modellierung eines Problems zeigt man, ob und wie es verstanden wurde. Damit ist sie Voraussetzung und Maßstab für die Lösung und sie liefert meist auch den Schlüssel für einen systematischen Entwurf. Als Ausdrucksmittel für die Modellierung steht ein breites Spektrum von Kalkülen und Notationen zur Verfügung. Sie sind spezifisch für unterschiedliche Arten von Problemen und Aufgaben. Deshalb werden in den verschiedenen Gebieten der Informatik unterschiedliche Modellierungsmethoden eingesetzt. In den entwurfsorientierten Gebieten (Softwaretechnik, Hardware-Entwurf) ist die Bedeutung der Modellierung und die Vielfalt der Methoden natürlich besonders stark ausgeprägt.

## Lernergebnisse / Fachkompetenzen

Studierende kennen wesentliche Techniken zur Modellierung informatischer Probleme. Sie können für ein gegebenes Problem eine geeignete Modellierungstechnik auswählen und das Problem mit dieser Technik beschreiben. Sie können grundlegende Techniken erweitern und verfeinern, um so neuartige Probleme zu modellieren.

## Angestrebte Lernergebnisse:

Schaffung der methodischen Grundlagen zur Umsetzung realweltlicher Problemstellungen in komplexe Softwaresysteme  
Schaffung eines Grundverständnisses für die Modellierung  
Erlernen von Techniken für die Prozess- und Datenmodellierung auf fachkonzeptueller Ebene  
Erlernen von objektorientierten Modellierungstechniken auf DV-konzeptueller Ebene  
Vermittlung praktischer Erfahrungen in der modellgetriebenen Systementwicklung

## Inhalt:

Modellierungstheorie: Von der Diskurswelt zu formalisierten Informationsmodellen  
Prozesse, Workflows und Geschäftsprozesse  
Meta-Modelle  
Referenzmodellierung  
Grundsätze ordnungsmäßiger Modellierung  
Fachkonzeptuelle Modellierung mit höheren Petri-Netzen und der Entity Relationship-Methode  
Grundlagen der Model Driven Architecture  
Objektorientierte Modellierung mit UML  
Umsetzung konkreter Aufgabenstellungen mit Modellierungswerkzeugen (Income, Rational Rose) und Java



## Lernziele/Kompetenzen

Die Studierenden kennen die Techniken und Notationen der Softwareentwicklung. Sie können die Basistechniken, die die Grundlage des modernen Software Engineering bilden, anwenden.

Die Studierenden

- kennen die Grundlagen der Projektplanung
- verstehen das Zusammenwirken von Softwaremodulen und größeren, sequentiellen Softwaresystemen sowie Frameworks; sie können dies anhand einfacher Beispiele selbständig in der Softwareentwicklung anwenden
- kennen wichtige Grundbegriffe zu Entwurfsmustern und elementaren Modellierungstechniken

## Inhalt

- Standard-Prozessmodelle der Softwareentwicklung
- Anforderungsbeschreibungen (Lasten-/Pflichtenheft)
- Projektplanungstechniken (Netzplan, Gantt-Diagramm, Aufwandsberechnungen)
- Softwareentwicklungsphasen (Analyse, Entwurf und Implementierung)
- Modularisierung von Programmen und Software (z.B. funktionale Dekomposition, Objektorientierung)
- Notationen für die funktional dekomponierende Softwareentwicklung in der Analyse- und Entwurfsphase, z.B. Structured Analysis, Real Time Analysis, Structured Design
- Notationen für die objektorientierte Softwareentwicklung in der Analyse- und Entwurfsphase, z.B. **UML**
- Programmgerüste (Program frameworks) und deren Anwendung
- wichtige Entwurfsmuster (Composite, Beobachter, Fassade)
- Qualitätssicherung, Techniken zur Prüfung von Software

## Was unterscheidet die Lehre in Ausbildung und Studium?

- Im Studium wird nicht *mehr* gemacht, sondern etwas Anderes!
- Es gibt aber signifikante **Überschneidungen:**  
Notationen, ihre grobe Bedeutung, ihr Einsatz (zum Beispiel UML)
- **Nur in Ausbildung:**  
Konkrete Anwendung von Modellierungsmethoden (unternehmensabhängig)  
im Software-Entwurf  
**Ziel:** Beherrschung der in diesem Unternehmen heute üblichen  
modellbasierten Methoden der Softwareentwicklung
- **Nur in Studium:**  
Beziehungen der Notationen zueinander, formale Semantik  
**Ziel:** Möglichkeit der schnellen Einarbeitung und Gestaltung von modell-  
basierten Methoden der Softwareentwicklung heute und in der Zukunft

# Modellieren lehren

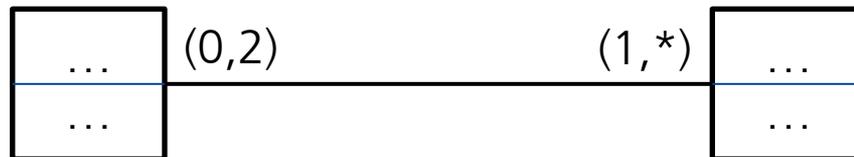
## Schwerpunkte der Lehre im Studium

- Was heißt eigentlich modellieren, was ist ein Modell?
- Was genau bedeutet ein Modell (und was nicht)?
- Wo kommen die Modellierungssprachen her, und warum?
- Wann ist ein scheinbar korrektes Modell ungeeignet?
- Mathematik als Modellierungssprache

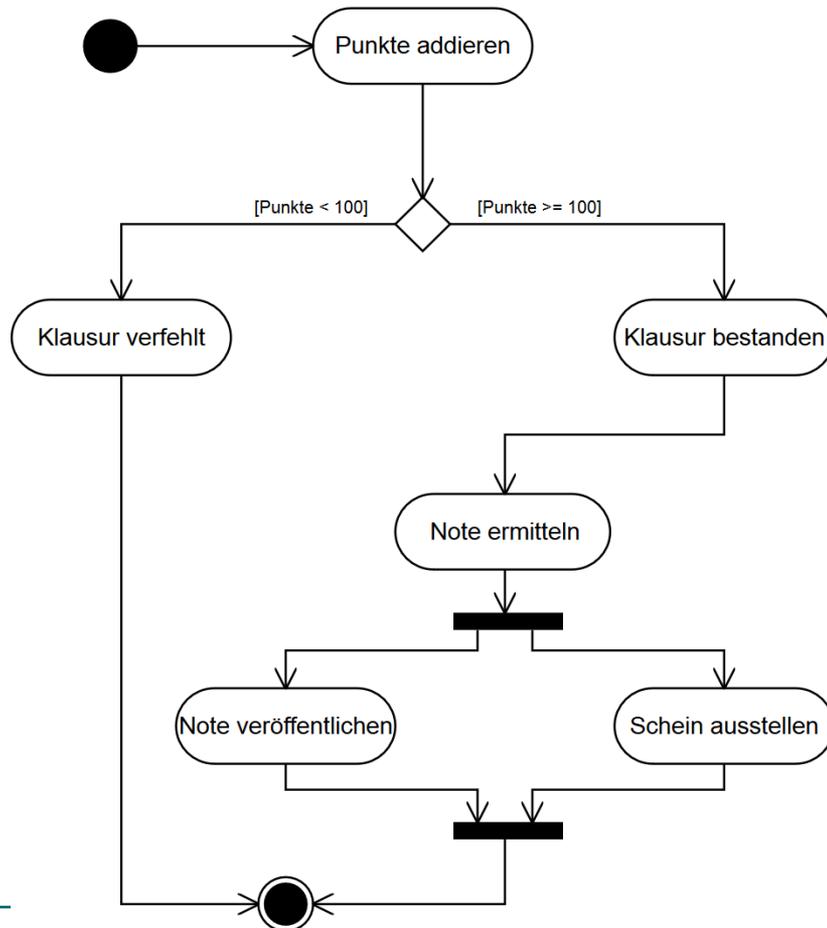
## Was heißt eigentlich modellieren, was ist ein Modell?

- Bezug zu Stachowiak: Abbildung, Reduktion, Pragmatik
- Welchen Weltausschnitt modelliert man, was sind die Schnittstellen?
- Wenn ein Modell aus mehreren Diagrammen besteht:  
Was genau ist ihr Zusammenhang?
- Wie geht man mit dem nicht modellierten (wegabstrahierten) um?
- Was hat Modellierung mit dem Rest des Informatikstudiums zu tun?

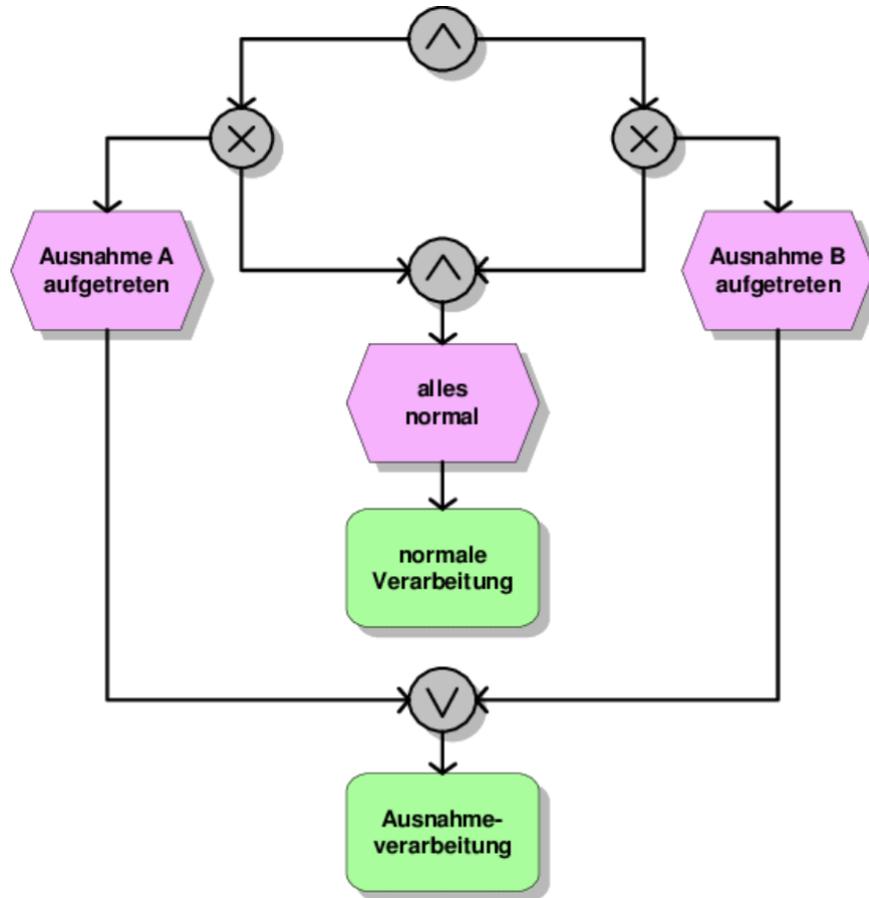
## Was genau bedeutet ein Modell?



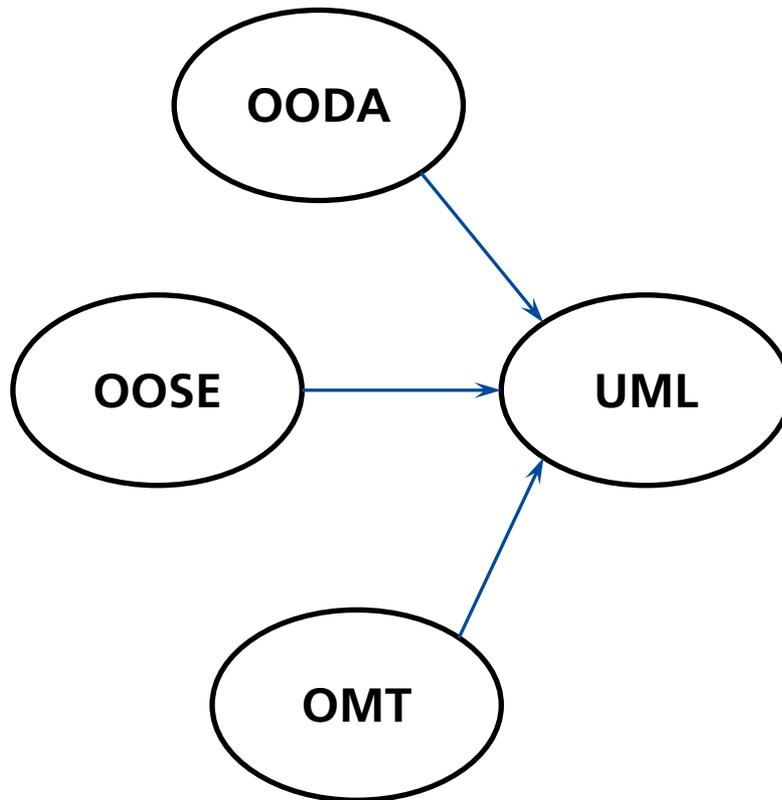
# Was genau bedeutet ein Modell?



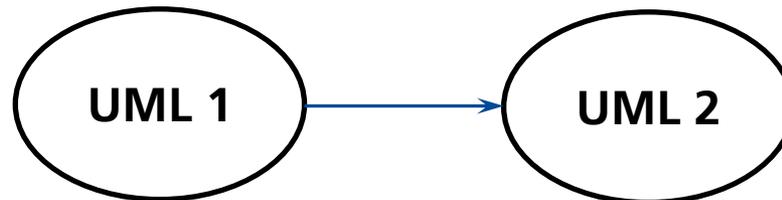
# Was genau bedeutet ein Modell?



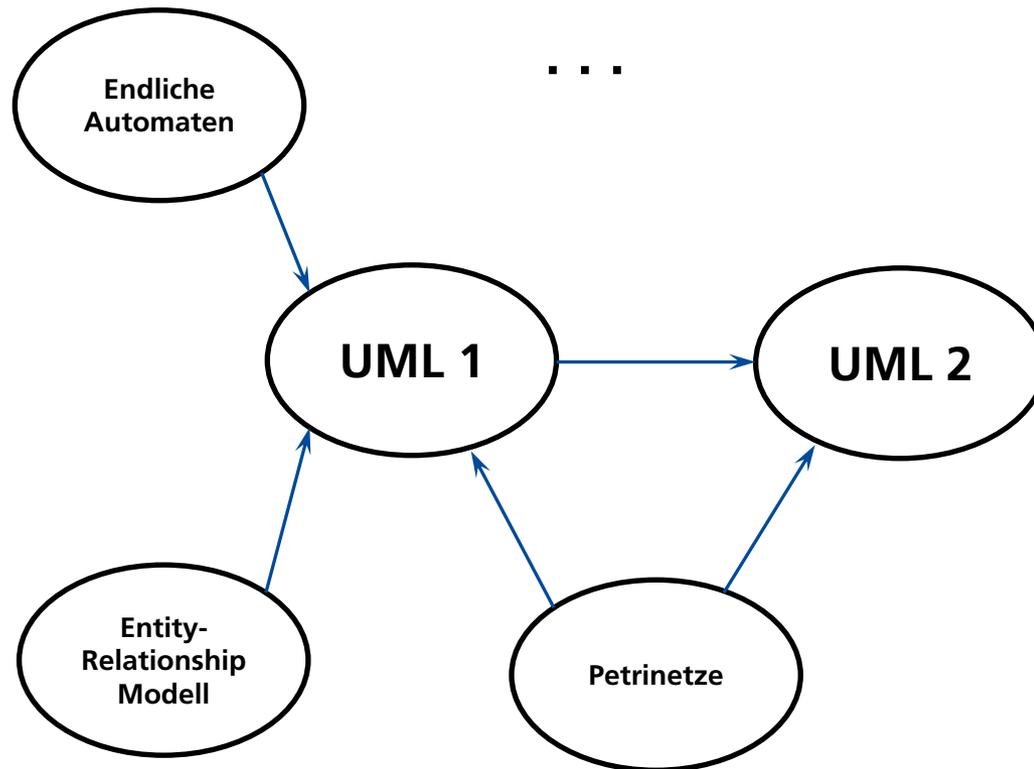
## Historische Perspektive



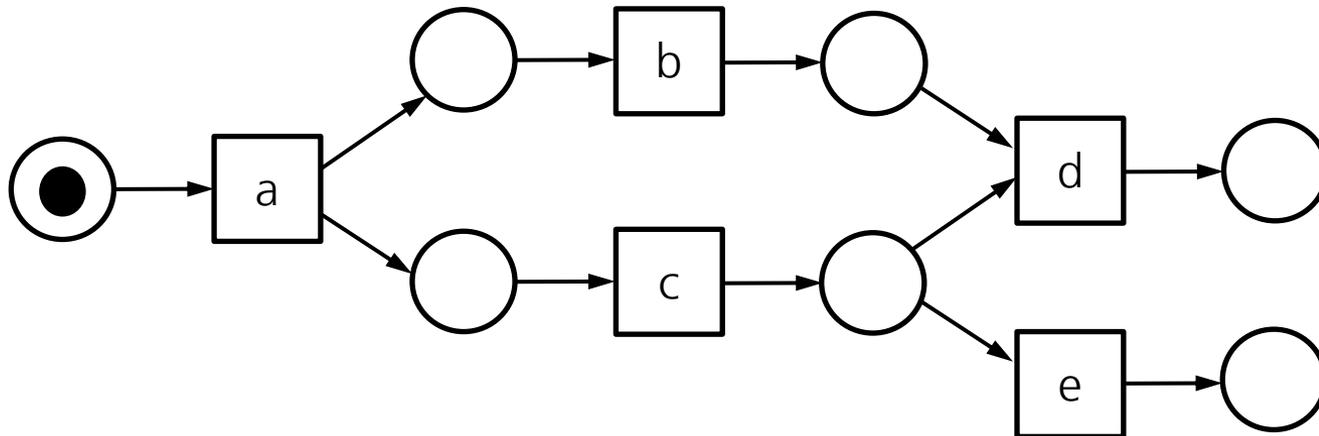
# Historische Perspektive



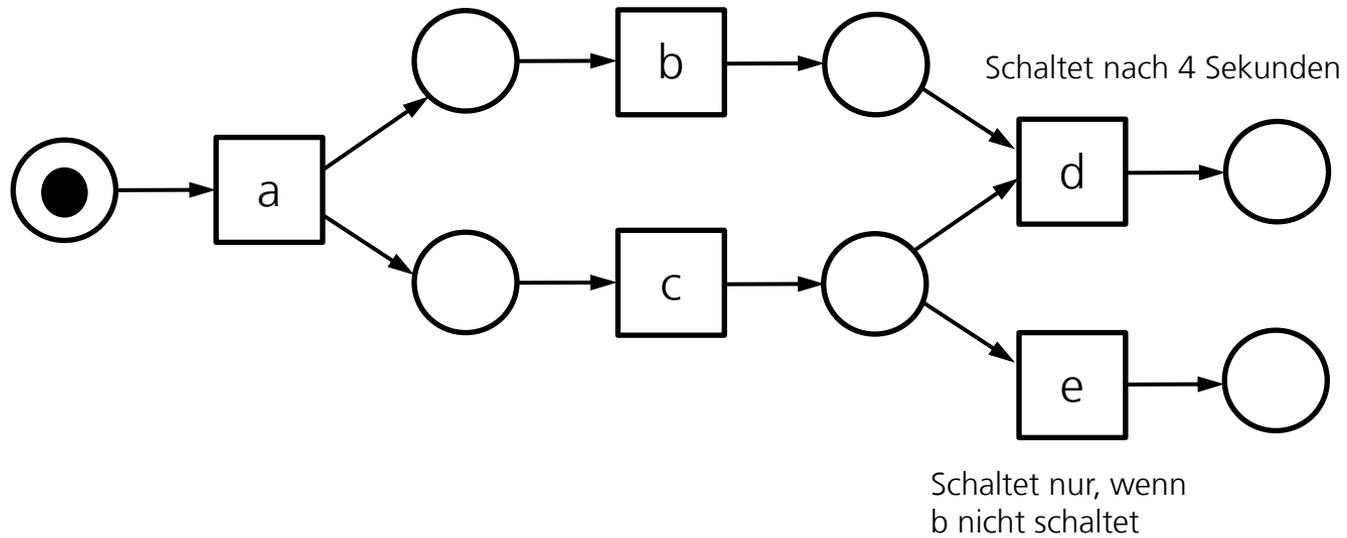
# Historische Perspektive und Bezug zwischen Sprachen



## Wann ist ein Modell ungeeignet?



# Wann ist ein Modell ungeeignet?



## Wann ist ein Modell ungeeignet?

- UML für Real-Time Anwendungen in der Ingenieurwissenschaft
- UML für Software-Architektur
- Petrinetze für nichtverteilte System
- ...

## Mathematik (und Logik) als Modellierungssprache

- Wozu lernen Informatikstudenten Mathematik?
- Warum sind Mathematiker in der Softwareindustrie begehrt?
- Was kann man mit Mathematik modellieren?
- Ein mathematisches Modell ist präzise und hat scharfe Grenzen
- Mit mathematischen Modellen kann man rechnen, beweisen, transformieren

## Einbettung der Modellierung in Informatik-Studiengänge

- Modellierung in der GI: **Querschnitts**fachausschuss
- Modellierung ist auch im Studium Querschnittsthema!
- Modellierung am Anfang oder am Ende des Studiums?
- Abschlussprüfungen (á la Diplomprüfungen) fehlen
- ...

# Lehren modellieren

## Studiengänge modellieren

- Wie berücksichtigt Lehre individuelle fachliche Kompetenzen?
- Wie berücksichtigt Lehre individuelle überfachliche Kompetenzen?
- Wie berücksichtigen Studiengänge nachgewiesene Kompetenzen?
  - ... aus anderen Hochschulen: Anerkennung
  - ... aus Beruf, Ausbildung usw.: Anrechnung

## Studiengänge modellieren

- Wie berücksichtigt Lehre individuelle fachliche Kompetenzen?
- Wie berücksichtigt Lehre individuelle überfachliche Kompetenzen?
- **Wie berücksichtigen Studiengänge nachgewiesene Kompetenzen?**
  - ... aus anderen Hochschulen: **Anerkennung**
- - ... aus Beruf, Ausbildung usw.: **Anrechnung**

## Szenario 1

**Studentin A hat eine ähnlich benannte Vorlesung bereits an einer anderen Vorlesung erfolgreich besucht**

Anerkennung, wenn die Kompetenzen (und Inhalte) der **Modulbeschreibung** annähernd übereinstimmen, und nur dann!

Workload (Credit points) spielen eine untergeordnete Rolle

## Szenario 2

**Studentin B hat Kompetenzen (und Inhalte) in anderen Vorlesungen bereits erworben, alle Prüfungen bestanden**

... wie bei Szenario 1

## Szenario 3

### **Studentin C hat eine andere Programmiersprache als in Vorlesung behandelt gelernt (C++ statt Java)**

- in Modulhandbuch steht: kann Programme in Java entwickeln
- Welche Rolle spielt diese Vorlesung in dem Studienverlauf?
- Geht es eigentlich nur um objektorientiertes Programmieren (natürlich mit Beispiel)?
- Gibt es eine weitere Vorlesung C++?

## Szenario 4

**Studentin D hat ein anderes Nebenfach abgeschlossen, das im Rahmen des Studiengangs nicht angeboten wird**

- Modulbeschreibungen unterscheiden sich deutlich!
- Welche Rolle spielt das Nebenfach?
- Vielleicht nur: „schauen über den Tellerrand“?

## Szenario 5

**Studentin E hat im Rahmen von ERASMUS bei einer Partneruniversität eine Vorlesung besucht, die zu ihrem Studienschwerpunkt sehr gut passt, an der Heimatuniversität aber nicht angeboten werden kann**

- Modulbeschreibungen unterscheiden sich deutlich!
- Welche Rolle spielt der Studienschwerpunkt

## Szenario 6

**Studentin F hat fast das vollständige Informatikstudium an einer Fachhochschule absolviert, ihr fehlt nur noch die Abschlussarbeit**

- Welche Rolle spielt Ausrichtung des Studiengangs an der anderen Hochschule?

## Szenario 7

**Studentin G hat einen Bachelor in XYZ-Informatik in der Schweiz erworben.**

- Kann sie für den Master Informatik eingeschrieben werden?
- Welche Rolle spielen Empfehlungen für die Studienganggestaltung (zum Beispiel von GI, IEEE, ...)?

## Szenario 8

**Student H hat eine Ausbildung als Fachinformatiker absolviert.  
Dabei hat er Programmieren in Java gelernt**

- Kann dies für die Vorlesung Programmieren in Java angerechnet werden?

## Szenario 9

**Student H hat eine Ausbildung als Fachinformatiker absolviert. Dabei hat er Datenbankentwurf und SQL gelernt. Normalformen kann er intuitiv, aber nicht formal benennen**

- Kann dies für die Vorlesung Datenbanken angerechnet werden?
- Wie kann man Teile von Modulen anrechnen?
- Wie muss eine Datenbankvorlesung für ihn gestaltet werden?  
Welchen Workload hat diese Vorlesung für ihn?

## Szenario 10

**Student I hat eine Ausbildung als Fachinformatiker absolviert.  
Darüber hinaus hat er weitreichende Projekterfahrung.  
Insbesondere hat er die Steuerungssoftware für einen medizinischen  
Roboter entworfen.**

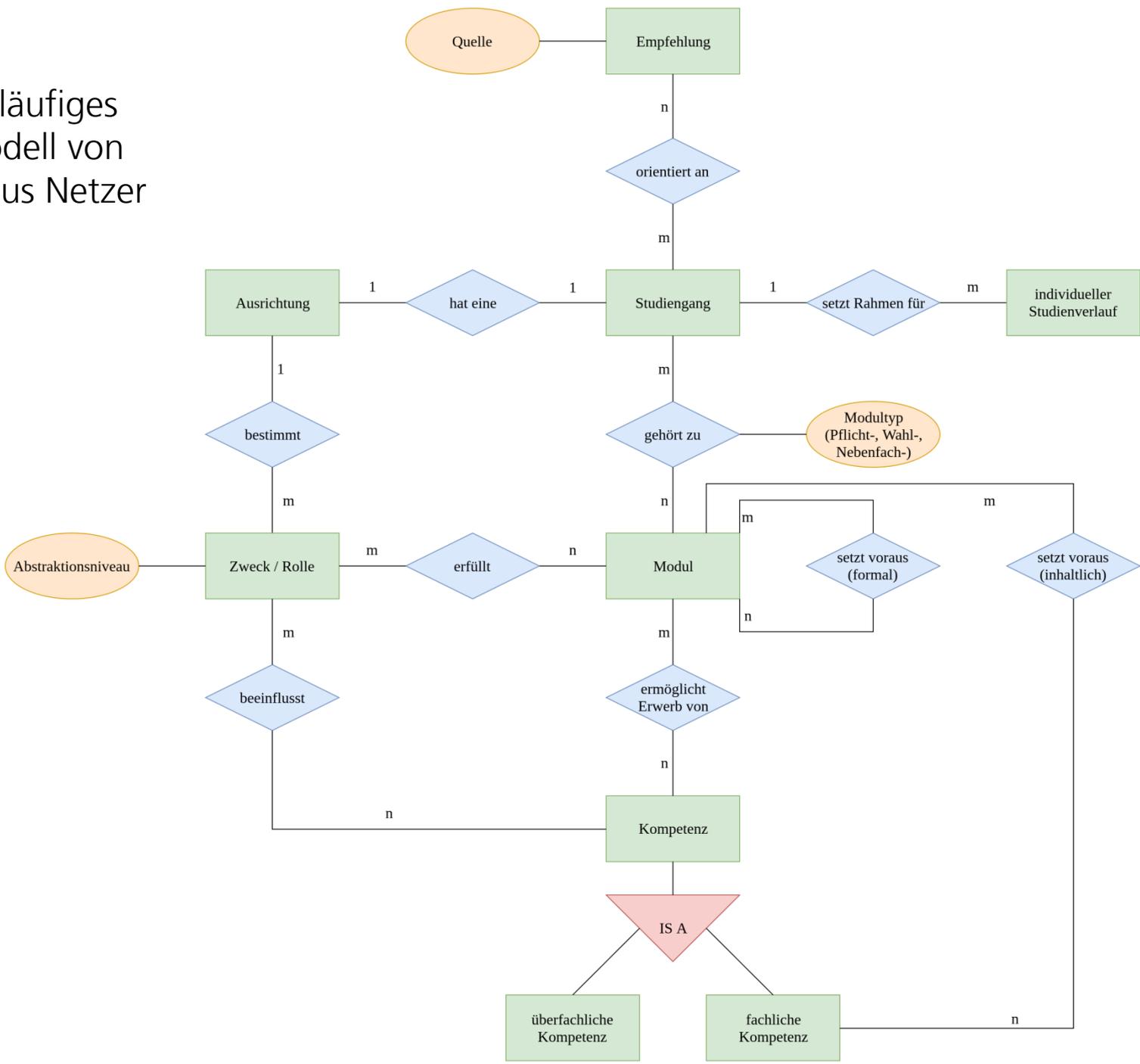
- Kann dies als Nebenfach angerechnet werden?
- Ist dies nicht auch „über den Tellerrand schauen“?

## Wir benötigen Anrechnungs- und Anerkennungsprinzipien, die über Modulbeschreibungen hinausgehen!

### Zu berücksichtigen:

- Kompetenzen und Inhalte von Teilen von Modulen
- Kompetenzen und Inhalte von Modulen
- Rolle von Modulen in Studiengängen  
(dabei auch Unterscheidung von Wahl und Pflicht)
- Vertiefungsrichtungen
- Ausrichtungen von Studiengängen an Hochschulen
- Generelle Charakteristika von Studiengängen eines Faches und Empfehlungen
- Vorschriften zur Gestaltung von Studiengängen allgemein  
(KMK, Hochschulgesetz, ...)

# vorläufiges Modell von Cajus Netzer





Fakultät für  
**Mathematik und  
Informatik**

# Modellieren lehren Lehren modellieren

Jörg Desel

Fakultät für **Mathematik und Informatik**

Lehrgebiet Softwaretechnik und Theorie der Programmierung  
Prof. Dr. Jörg Desel