Formale Methoden der Softwaretechnik 1
Vorlesung vom 19.10.09: Einführung

Christoph Lüth, Lutz Schröder

WS 09/10

Organisatorisches

- Veranstalter:

Christoph Lüth	Lutz Schröder
christoph.lueth@dfki.de	lutz.schroeder@dfki.de
Cartesium 2.043, Tel. 64223	Cartesium 2.051, Tel. 64216

- Termine:

Termine:	Vorlesung:	Montag, 12 – 14, MZH 7210
	Übung:	Mittwoch, 12 – 14, MZH 7220

Therac-25

- Neuartiger Linearbeschleuniger in der Strahlentherapie.
  - Computergesteuert (PDP-11, Assembler)
- Fünf Unfälle mit Todesfolge (1985– 1987)
  - Zu hohe Strahlendosis (4000 – 20000 rad, letal 1000 rad)
- Problem: Softwarefehler
  - Ein einzelner Programmierer (fünf Jahre)
  - Alles in Assembler, kein Betriebssystem
  - Programmierer auch Tester (Qualitätskontrolle)

Ariane-5

Die Vasa

Lernziele

1. Modellierung — Formulierung von Spezifikationen

2. Formaler Beweis — Nachweis von Eigenschaften

3. Verifikation — Beweis der Korrektheit von Programmen

Darüber hinaus:

- Vertrautheit mit aktuellen Techniken

Themen

- Grundlagen:
  - Formale Logik, formales Beweisen
- Anwendung:
  - Der Theorembeweiser Isabelle
- Formale Spezifikation und Verifikation
  - Funktionale Programme
  - Imperative Programme

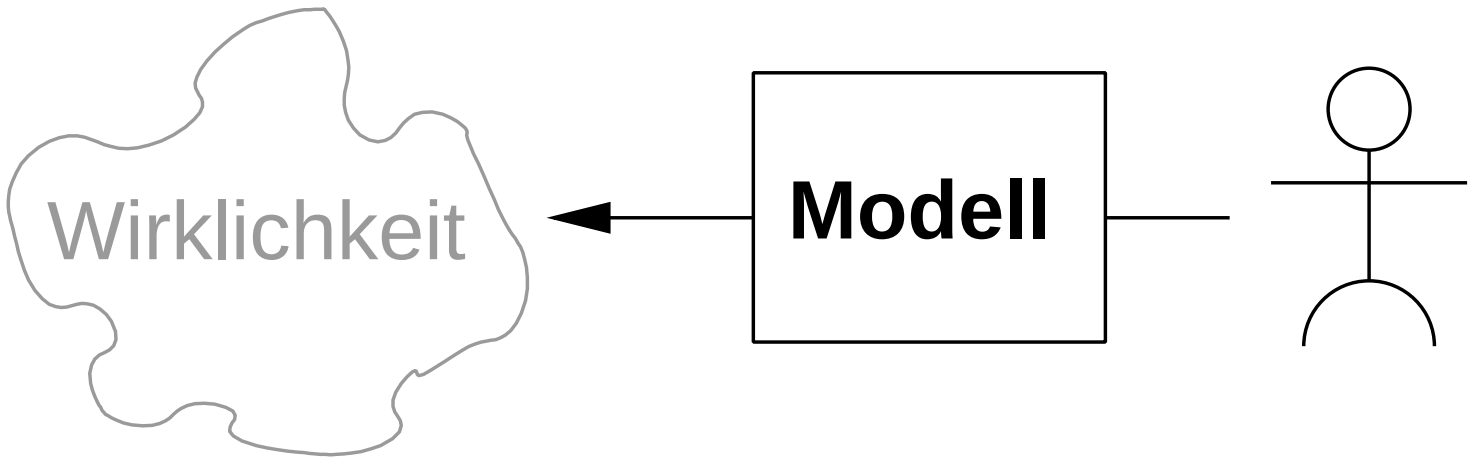
Plan

- Nächste sieben Wochen:
  - Formale Logik und formaler Beweis
  - Vorlesung: Grundlagen
  - Übung: Isabelle
- Nach Weihnachten:
  - Grundlagen der Verifikation imperativer Programme
  - Semantik, Hoare-Kalkül.

Der Theorembeweiser Isabelle

- Interaktiver Theorembeweiser
- Entwickelt in Cambridge und München
- Est. 1993 (?), ca. 500 Benutzer
- Andere: PVS, Coq, ACL-2
- Vielfältig benutzt:
  - VeriSoft (D) — `http://www.verisoft.de`
  - L4.verified (AUS) — `http://ertos.nicta.com.au/research/l4.verified/`
  - SAMS (Bremen) — `http://www.projekt-sams.de`

Das Problem



Formale Logik

- Formale (symbolische) Logik: Rechnen mit Symbolen
- Programme: Symbolmanipulation
- Auswertung: Beweis
- Curry-Howard-Isomorphie: funktionale Programme $\cong$ konstruktiver Beweis

Geschichte

- Gottlob Frege (1848– 1942)
  - 'Begriffsschrift, eine der arithmetischen nachgebildete Formelsprache des reinen Denkens' (1879)
- Georg Cantor (1845– 1918), Bertrand Russel (1872– 1970), Ernst Zermelo (1871– 1953)
  - Einfache Mengenlehre: inkonsistent (Russel's Paradox)
  - Axiomatische Mengenlehre: Zermelo-Fränkel
- David Hilbert (1862– 1943)
  - Hilbert's Programm: 'mechanisierte' Beweistheorie
- Kurt Gödel (1906– 1978)
  - Vollständigkeitssatz, Unvollständigkeitssätze

Grundbegriffe der formalen Logik

- Ableitbarkeit $\mathcal{T}h \vdash P$
  - Syntaktische Folgerungung
- Gültigkeit $\mathcal{T}h \models P$
  - Semantische Folgerung — hier nicht relevant
- Klassische Logik: $P \vee \neg P$
- Entscheidbarkeit
  - Aussagenlogik
- Konsistenz: $\mathcal{T}h \nvdash \bot$
  - Nicht alles ableitbar
- Vollständigkeit: jede gültige Aussage ableitbar
  - Prädikatenlogik erster Stufe

Unvollständigkeit

- Gödels 1. Unvollständigkeitssatz:
  - Jede Logik, die Peano-Arithmetik formalisiert, ist entweder inkonsistent oder unvollständig.
- Gödels 2. Unvollständigkeitssatz:
  - Jeder Logik, die ihre eigene Konsistenz beweist, ist inkonsistent.
- Auswirkungen:
  - Hilbert's Programm terminiert nicht.
  - Programme nicht vollständig spezifierbar.
  - Spezifikationssprachen immer unvollständig (oder uninteressant).
  - Mit anderen Worten: Es bleibt spannend.