

SEMINAR "MODEL-CHECKING"

Dirk Fahland

fahland@informatik.hu-berlin.de

www.informatik.hu-berlin.de/top/lehre/WS09-10/se_mc/

VERIFIKATION

- **Testen:** kann nur die Anwesenheit von Fehlern feststellen, nicht ihre Abwesenheit. (E. Dijkstra)
- **Konstruktion:** Erzeugen des Systems aus der Spezifikation
- **Verifikation:** Nachweis der Korrektheit (theoretisch)
 - Fähigkeit, subtile Fehler zu finden (praktisch)

ANWENDUNGSGEBIETE

- Safety critical systems (Autos, Flugzeuge, Medizintechnik, Atomkraftwerke...)
- Mission critical systems (Raumsonden)
- Hardware
- Standards, Protokolle
- Imagerträgliche Software (Windows, Office, ...)
- Tendenz wachsend.....

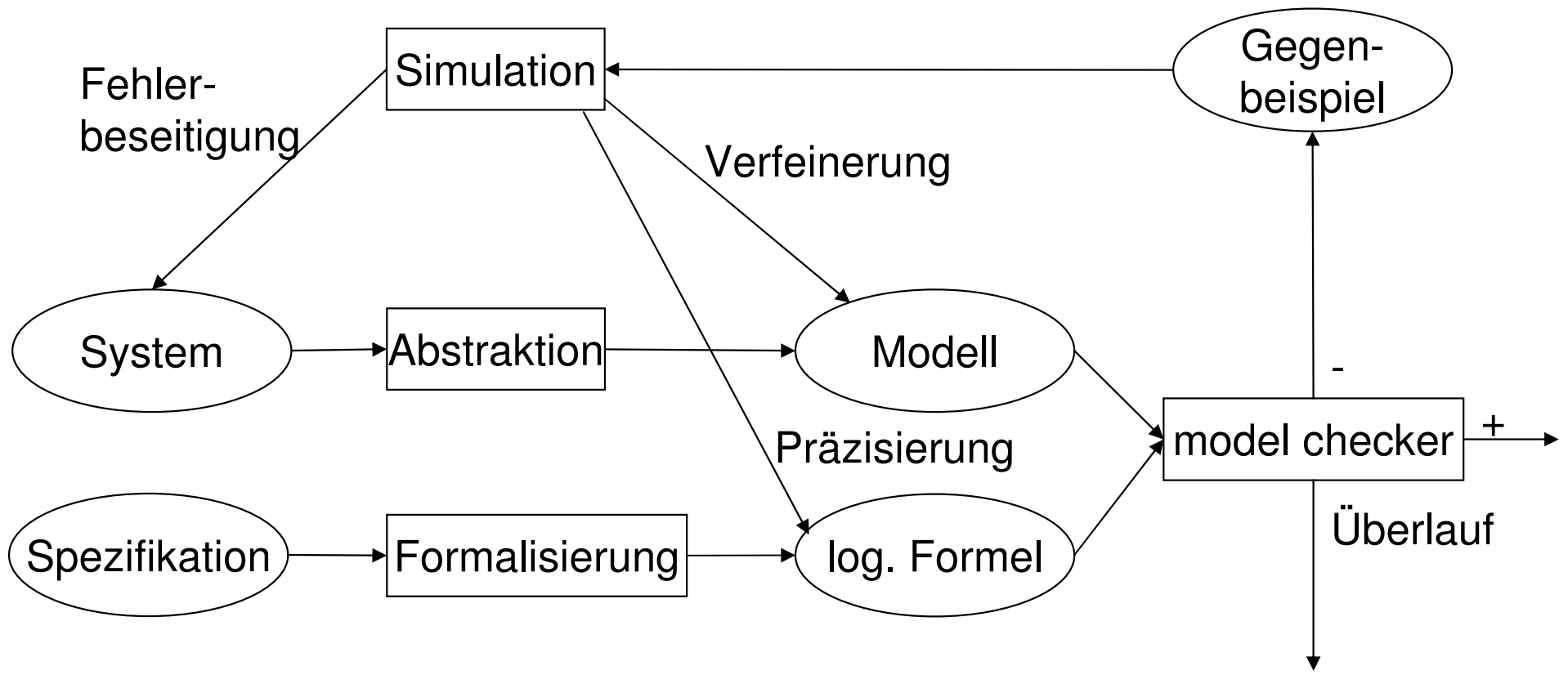
WAS IST MODELCHECKING?

- Erschöpfende Durchmusterung der Zustände eines Systems zur Prüfung einer vorgegebenen Eigenschaft
- erster Ansatz: 1986
- Durchbruch: 1992
- inzwischen: einige Erfolgsgeschichten, Einsatz in einigen Firmen, stetige Steigerung der Leistungsfähigkeit
- Grundproblem: Zustandsexplosion

ZUSTANDSEXPLOSION

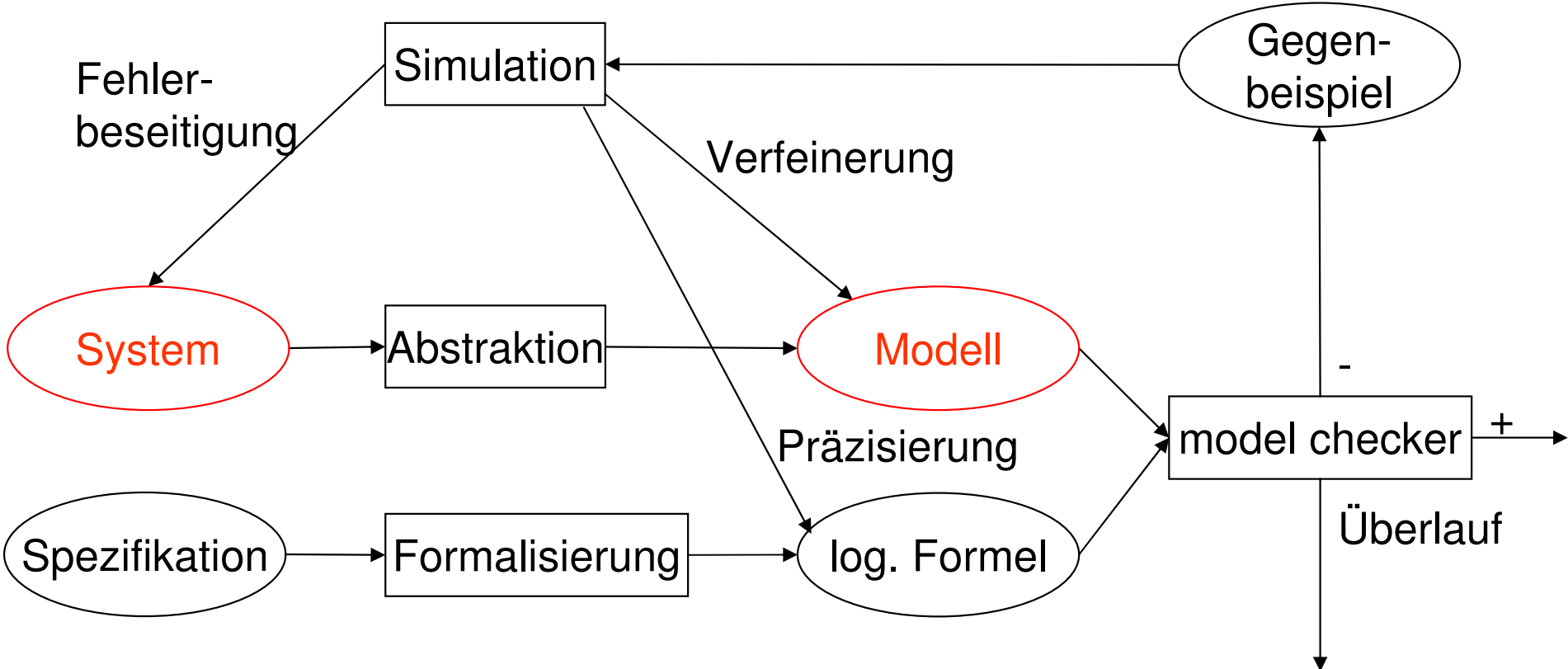
- Wieviele Zustände kann man in welcher Zeit durchmustern?
- Annahme: 2.4 GHz, genug Speicher, ein *neuer* Zustand pro Prozessorzyklus
 - 2,400,000,000 pro Sekunde
 - 144,000,000,000 pro Minute
 - 8,840,000,000,000 pro Stunde
 - 207,360,000,000,000 pro Tag
 - 75,738,240,000,000,000 pro Jahr
 - 1,514,764,800,000,000,000,000,000,000 seit Urknall ($< 10^{28}$)

MODEL CHECKING: ÜBERBLICK

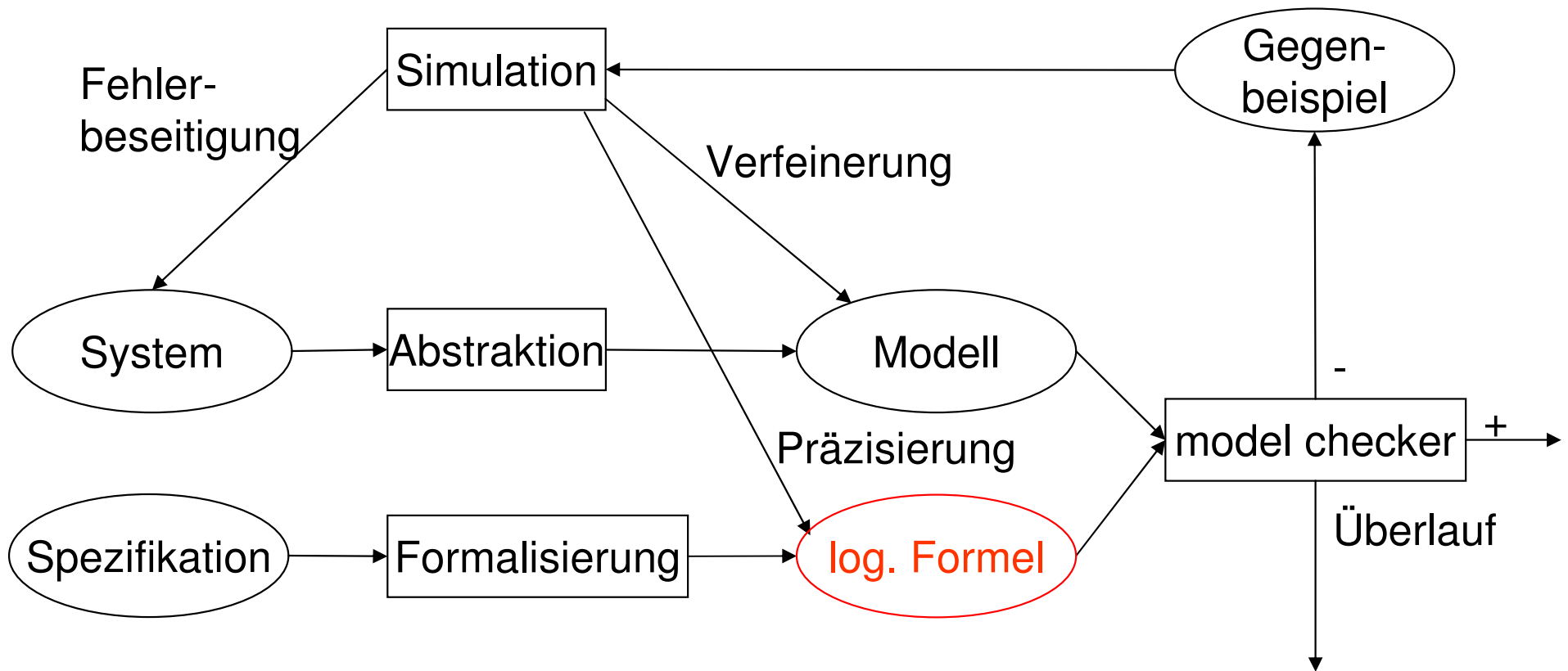


SEMINAR-INHALT

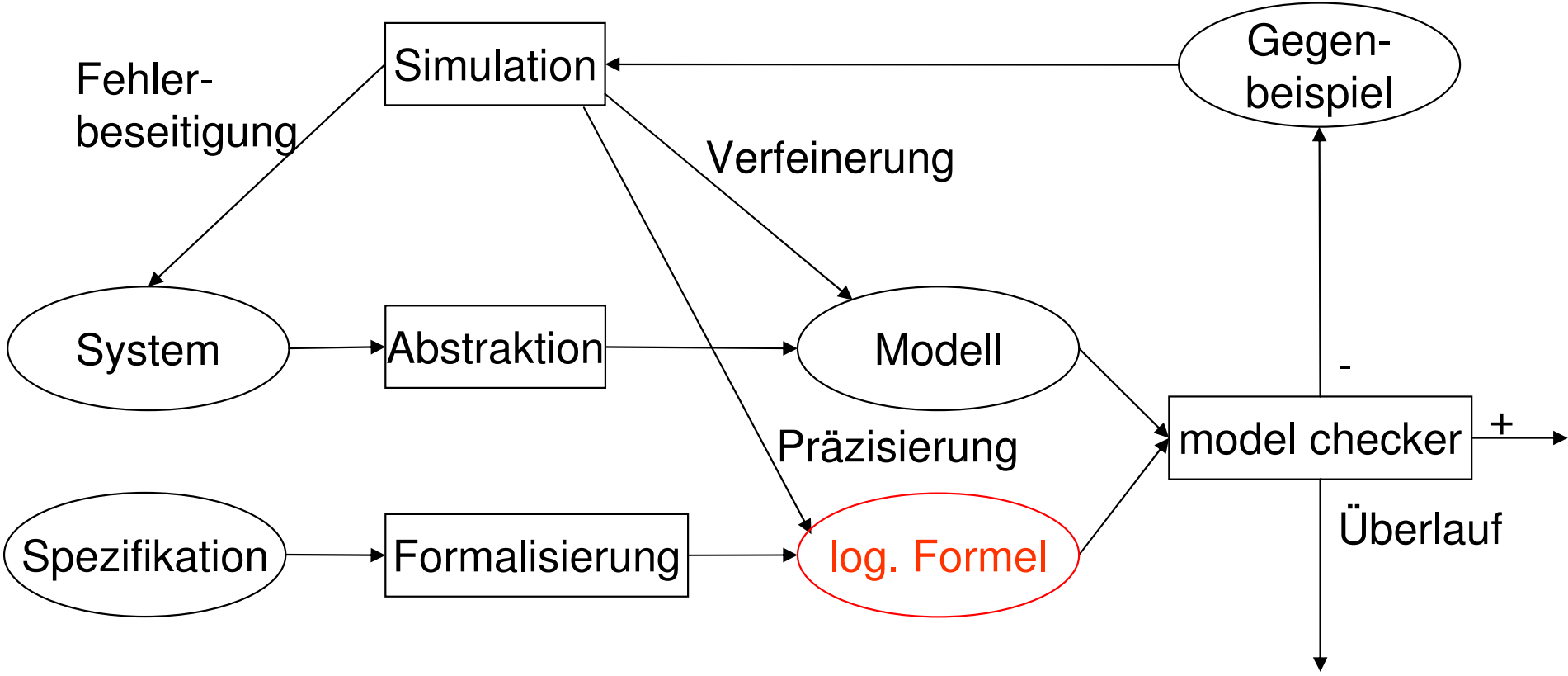
- System und Modelle



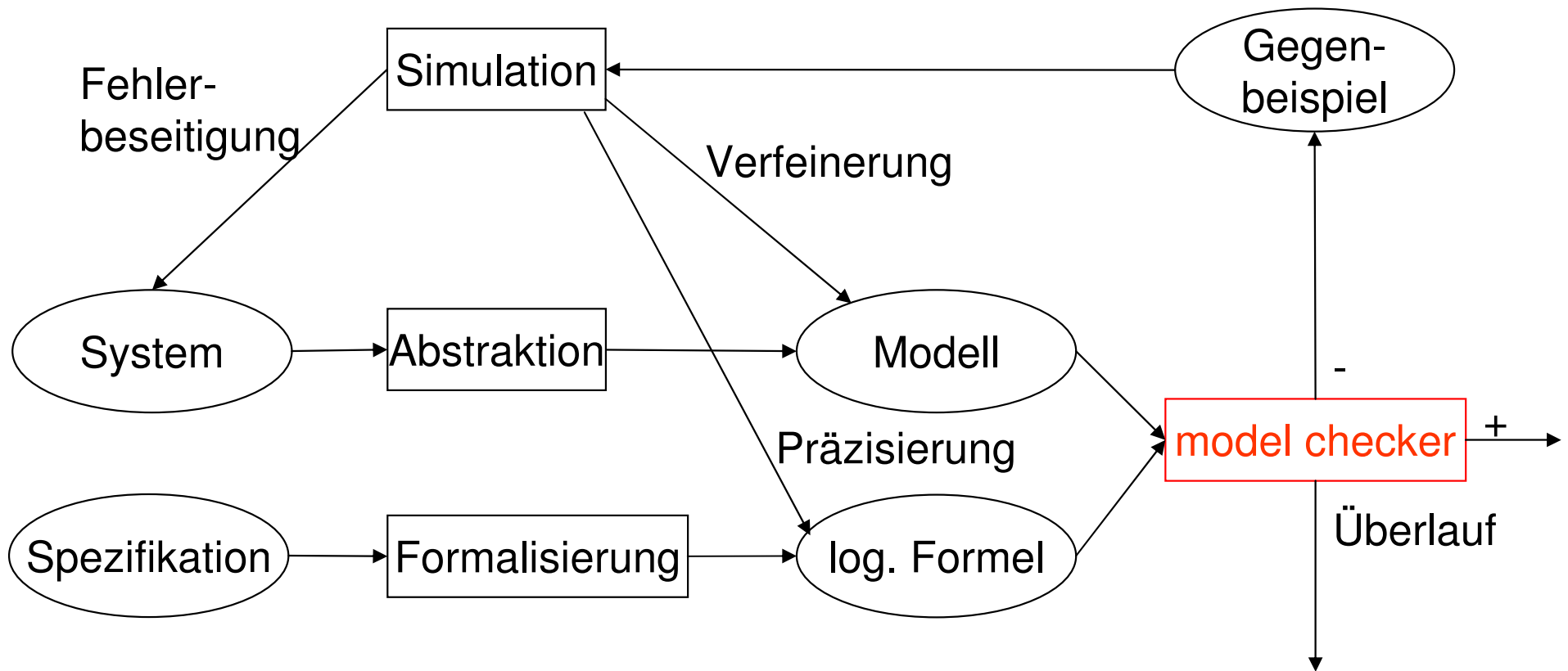
- Temporale Logiken

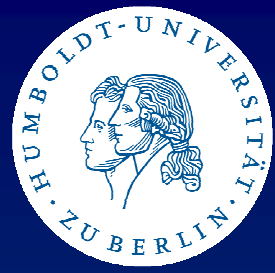


- Eigenschaften von Systemen



- Explizites und symbolisches MC





SEMINAR-INHALT - KONKRET

THEMEN I

1. System

1. Was ist ein System? Was ist ein Zustand?
Systemgrößen, Arten von Systemen

2. Modelle

1. Petrinetze
2. Zustandsautomaten/ Kripkestrukturen

3. Temporale Logiken

1. CTL, LTL

4. Eigenschaften von Systemen

1. Sicherheit und Lebendigkeit
2. Nebenläufigkeit, Fairness
3. Verifikation verteilter Systeme

5. Äquivalenzklassen

1. Traces, ..., Simulation, Bisimulation

6. Explizites Modelchecking

1. LTL-Modelchecking
2. CTL-Modelchecking
3. Reduktion
 1. Symmetrien
 2. Partial Order Reduction

7. Symbolisches Modelchecking

1. BDD basiertes CTL-Modelchecking
2. SAT basiertes Modelchecking

8. Timed Automata

1. Modellierung
2. Timed-Modelchecking

9. Software-Modelchecking

1. Abstrakte Interpretation/Predicate Abstraction

10. Verteilte Systeme II

1. Unfoldings
2. Verifikation offener Systeme

- Ausarbeitung eines Themas
 - selbständige Literaturrecherche
 - Vortrag halten – mindestens 45min, Diskussion leiten
 - Handout zum Vortrag, bis 5 Seiten
- der Vortrag
 - das Thema möglichst anhand von Beispielen veranschaulichen
 - Zuhörer nicht mit Definitionen überschütten
 - Formalismen, Definitionen in Maßen einsetzen
 - ca. 2min pro Folie einplanen
- Scheinvergabe
 - am Ende des Semesters, wenn
 - Vortrag gut gehalten
 - Seminar besucht