

# Test von adaptiven Software- mechanismen zur Fehlerkompensation

Dr. Olaf Maibaum

Deutsches Zentrum für Luft- und Raumfahrt e.V.

Simulations- und Softwaretechnik



webdynamix

Gefördert durch



Bundesministerium  
für Bildung  
und Forschung

# Übersicht

- Adaptive Mechanismen zur Fehlerkompensation
- Aufbau der Simulationsumgebung
- Durchführung der Tests
  - Auswahl der Testfälle
  - Aufbau der Testumgebung
  - Durchführung der Testfälle
- Zusammenfassung

## • Technische Gründe

- ◆ Alterung der Sensorik und Aktuatorik
- ◆ Ausfall von Sensorik und Aktuatorik
- ◆ Unterbrechung von Kommunikationsverbindungen
- ◆ Sicherheitskritische Rahmenbedingungen

## • Soziologische Gründe

- ◆ Längere Wartungsintervalle
- ◆ Lange Nutzungszeiten
- ◆ Kundenzufriedenheit

## ❶ Kalte Redundanz

- Hardware wird abgeschaltet mitgeführt
- Inbetriebnahme nach Ausfall der primären Hardware
- Lebenszeitverlängerung
- Im Fahrzeug nur bedingt nötig, da Wartung möglich ist

## ❷ Warme Redundanz

- Hardware läuft mit
- Bei erkannten Defekten wird die warm redundante Hardware verwendet
- Votingmechanismen

## • Filtermechanismen

- ◆ Berücksichtigung einer Zustandschätzung (Kalmanfilter)

- ◆ Anpassen der verwendeten Rauschfunktion

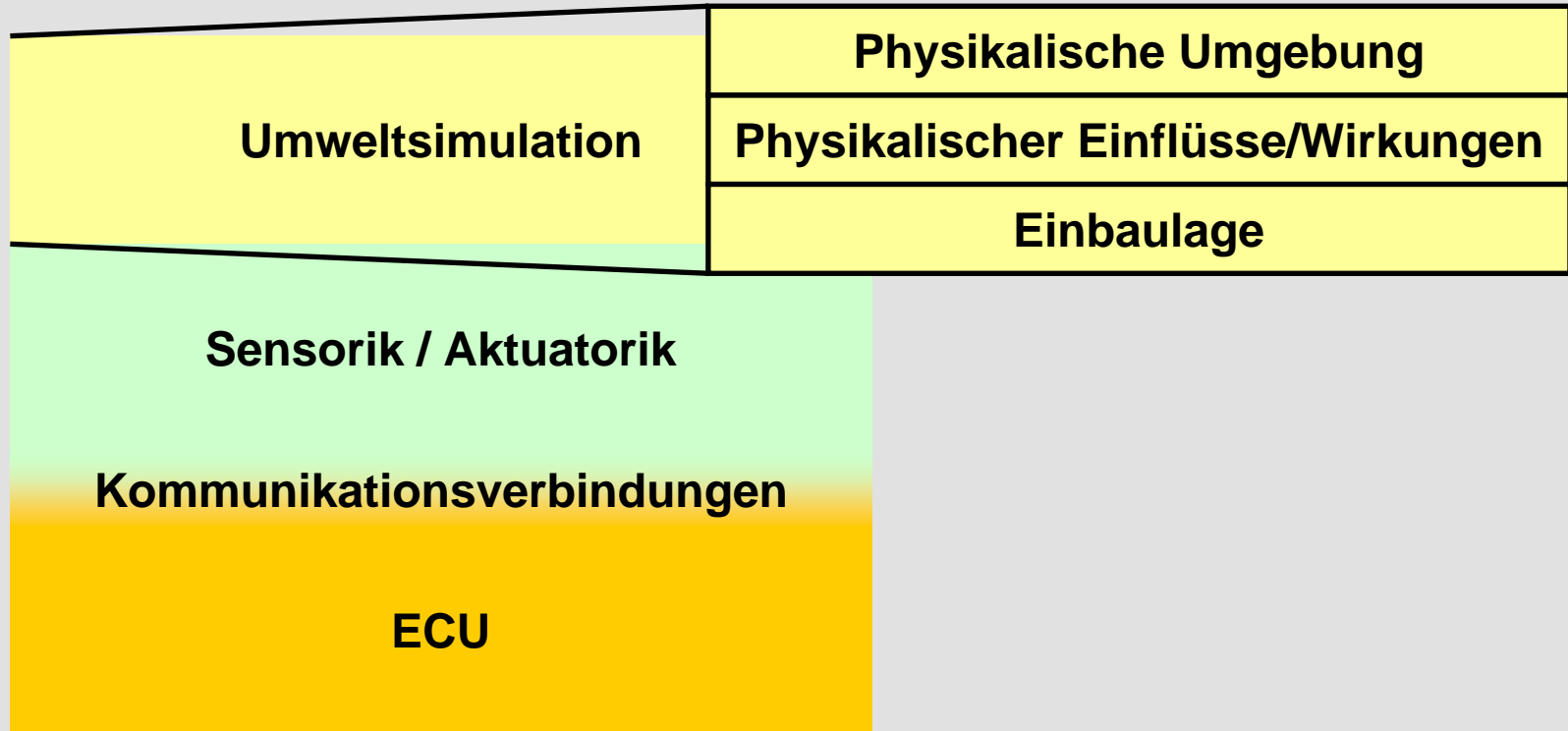
## • Künstliche Sensorik

- ◆ Zusammenfassen mehrerer Sensoren zu einem Sensor

- ◆ Ableiten von Messwerten aus nicht originär für diesen Zweck vorgesehenen Sensorwerten

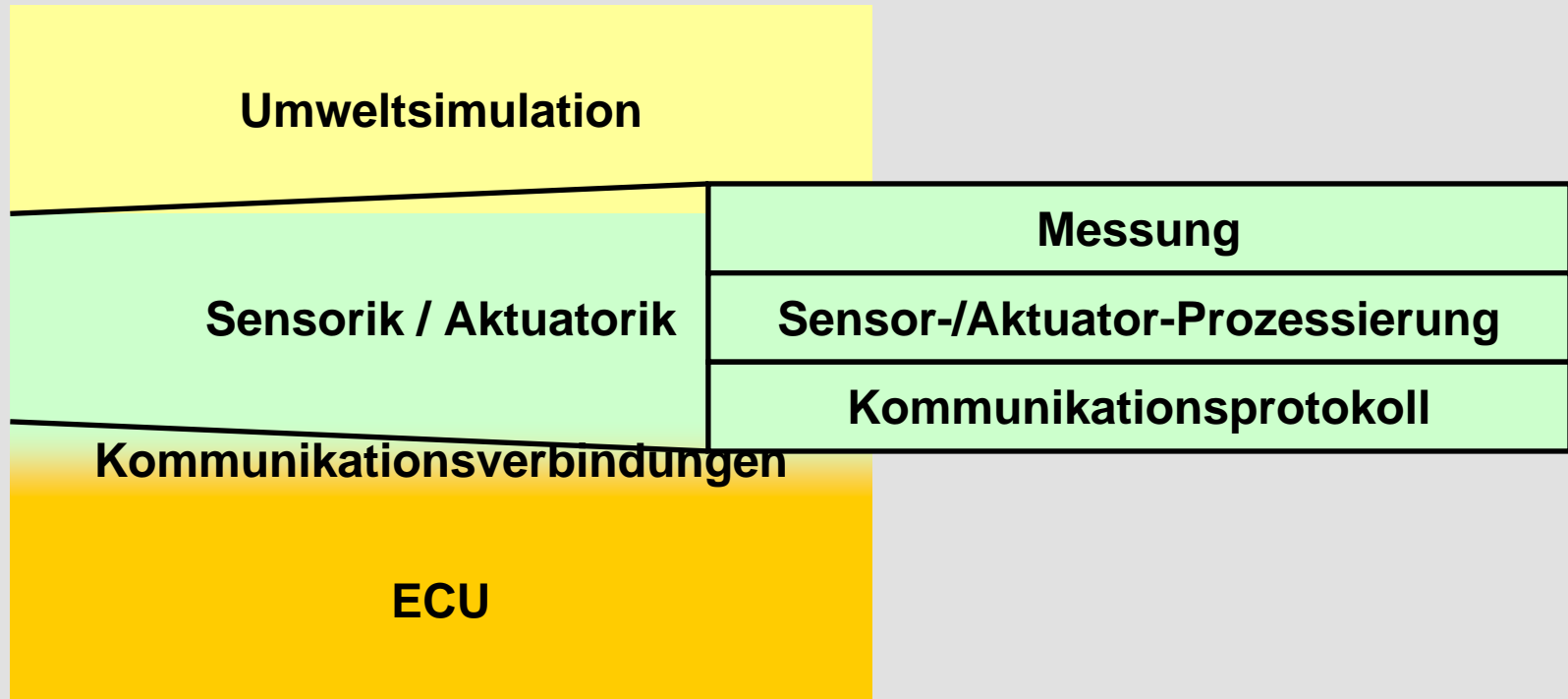
# Aufbau der Simulationsumgebung

## ● Schichtenmodell



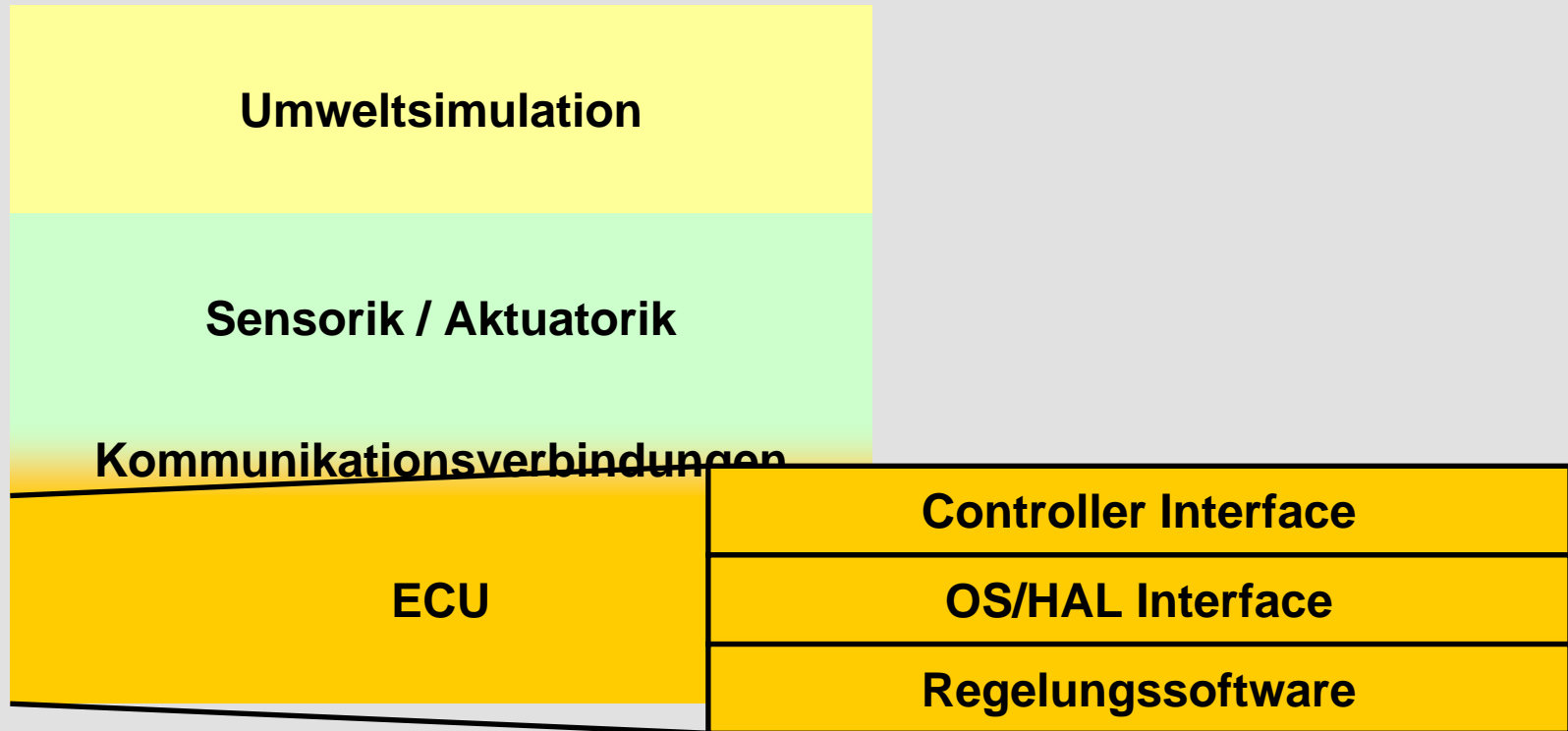
# Aufbau der Simulationsumgebung

## ● Schichtenmodell



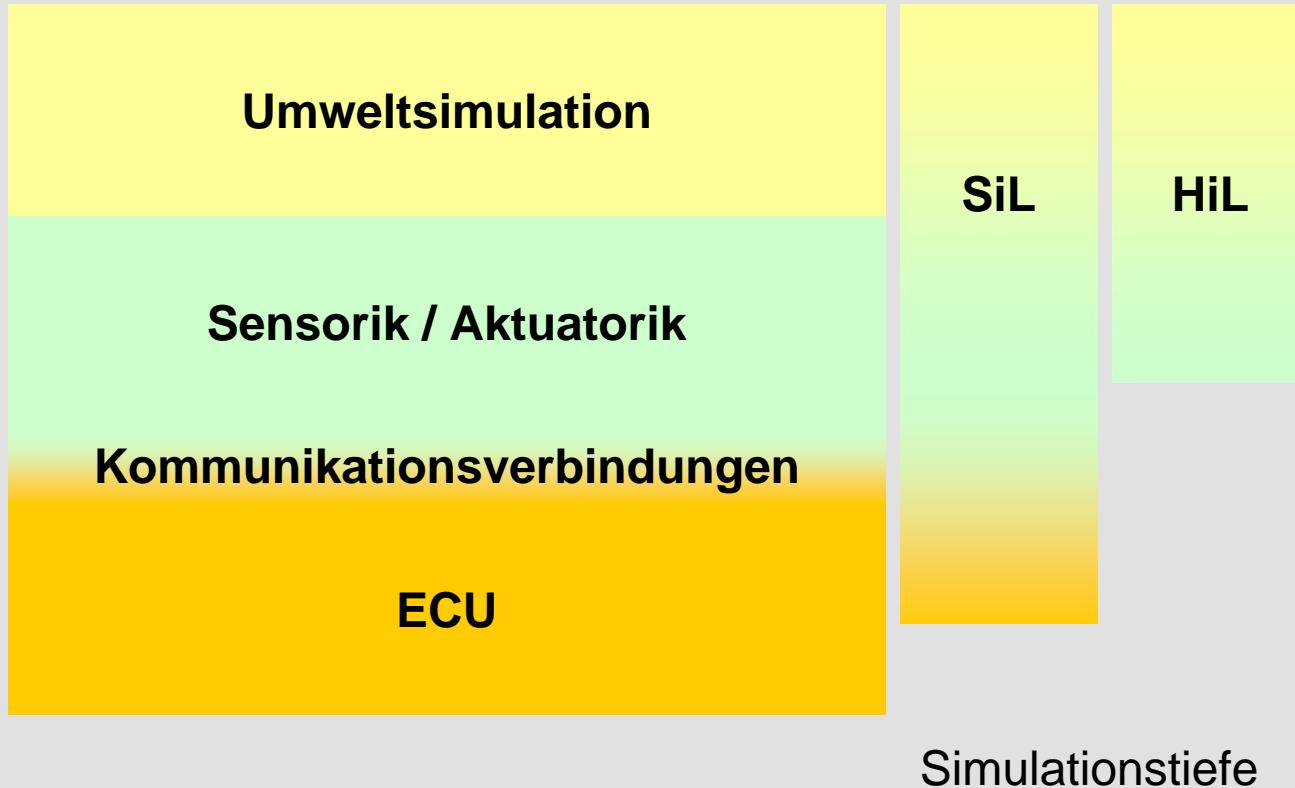
# Aufbau der Simulationsumgebung

## ● Schichtenmodell



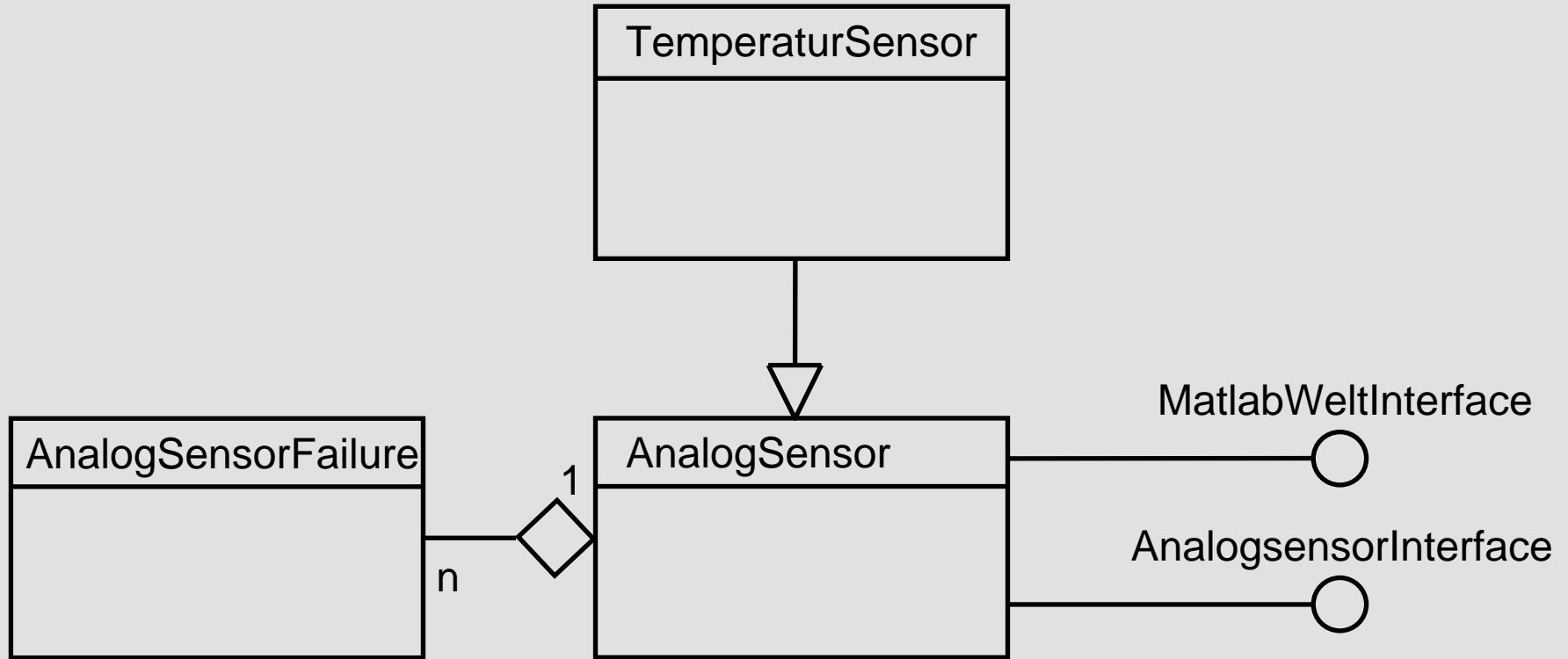
# Aufbau der Simulationsumgebung

## ● Unterschied HiL-Test und SiL-Test

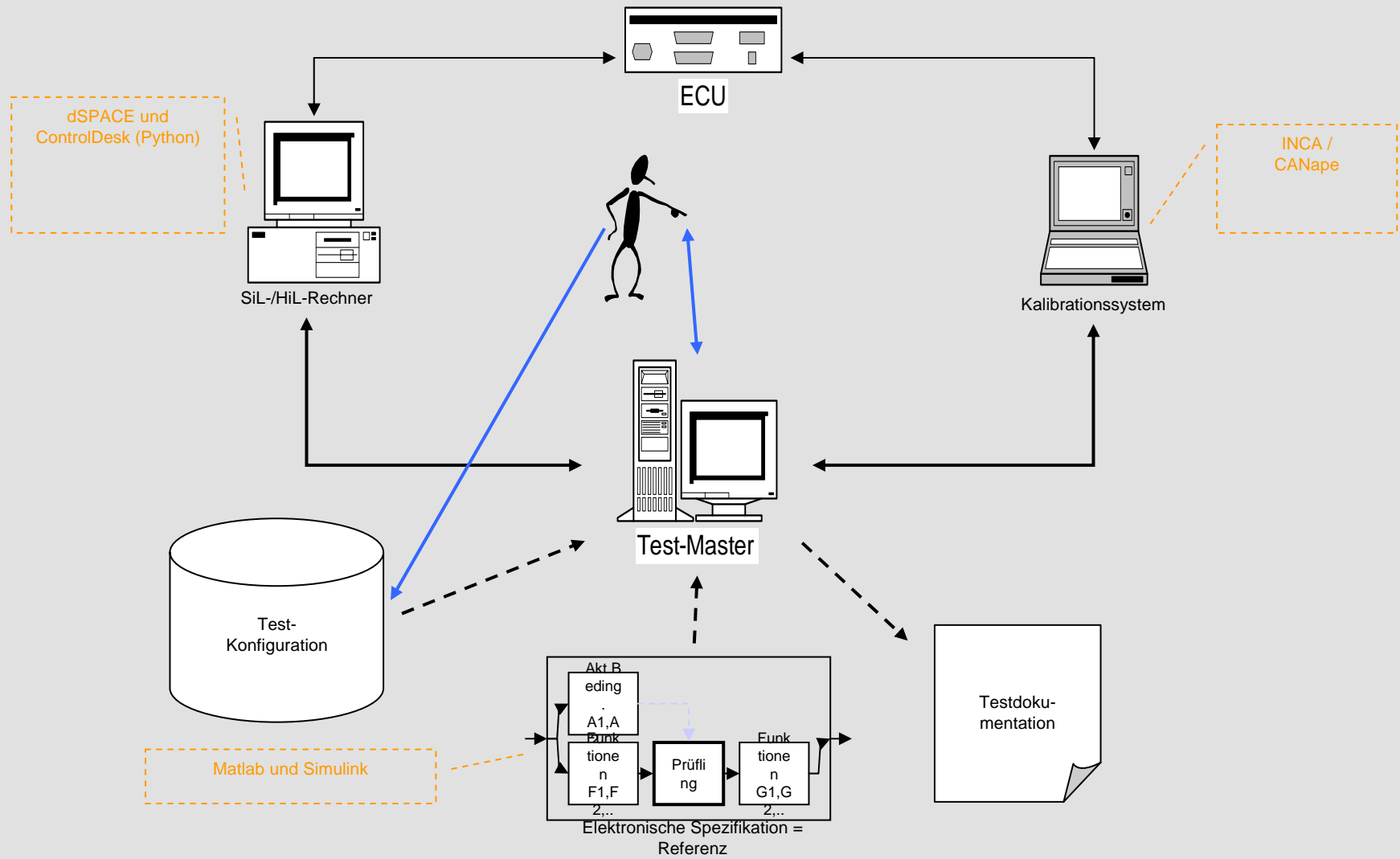


# Aufbau der Simulationsumgebung

## UML-Modellierung der Simulation



# Aufbau der Testumgebung



# Auswahl der Testfälle

## ● Ereignis

### ◆ Betriebsszenario

- Nutzeraktivität
- Änderungen in der Umwelt

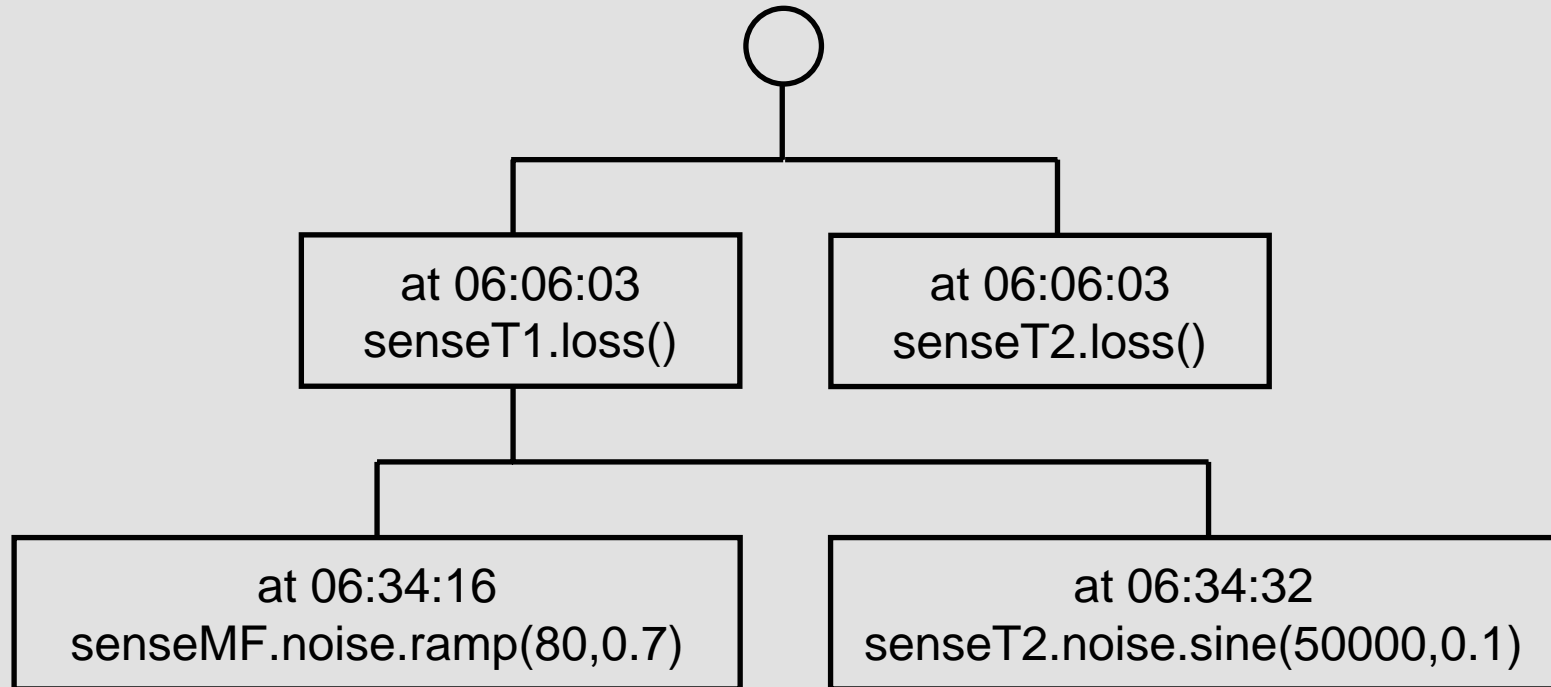
### ◆ Versagensszenario

- Ausfall von Hardware
- Defekt oder Störung in der Hardware
  
- Ermittlung auf Basis einer FMEA

FMEA = Failure Modes and Effect Analysis

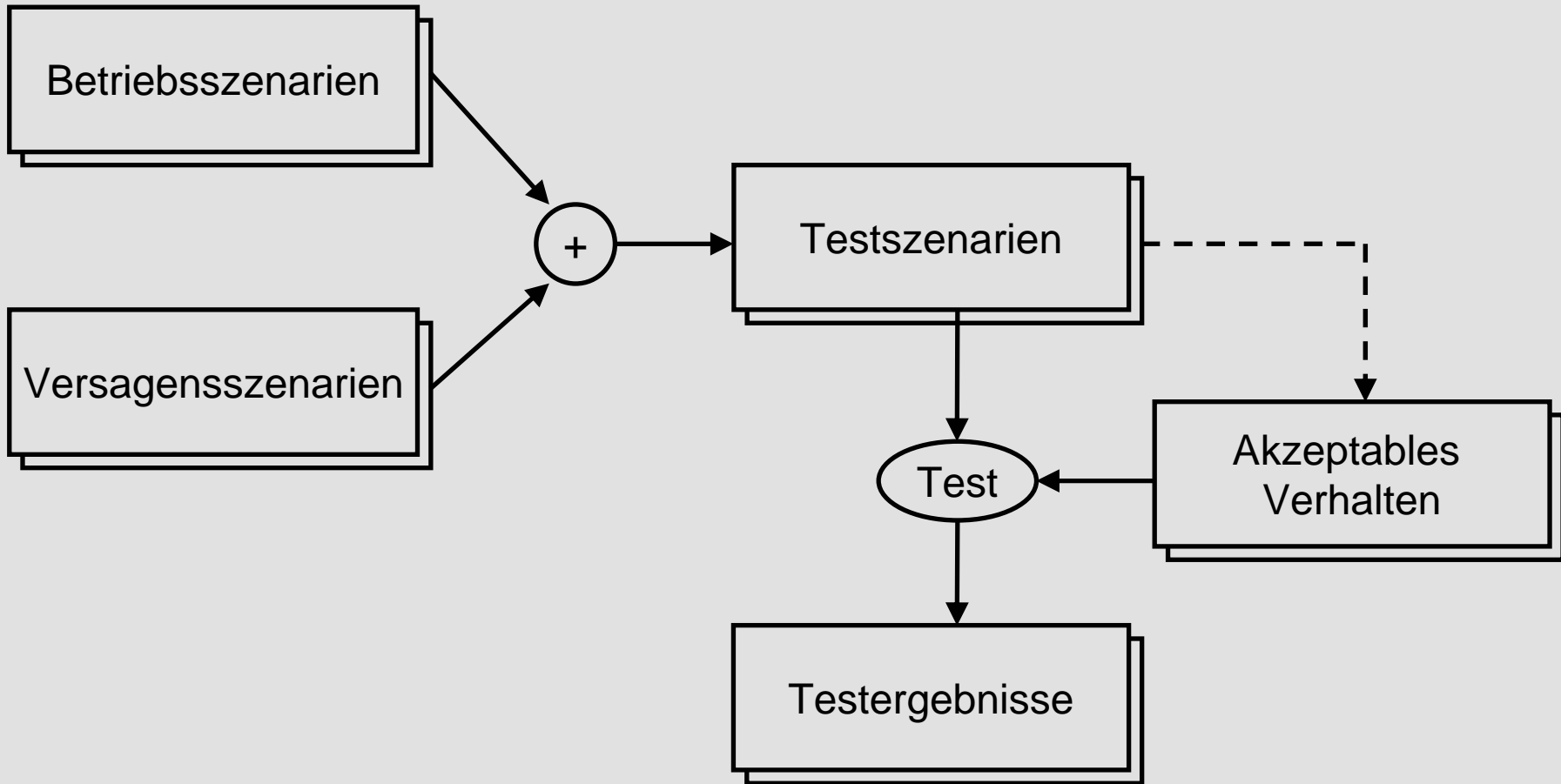
# Auswahl der Testfälle

## ● Ereignisbaum



# Durchführung der Tests

## ● Testprozess



# Durchführung der Tests

- Start-Stopp-Betrieb
  - ◆ Betrieb des Systems bis zu einem definierten Zeitpunkt
  - ◆ Stoppen der Simulation und der ECU und Anlegen eines Datendumps
  - ◆ Verwenden des Datendumps als Initialwert für weitere Testabläufe
  - ◆ Erweiterte Debugging-Möglichkeiten

# Zusammenfassung

- Schichtenbasierter und objektorientierter Aufbau der Umgebungssimulation
- Software und Hardware in the Loop Test
- Überprüfung des Softwareverhaltens in Ausfallsituationen
- Gewinnung der Testfälle aus einer FMEA
- Betrachtung von Mehrfachfehlern und Alterungserscheinungen im Test
- Wiederaufsetzen von weiteren Tests auf gewonnenen Zuständen