

Das Orakelproblem beim Testen

DAS UNTESTBARE TESTEN

Matthias Hamburg

Software QS-Tag

Frankfurt am Main, 7. Nov. 2024

Matthias Hamburg

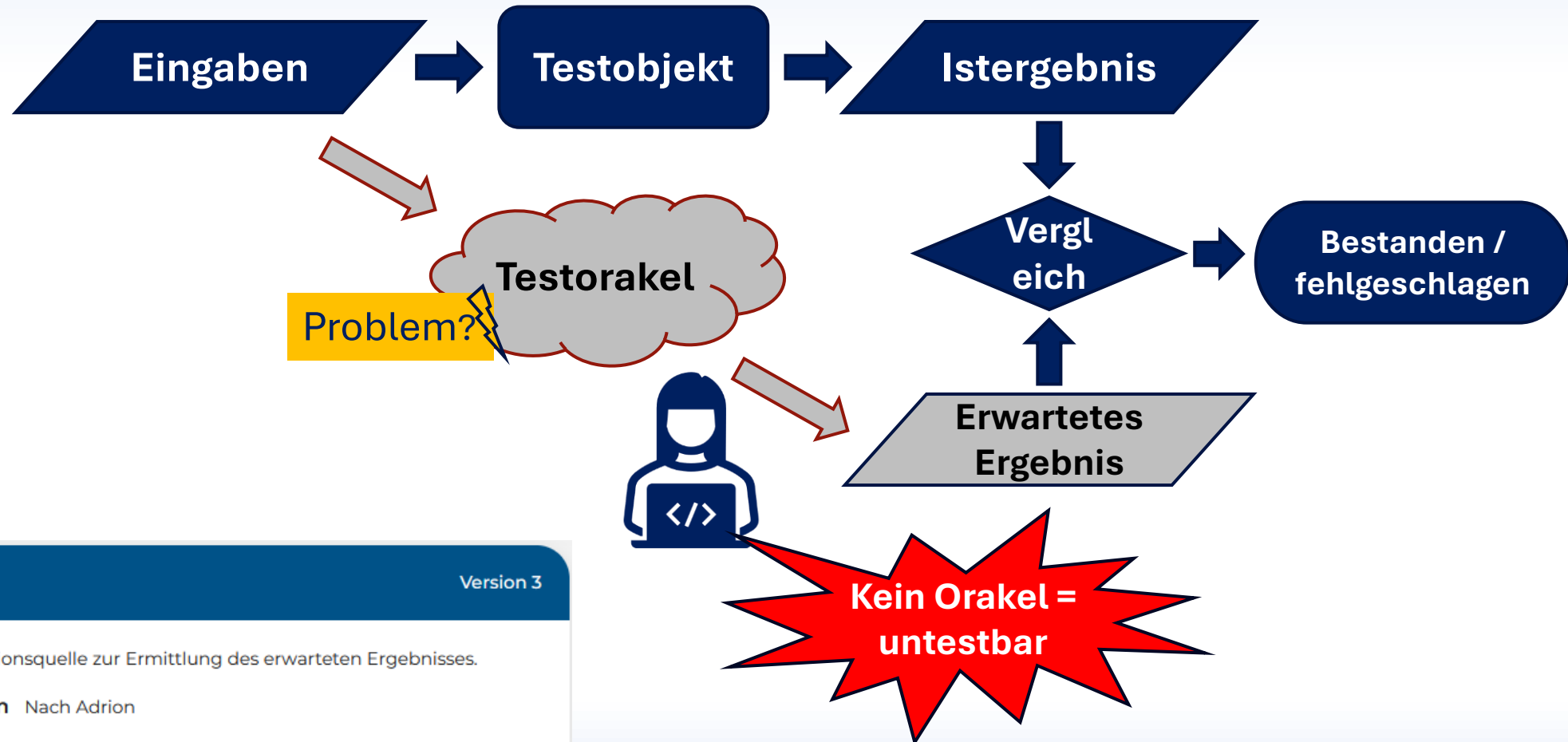


- Aktiver Rentner
 - Bis 2019 Managing Consultant bei Sogeti Deutschland
- GTB-Mitglied
 - Leiter der Arbeitsgruppe Glossar
- Produkt-Owner und Abnahmetester der ISTQB® Glossary App
- Leiter der ISTQB® Task Force „Advanced Level Test Analyst“
 - Aktuell: Entwicklung einer neuen Hauptversion v4.0

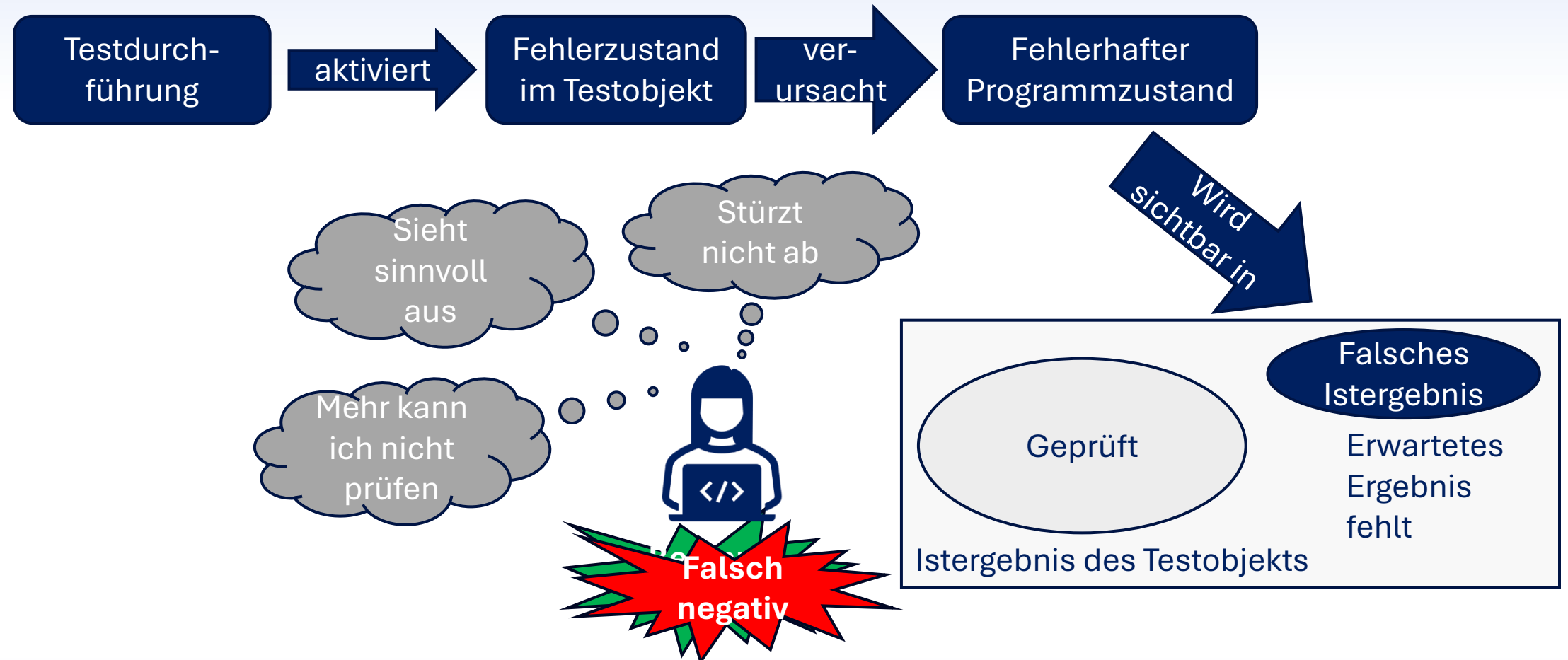


Was ist ein Testorakel?

Dynamisches Testen



Das Ignorieren des Orakelproblems führt zu falsch-negativen Ergebnissen



Nach: Li and Offutt, Test oracle Strategies for Model-Based Testing, 2017

Ist das ein neues Problem?



Eine hellenistische Amphora

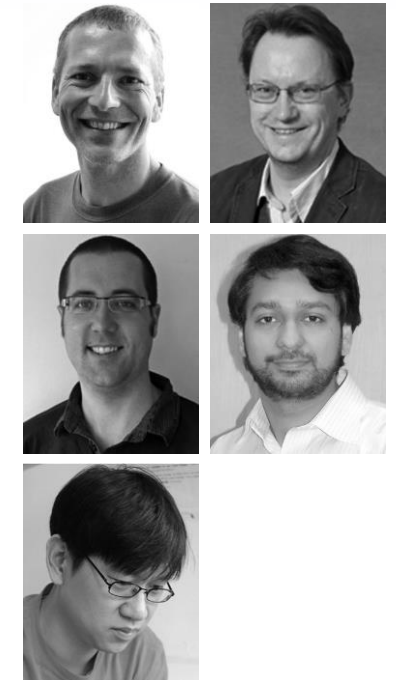


William E. Howden, "Theoretical and empirical studies of program testing", 1978



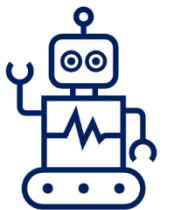
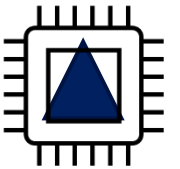
Elaine J. Weyuker, "On testing non-testable programs", 1982.

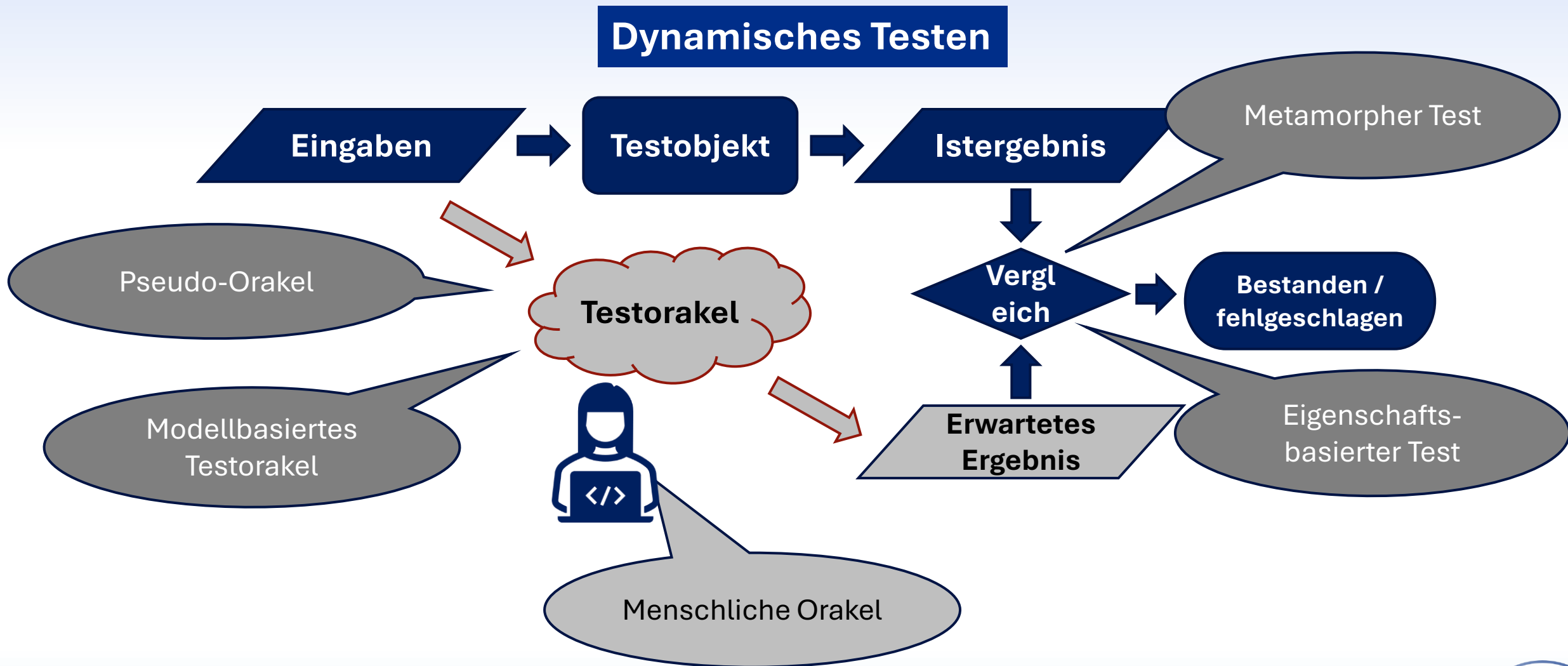
„Die Entwicklung weiterer Lösungen des Orakelproblems ist eine Kernaufgabe von Forschung und Entwicklung.“



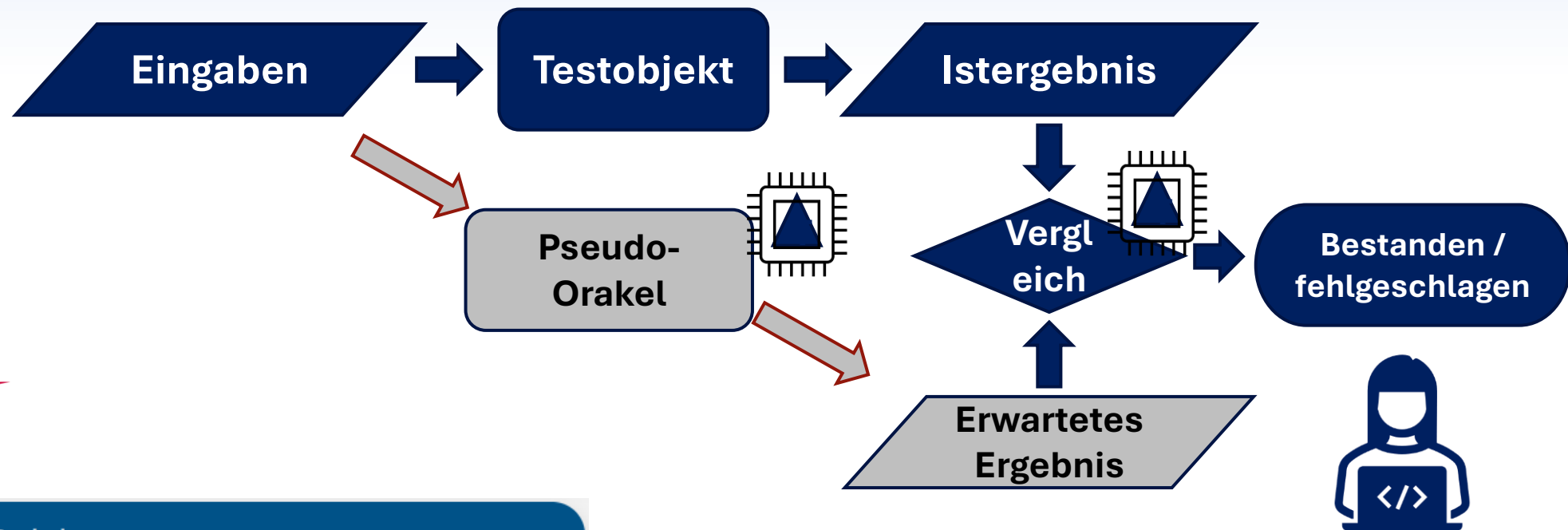
Earl T. Barr et al.: The Oracle Problem in Software Testing: A Survey, 2015

- Beim manuellen Testen erwartet man von den Testern, alle Anomalien zu erkennen
- Vollständig automatisiertes Testen braucht für generierte konkrete Testfälle ein Testorakel
 - Agile Vorgehensweisen und DevOps mit automatisierten Tests
 - Kontinuierliche Integration, kontinuierlicher Test
- KI-basierte und andere neue Systeme: Das erwartete Ergebnis ist oft nichtdeterministisch
 - Nicht vollständig vorhersehbar
 - Bleibt über die Zeit nicht konstant





Was ist ein Pseudo-Orakel?



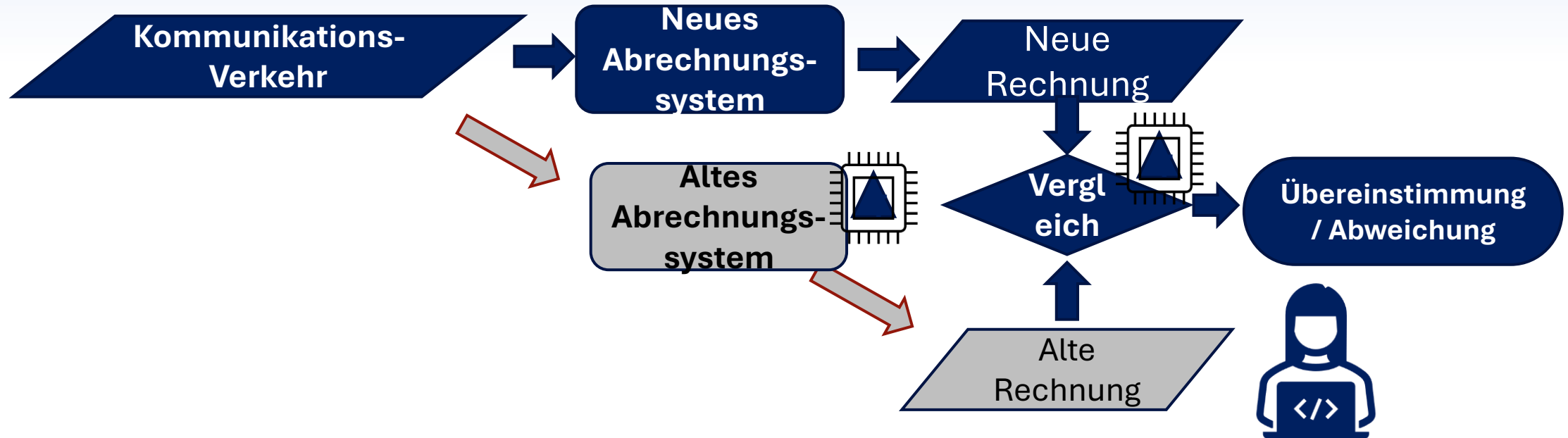
Pseudo-Orakel

Version 1

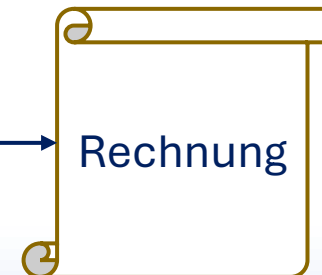
Eine unabhängig abgeleitete Variante des Testobjekts, die zur Erzeugung von Ergebnissen verwendet wird, welche mit den Ergebnissen des ursprünglichen Testobjekts bei gleichen Testeingaben verglichen werden.

Referenzen ISO 29119-11

Softwemigration eines Mobilfunk-Abrechnungssystems



Pierre-Auguste Renoir: Alphonsine Fournaise (1879)



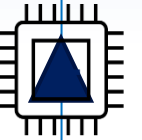
80% Übereinstimmung
10% neue Rechnung falsch
5% alte Rechnung falsch
5% beide Rechnungen falsch

Vor- und Nachteile

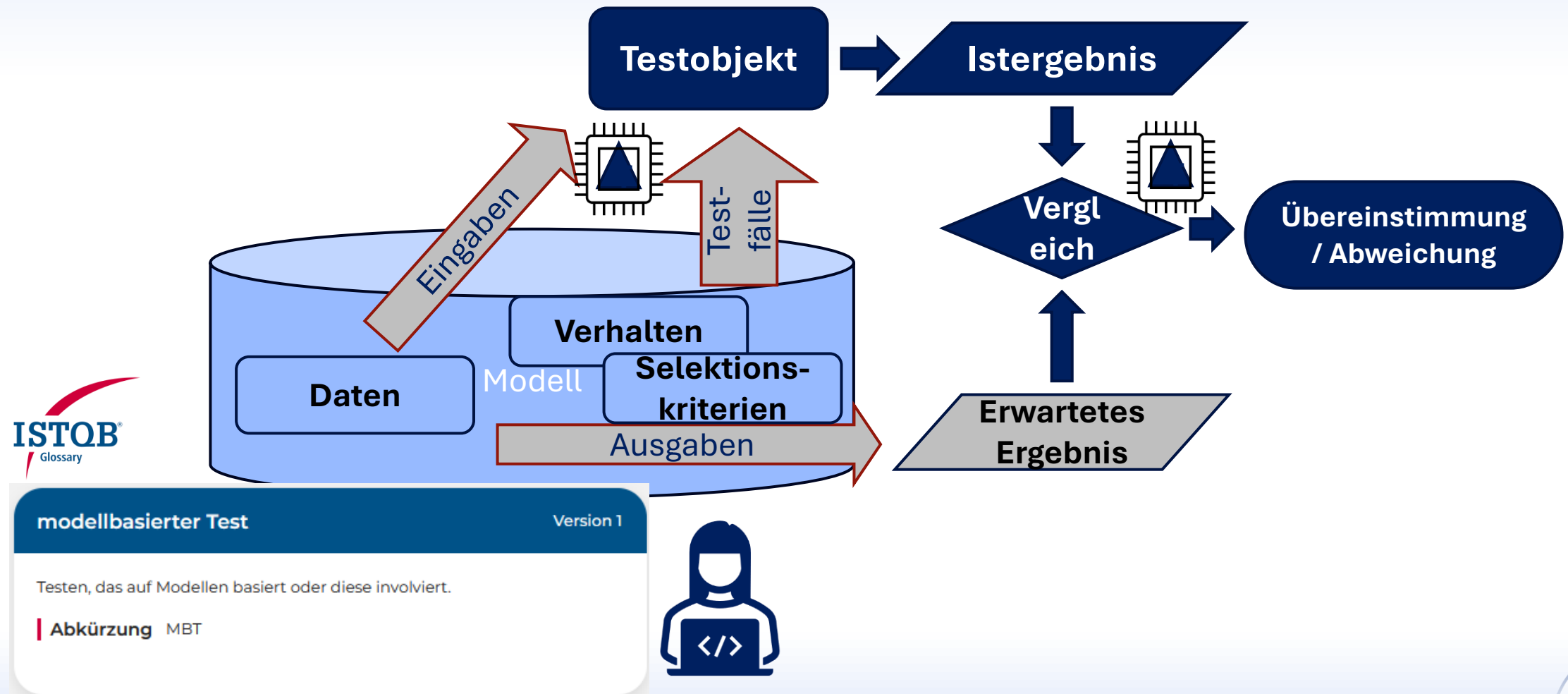
- Vorteile:
 - Die umfassendste Lösung
- Nachteile:
 - Die Entwicklung eines dedizierten Pseudo-Orakels kann aufwändig / teuer sein

Anwendbarkeit

- Bestehendes System als Pseudo-Orakel
 - Softwaremigration
 - Backup-Lösung von ausfallsicheren Systemen
- Entwicklung eines dedizierten Pseudo-Orakels
 - Kernfunktion bei funktionalen Korrektheitstests, z.B. mit Tabellenkalkulation
 - Mehrfachentwicklung bei hochkritischen Anwendungen

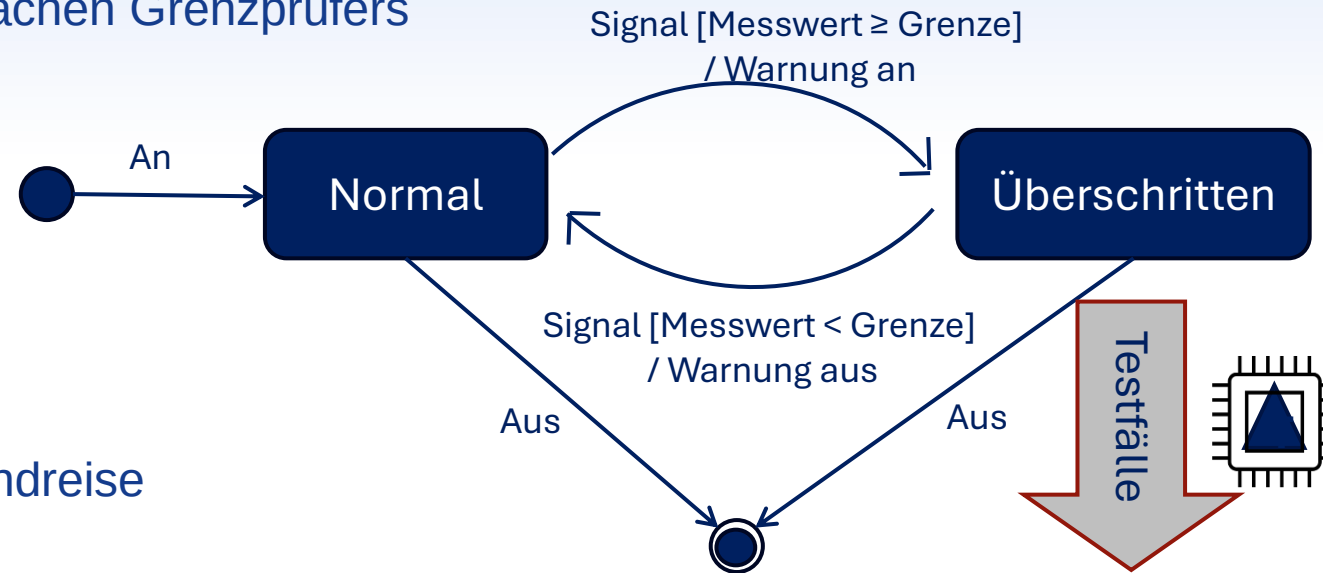


Was ist ein Modellbasiertes Testorakel?



Modellbasiertes Testorakel: Ein Beispiel

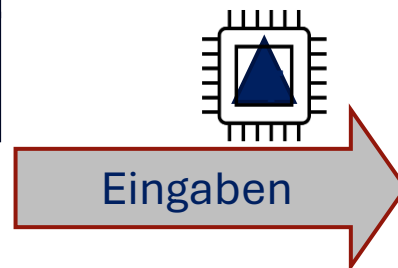
- Zustandsmodell eines einfachen Grenzprüfers
- Verhaltensmodell



- Testselektionskriterien: Rundreise
- Datenmodell

Messwert
-1
0
99
100
101
119

Grenze
0
100



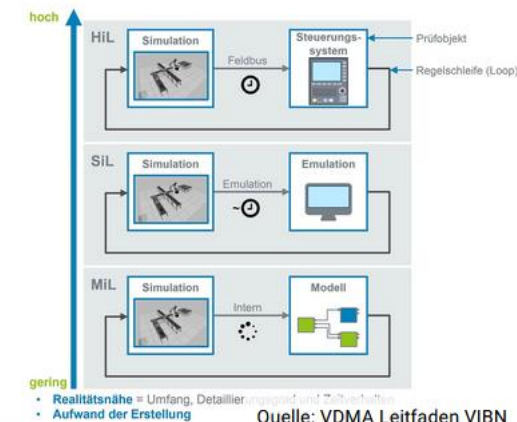
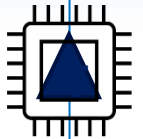
	Grenze	Status	Messwert	Ausgabe	Neuer Zustand
Testfall 1	100	Normal	101	Warnung an	Überschritten
	100	Normal	99	Warnung aus	Normal
Testfall 2	100	Normal	99	-	Normal
Testfall 3	100	Überschritten	119	-	Überschritten

Vor- und Nachteile

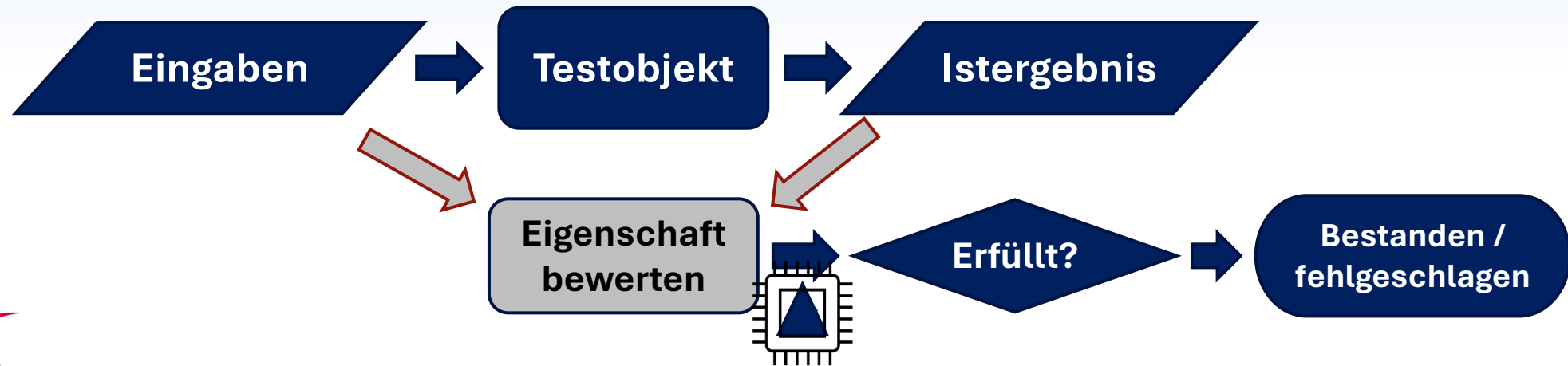
- Vorteile:
 - Echtes automatisiertes Testorakel
- Nachteile:
 - Erfordert ein ausführliches Modell der Spezifikation
 - Erstellung und Pflege eines dedizierten Modells ist aufwändig

Anwendbarkeit

- Bei verfügbaren MBT-Modellen, die einen Teil des Orakels enthalten
 - Zustandsabhängige Modelle, z.B. erweiterte endliche Automaten (EFSM)
 - Zustandslose Verhaltensmodelle, z.B. Entscheidungstabellen
- Eingebettete Systeme
 - Model-In-the-Loop (MIL) als Orakel für die Software-In-the-Loop (SIL)



Was ist ein eigenschaftsbasierter Test?



eigenschaftsbasierter Test

Version 1

Ein Testansatz, bei dem die Testergebnisse anhand festgelegter Beziehungen zwischen Eingaben und Ausgaben eines Testfalls verifiziert werden.

Siehe auch

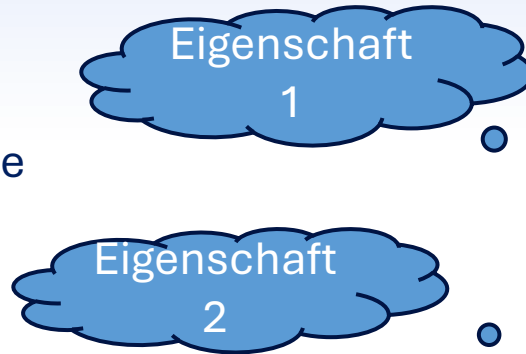
metamorpher Test

Eigenschaftsbasierter Test: Ein Beispiel

Abstrakter Testfall:
Einfache Suche nach einer Zeichenkette

Eigenschaft 1:
Angezeigte Trefferzahl == Anzahl der
Ergebniszeilen

Eigenschaft 2:
Die gesuchte Zeichenkette ist in jeder
Trefferzeile enthalten

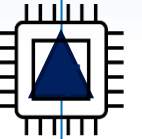


Vor- und Nachteile

- Vorteile:
 - Passt zur automatisierung der Testausführung
 - Mit Zufallstest gut kombinierbar
- Nachteile:
 - Löst das Orakelproblem nur partiell
 - Bietet keine überdeckungsbasierten Endekriterien

Anwendbarkeit

- Anwendungen mit automatisiert prüfbaren Kerneigenschaften
- Anwendungen mit hohen Risiken bezüglich gewisser Eigenschaften
 - Kompensieren die teilweise Verifizierung durch zahlreiche konkrete Testfälle
 - In Kombination mit einem Zufallsgenerator für konkrete Eingaben



Was ist ein metamorpher Test?



metamorphe Relation

Version 1

Eine Beschreibung, wie eine Änderung der Eingaben vom Ausgangstestfall zum Folgetestfall eine Änderung der erwarteten Ergebnisse vom Ausgangstestfall zum Folgetestfall beeinflusst.

Abkürzung MR

Referenzen ISO 29119-11

metamorpher Test

Version 1

Ein Testverfahren, bei dem die Eingaben und erwarteten Ergebnisse aus einem bestandenen Testfall unter Verwendung einer metamorphen Relation extrapoliert werden.

Abkürzung MT

Bestimme metamorphe Relationen

Entwerfe Ausgangstestfälle

Entwerfe Nachfolgetestfälle

Führe den Test aus und verifiziere die metamorphen Relationen

Metamorpher Test: Ein Beispiel

- Autonome Fahrzeugsteuerungssysteme
 - Basieren auf tiefen neuronalen Netzen
 - Haben den klassischen Systemtest bestanden
- Orakelproblem
 - Was ist das korrekte Fahrverhalten in der gegebenen Situation?



Ausgangstestfall

Regen zugefügt

Quelle: deeplearningtest.github.io/deepTest/

- Metamorphe Relationen
 - Ausgangstestfall: Lenkungsaufgabe unter guten Sichtbedingungen
 - Folgetestfälle: Die gleiche Lenkungsaufgabe unter geänderten Sichtbedingungen: Regen, Nebel, Unschärfe, Bildgröße, Kontrast, Bildverschiebung, Rotation, Helligkeit, usw.
- Testergebnisse
 - Hunderte von Fehlschlägen, viele davon können zum fatalen Unfall führen

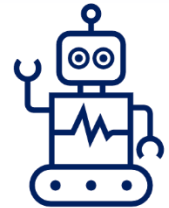
Tian Y., Pei K., Jana S., Ray B.: DeepTest: Automated Testing of Deep-Neural-Network-driven Autonomous Cars. ACM/IEEE 40th ICSE, 2018

Vor- und Nachteile

- Vorteile:
 - Einsetzbar bei harten Orakelproblemen
 - Mit Zufallstest und MBT gut kombinierbar
- Nachteile:
 - Löst das Orakelproblem nur partiell
 - Bietet keine überdeckungsbasierten Endekriterien
- Die Analyse der metamorphen Relationen ist ein entscheidender Erfolgsfaktor

Anwendbarkeit

- KI-basierte Systeme
 - Autonomes Fahren
 - Flugsteuerungssysteme
 - Empfehlungssysteme
- Suchmaschinen
- Wissenschaftliche Anwendungen



$$E = mc^2$$

- Behalten Sie Orakelprobleme und ihren Einfluss auf die Testbarkeit im Auge
- Machen Sie sich mit den bekannten Lösungsansätzen vertraut
- Setzen Sie die geeigneten Lösungen ein
- Kombinieren Sie verschiedene Lösungen wenn möglich
 - Z.B.: Modellbasierter Test und metamorpher Test



Gerne beantworte ich Ihre Fragen

DANKE FÜR IHRE AUFMERKSAMKEIT

Powered by

German Testing Board e. V.

Vorsitzender

Florian Fieber

Stellvertretende Vorsitzende

Dietrich Leimsner

Dr. Klaudia Dussa-Zieger

Geschäftsführer

Dr. Armin Metzger

Backoffice

Andrea Kränzlein

OFFICE-MANAGEMENT

Koldestrasse 8 b

91052 Erlangen

GERMANY

Tel.: +49 (0) 91 31 / 97 61 06

Mobil: +49 (0) 171 / 63 90 749

E-mail: backoffice@gtb.de