

# Software Engineering

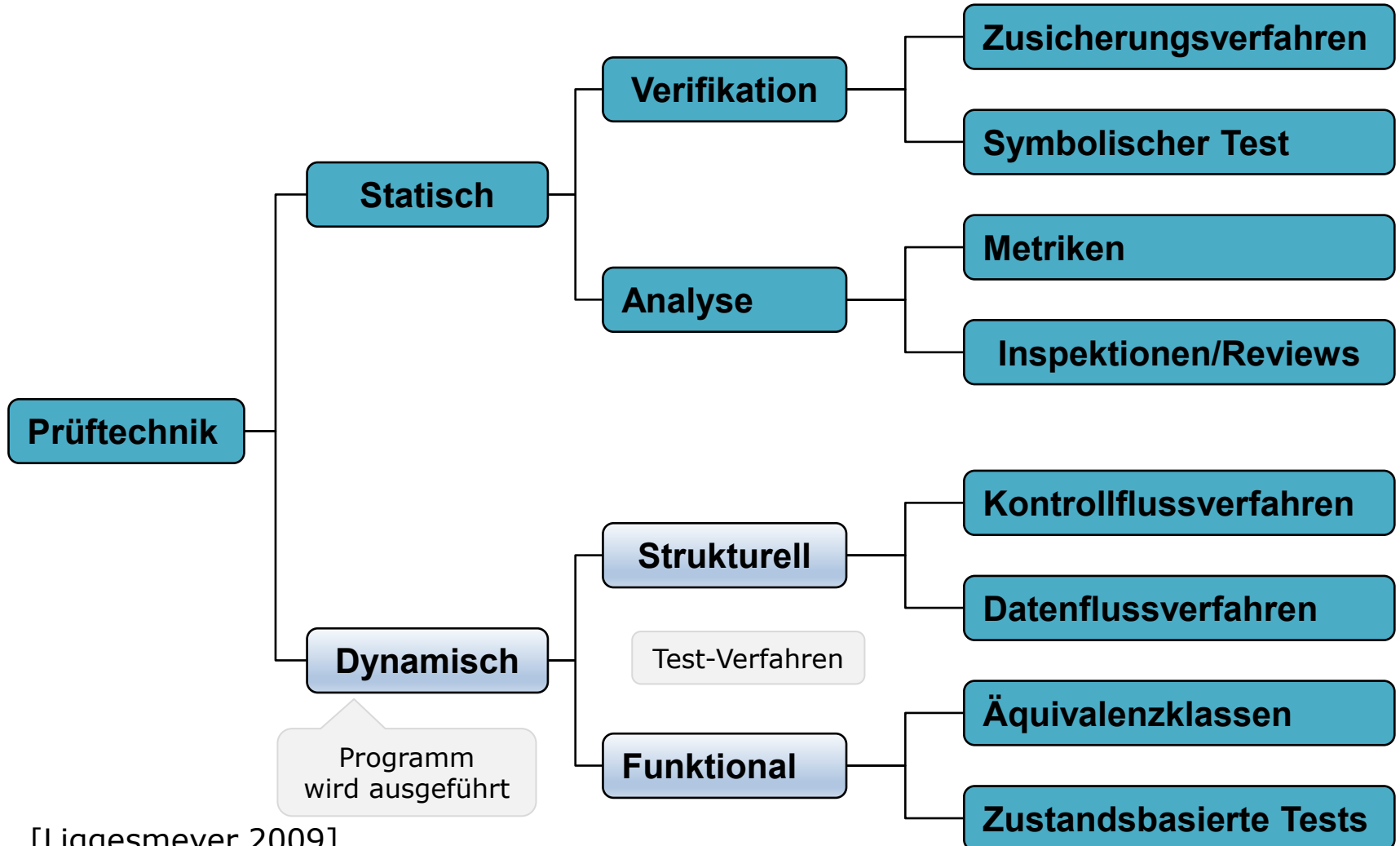
Test 1 / 2

Prof. Dr. Bodo Kraft



# Übersicht der Prüftechniken

## Verfahren der QS im Detail



[Liggesmeyer 2009]

# Klassifikation von Tests

## Dynamische Prüfverfahren

### Abstraktion

#### 1. Black-Box-Test (funktional)

- Ableitung von Testmengen aus der Spezifikation
- Programmstruktur verborgen

#### 2. White-Box-Test (strukturell)

- Ableitung von Testmengen aus der Programmstruktur

### Granularität

#### 1. Testen im Kleinen

- Testen einzelner Komponenten

#### 2. Testen im Großen

- Testen oberhalb von Komponenten (Teilsysteme oder Systeme)

### Zeit

#### 1. Komponententest (Unit-Test)

- Wird nach der Implementierung einer Komponente durchgeführt

#### 2. Integrationstest

- Wird nach den Komponententests durchgeführt, um das Zusammenspiel zu testen

#### 3. Systemtest

- Wird vom Auftragnehmer vor dem Abnahmetest durchgeführt

#### 4. Abnahmetest

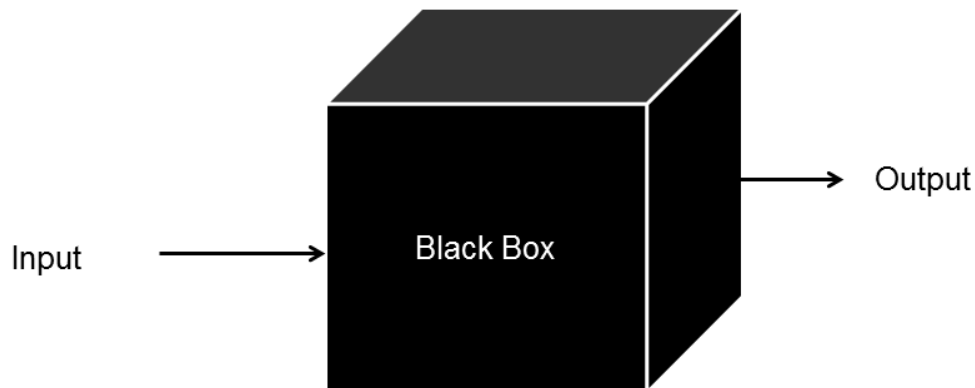
- Wird vom Kunden bei der Auslieferung durchgeführt

# Der Black-Box-Test

## Dynamisch, funktionale Prüfverfahren

### Charakteristisch:

- Das Programm wird als „schwarzer Kasten“ betrachtet
- Keine Kenntnis von der Programmstruktur
- Nur die Spezifikation bekannt, gegen die wird getestet



*Internal behavior of the code is unknown*

# Äquivalenzklassenbildung

## Dynamisch, funktionale Prüfverfahren

### Vorgehen:

1. Mögliche Eingaben in Bereiche aufteilen, sogenannte **Klassen**
  - Alle Elemente einer Klassen zeigen identisches Verhalten
2. Zulässige und unzulässige Klassen trennen
3. Für jede Klasse mindestens einen **Repräsentanten** wählen für die Testdaten
4. Vereinigung der Äquivalenzklassen sollte der gesamte Wertebereich sein.

### Beispiel:

Der Parameter einer Methode soll zwischen zwei Grenzen liegen:  $1 \leq \text{tag} \leq 31$

Äquivalenzklasse	Typ	Repräsentant
$\text{Tag} < 1$	Unzulässig	0, -4
$1 \leq \text{Tag} \leq 31$	Zulässig	10
$31 < \text{Tag}$	Unzulässig	100



# Äquivalenzklassenbildung Beispiel

## Dynamisch, funktionale Prüfverfahren

Eine Prozedur bekommt als Parameter drei Zahlen, die als Seitenlängen eines Dreiecks interpretiert werden. Sie soll bestimmen, ob das Dreieck gleichschenkelig und/oder gleichseitig ist:

Gesamter Wertebereich:  $\mathbb{IN} \times \mathbb{IN} \times \mathbb{IN}$   
 Beispiel-ÄK:  $\{(a,b,c) \in \mathbb{IN}^3 \mid a + b > c\}$

### Aufgabe:

Finden Sie geeignete Repräsentanten für die gegebenen Äquivalenzklassen

Äquivalenzklasse		Repräsentant		
		a	b	c
1) $a + b > c \geq 0$	(zulässig, —)			
2) $a = b, a \neq c, 1)$	(zulässig, gleichschenkelig)			
3) $a = c, a \neq b, 1)$	(zulässig, gleichschenkelig)			
4) $b = c, a \neq b, 1)$	(zulässig, gleichschenkelig)			
5) $a=b=c$ und 1)	(zulässig, gleichseitig)			
6) $a + b = c$	(unzulässig, entartetes D.)			
7) $a + b < c$	(unzulässig, kein Dreieck)			

- Häufig **Verfeinerung** des Äquivalenzklassentests
- Dazu Repräsentanten **am Rand** der Äquivalenzklassen wählen

### Beispiel Monatstag:

ÄK	Äquivalenzklasse	Typ	Repräsentant
1	Tag < 1	Unzulässig	0, -4
2	$1 \leq \text{Tag} \leq 31$	Zulässig	10
3	$31 < \text{Tag}$	Unzulässig	100

ÄK	Typ	Randwert
1	Unzulässig	0
2	Zulässig	1
2	Zulässig	2
2	Zulässig	30
2	Zulässig	31
3	Unzulässig	32

- **Leere/fehlende Eingaben**
  - zum Beispiel null
  - oder fehlende Kommandozeilenparameter
- **Randelemente**
  - erstes und letztes Element einer sequentiellen Datenstruktur (Liste, Datei, Array, Tabelle, ...)
- **Extremwerte**
  - Minimal zulässige Zahlenwerte
  - Maximal zulässige Zahlenwerte (z.Bsp.: `INT_MAX`)
- **“Ungünstige” Eingaben**
  - Sortierfunktion bekommt umgekehrt sortiertes Feld
  - Suchfunktion findet Element erst am Ende / gar nicht, ...