

Verlässliche, adaptive Informationssysteme

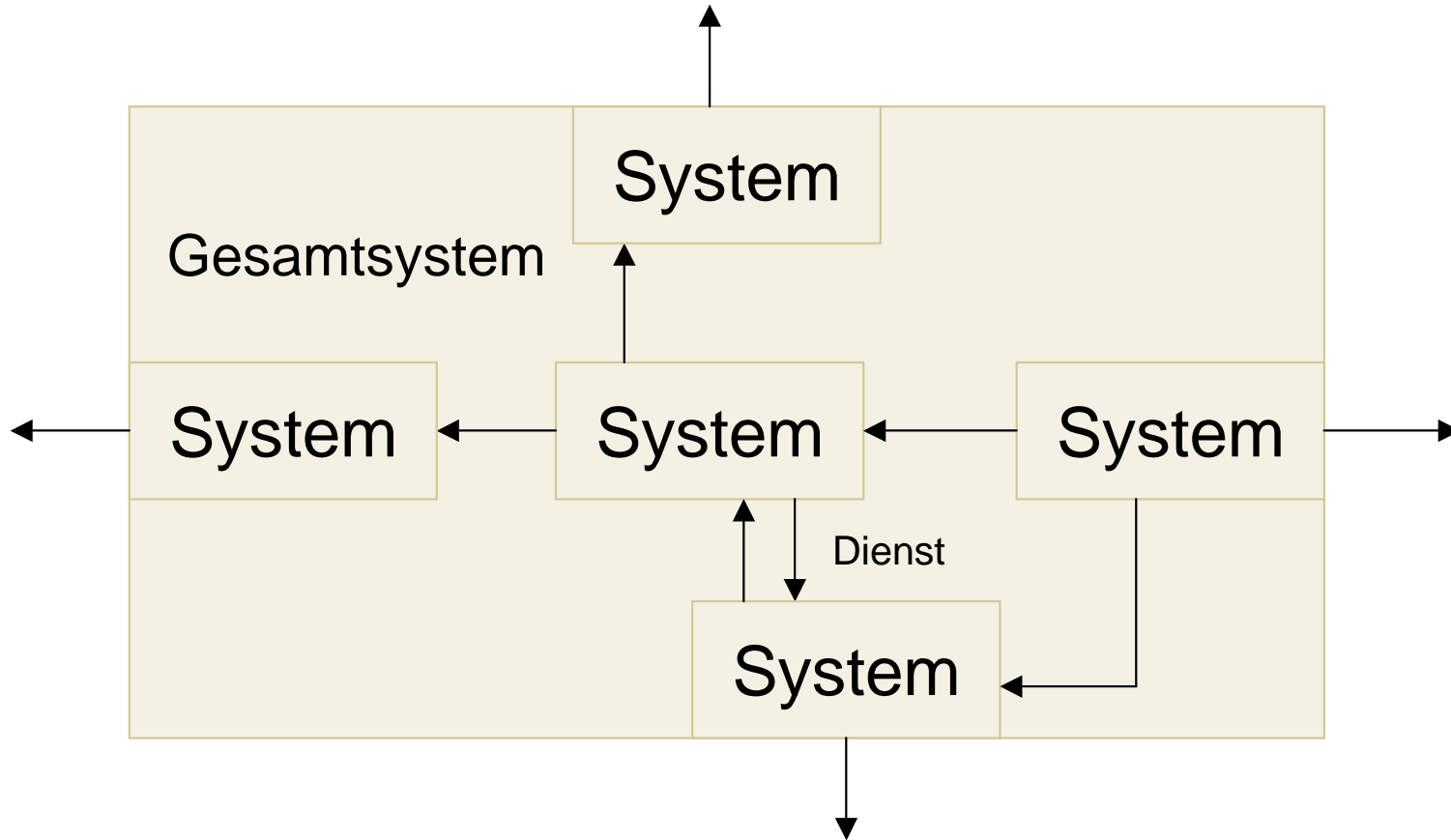
Thema: Zuverlässigkeit und QoS

Markus Edinger

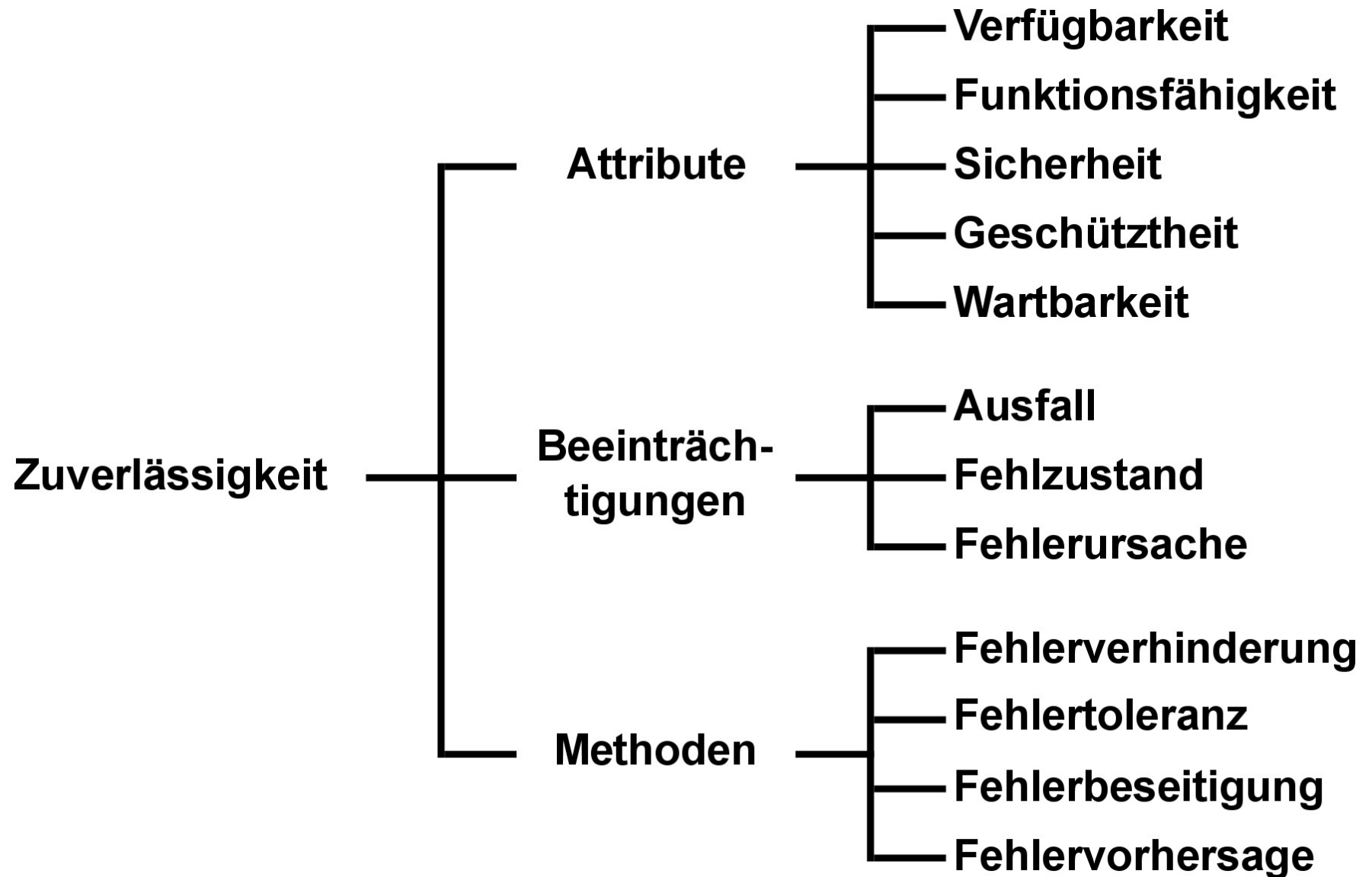
Vortragsgliederung

- § Zuverlässigkeit
- § Fehlerklassen
- § QoS in verteilten Systemen
- § Dependability Benchmarking

Informationssystem



Zuverlässigkeit



Attribute der Zuverlässigkeit

- § Verfügbarkeit (availability)
„Bereitsein zum Gebrauch“
- § Funktionsfähigkeit (reliability)
„Kontinuität der Leistung“
- § Sicherheit (safety)
„Vermeidung katastrophaler Folgen für die Umwelt“
- § Geschütztheit (security)
„Verhindern von nichtauthorisiertem Zugriff auf Informationen“
- § Wartbarkeit (maintainability)
„Möglichkeit, das System Änderungen zu unterziehen“

Beeinträchtigungen der Zuverlässigkeit

1. Ausfall

- § *Ausfall*: Dienst erfüllt nicht mehr seine Funktion
- § *Dienstausfall*: Zeit in der das System einen inkorrekten Dienst liefert
- § *Charakterisierung*:
 - § Art des Ausfalls: Inhalt oder Zeitpunkt
 - § Erkennbarkeit
 - § Konsistenz
 - § Konsequenz

Beeinträchtigungen der Zuverlässigkeit

2. Fehlzustand

- § *Fehlzustand*: Teil des Systemzustands, der für einen Ausfall verantwortlich ist
- § *Latenter Fehlzustand*: unentdeckter Fehlzustand
- § Ein Fehlzustand führt nur zu einem Ausfall, wenn er Teil des externen Systemzustandes ist.

Beeinträchtigungen der Zuverlässigkeit

3. Fehlerursache

- § *Fehlerursache*: verantwortliche oder hypothetische *Ursache für einen Fehlzustand*.
- § *Schlafende Fehlerursache*: eine Fehlerursache, die keinen Fehlzustand hervorruft
- § drei Hauptkriterien: *Art, Ursprung* und *Dauer* der Fehlerursache.
 - § **Art der Fehlerursache**
 - § Unvermögen:
Versehen oder *Inkompetenz*
 - § Vorsatz:
unbewusst oder *bewusst*
 - § Absicht:
bösartig oder *nicht böartig*

Fehlerursache

§ Ursprung der Fehlerursache

§ Phänomenologische Gründe:

menschlicher oder *natürlicher* Ursprung

§ Systemgrenzen:

intern oder *extern*

§ Phase der Entstehung:

während der *Entwicklung* oder der während der *Benutzung*

§ Dimension:

Hardware- oder *Softwarefehler*

§ Dauer der Fehlerursache

dauerhaft (*permanent*) oder zeitlich begrenzt (*transient*)

Fehlerklassen

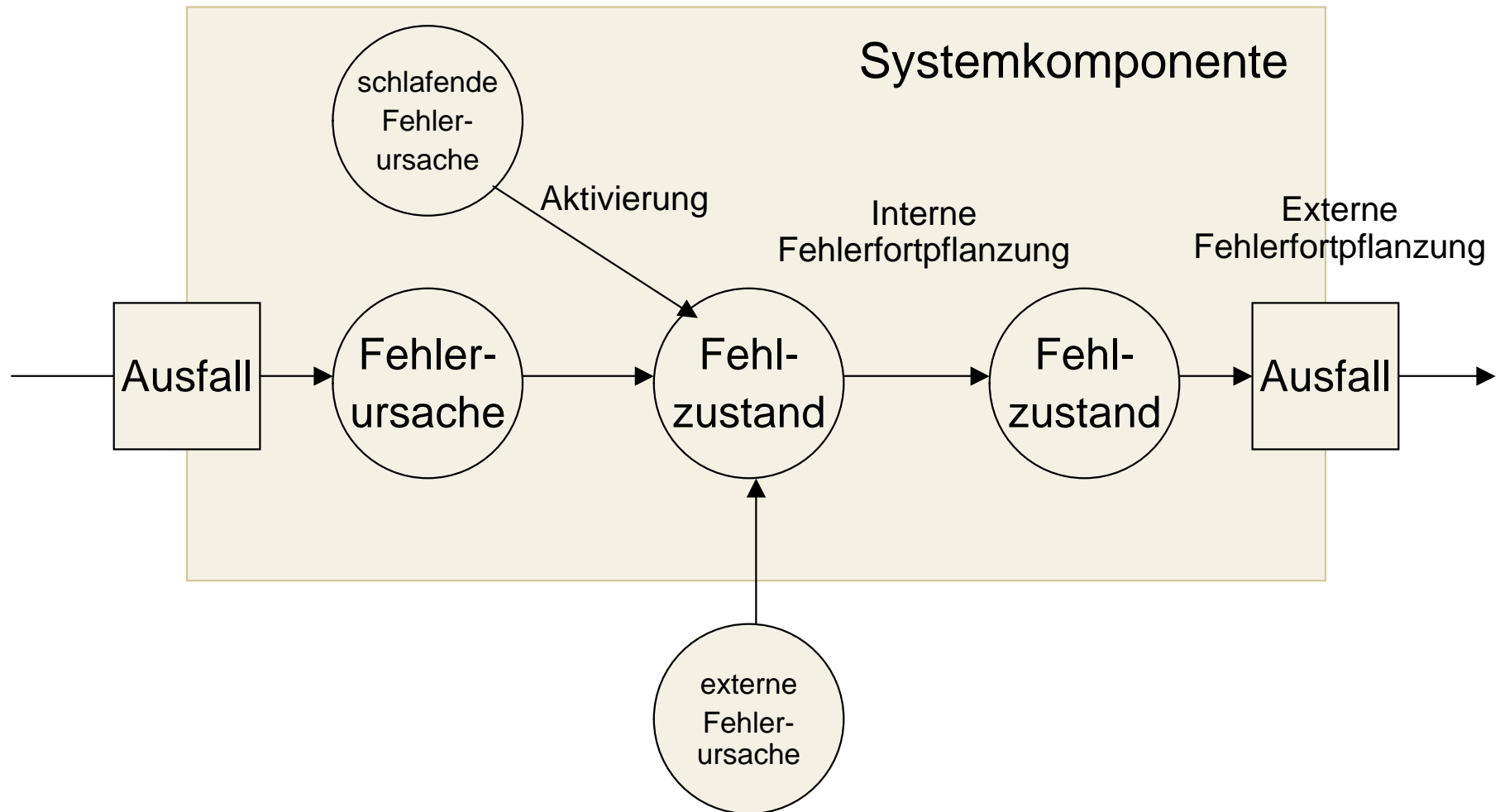
Fehlerklasse	Ursprung								Art						Dauer		
	Phase der Entstehung		Systemgrenzen		Phänomenologische Ursache		Dimension		Absicht		Voratz		Unvermögen				
	Entwicklungsfehler	Benutzungsfehler	Interne Fehler	Externe Fehler	Physikalische Fehler	Menschliche Fehler	Hardwarefehler	Softwarefehler	Nicht böswertig	böswertig	unbeabsichtigt	beabsichtigt	Versehen	Inkompetenz	permanente Fehler	transiente Fehler	
1	X		X					X		X			X		X		
2	X		X					X		X				X	X		
3	X		X					X		X			X	X	X		
4	X		X					X		X				X	X		
5	X		X					X			X			X	X		
6	X		X				X		X		X			X	X		
7	X		X				X		X		X		X		X		
8	X		X				X		X		X			X	X		
9	X		X				X		X			X		X	X		
10	X		X				X		X			X		X	X		
11	X		X			X		X		X		X		X	X		
12		X	X			X		X		X		X		X		X	
13		X	X			X		X		X		X		X			X
14		X		X		X		X		X		X		X	X		X
15		X		X		X		X		X		X		X			X
16		X		X		X		X		X		X		X	X		X
17		X		X		X		X		X		X		X	X		X
18		X		X		X		X		X		X		X	X		X
19		X		X		X		X		X		X		X	X		X
20		X		X		X		X		X		X		X	X		X
21		X		X		X		X		X		X		X	X		X
22		X		X		X		X			X			X	X		X
23		X		X		X		X			X			X	X		X
24		X		X		X		X			X			X	X		X
25		X		X		X		X			X			X	X		X
26		X		X		X		X		X		X		X	X		X
27		X		X		X		X		X		X		X	X		X
28		X		X		X		X		X		X		X	X		X
29		X		X		X		X		X		X		X	X		X
30		X		X		X		X		X		X		X	X		X
31		X		X		X		X		X		X		X	X		X

Entwicklungsfehler

Physikalische Fehler

Eingabefehler

Pathologie von Ausfall, Fehlerursache und Fehlzustand



Sicherstellen der Zuverlässigkeit

§ Zuverlässigkeitsverfahren

- § Fehlerverhinderung

- § Fehlertoleranz

§ Zuverlässigkeitsvalidation

- § Fehlerbeseitigung

- § Fehlervorhersage

Fehlertoleranz

§ Fehlererkennung

§ nebenläufig oder präventiv

§ Systemwiederherstellung

§ Fehlzustandsbehandlung

§ Rollback

§ Rollforward

§ Fehlermaskierung

§ Fehlerursachenbehandlung

§ Diagnose

§ Isolation

§ Rekonfiguration

§ Reinitialisierung

Fehlerbeseitigung

§ Während der Entwicklung

- § Verifikation
 - § statisch, dynamisch
- § Diagnose
- § Korrektur

§ Während des Betriebs

- § korrigierend
- § präventiv

Fehlervorhersage

- § Abschätzen der Zahl und Schwere der im System vorhandenen Fehlerursachen
- § Vorhersagen über das Auftreten von Ausfällen und deren Folgen

Quality of Service in verteilten Systemen

Definitionen:

§ Siemens Online Lexikon:

„Dienstgüte, Quality of Service (QoS):

Unter Dienstgüte versteht man alle Verfahren, die den Datenfluss in LANs und WANs so beeinflussen, dass der Dienst mit einer festgelegten Qualität beim Empfänger ankommt...“

§ International Telecommunication Union (ITU):

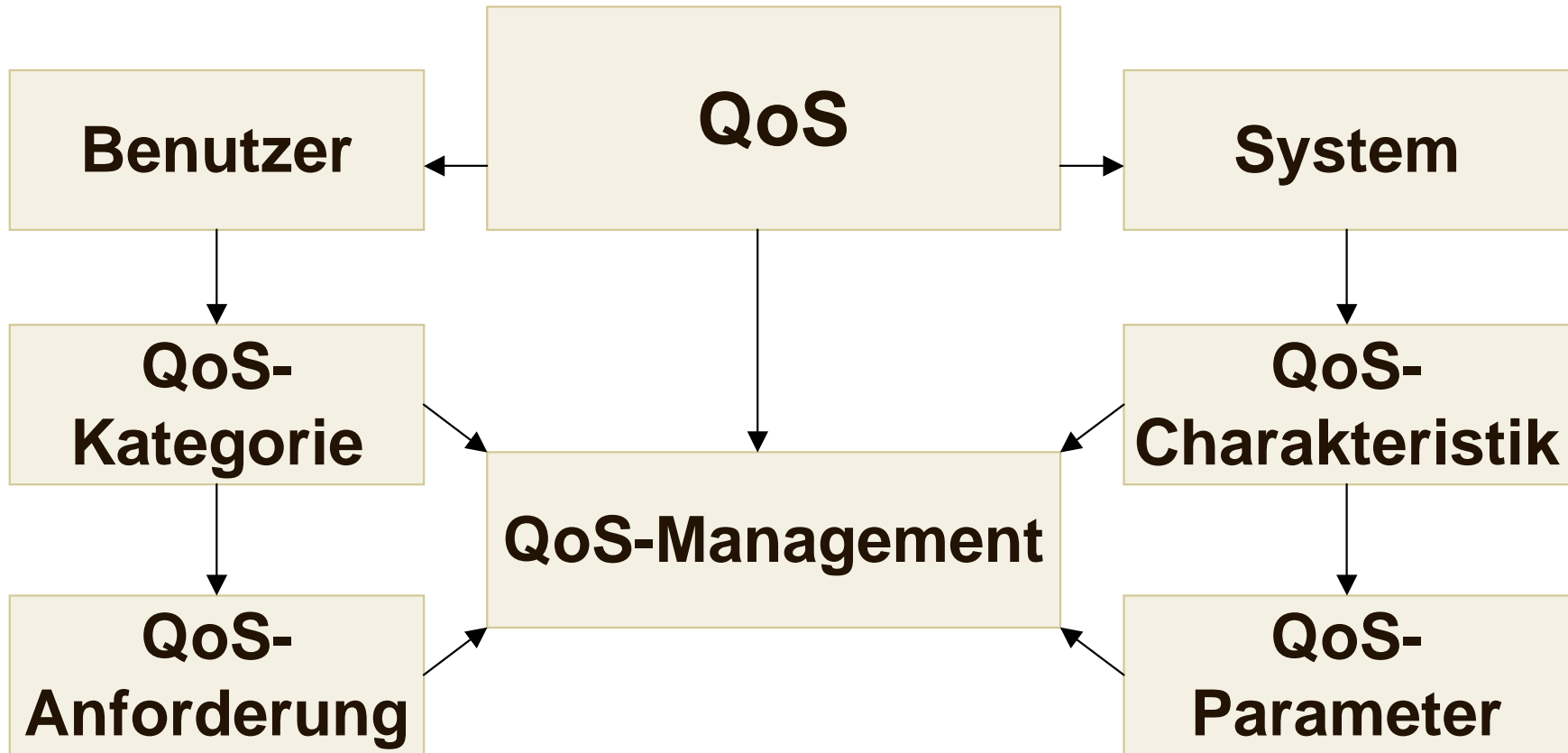
„Quality of Service (QoS):

A set of qualities related to the collective behaviour of one or more objects.“

Anforderungen an ein QoS-System:

- § Umgang mit Heterogenität
- § Adaptivität
- § Robustheit

Begriffsbildung



Dimensionen von QoS

§ **Statische bzw. dynamische QoS:**

- § *statische QoS*: Festlegen der Leistungsansprüche zur Entwicklungszeit
- § *dynamische QoS*: Erfüllung der Anforderungen während des Betriebs

§ **Ressourcenmanagement und Anwendungsanpassung:**

- § *Ressourcenmanagement*: dynamische Erfüllung der QoS-Parameter
- § *Anwendungsanpassung*: Anwendungen werden an die Ressourcen angepasst

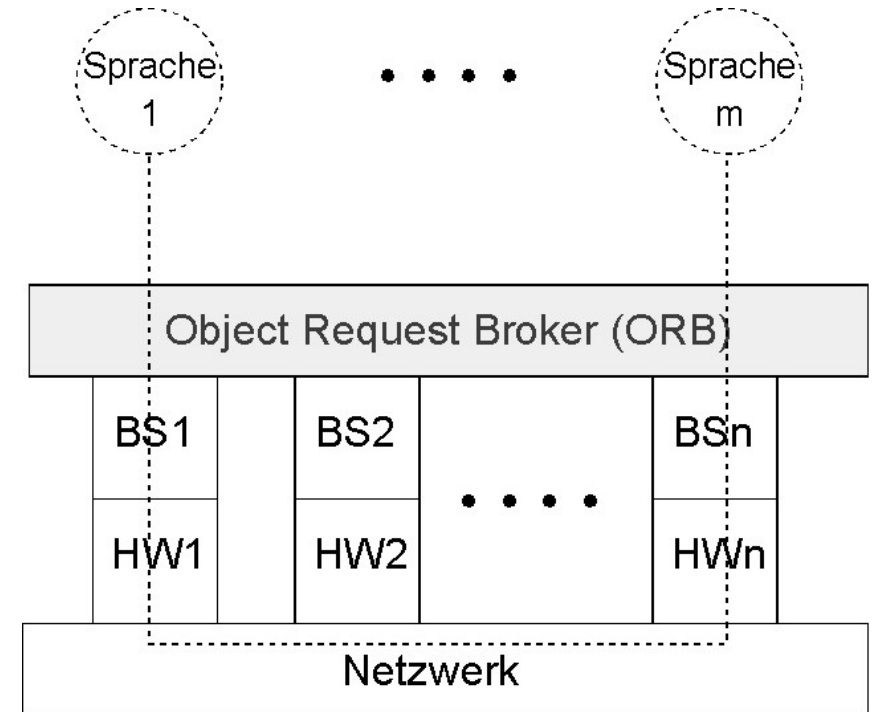
§ **Verhandlung und Abrechnung:**

- § *Bepreisung von Ressourcen*: Verhindert Überlastung

Elemente einer QoS-Architektur

§ Middleware:

- § *Object Request Broker (ORB)*
- § *Interface Definition Language (IDL)*
- § Die QoS-Integration auf allen Ebenen des Systems



§ QoS-Integration während der Entwicklung:

- § *Erweitern der IDL*
- § *Einführen einer QoS Definition Language (QDL)*

QoS-Integration während der Laufzeit

§ QoS-Bereitstellung

- § QoS-Mapping
- § Verfügbarkeitsüberprüfung
- § Ressourcenverwaltung

§ QoS-Management:

- § QoS-Überwachung
- § QoS-Anpassung
 - § QoS-Degradierung
 - § Erneutes Ressourcen-Management

Dependability Benchmarking

§ Aufgaben

- § Untersuchen eines Systems hinsichtlich seiner Zuverlässigkeitsattribute
- § Vergleich mehrerer Systeme

§ Benchmarkspezifikation:

§ Kategorisierungsklassifikation

- § Spezifikation von Anwendungsbereich, Ausführungsumgebung, Lebensphase des Systems, Benchmarkbenutzer und Zweck der Benchmarkerhebung

§ Maßeinheitenspezifikation

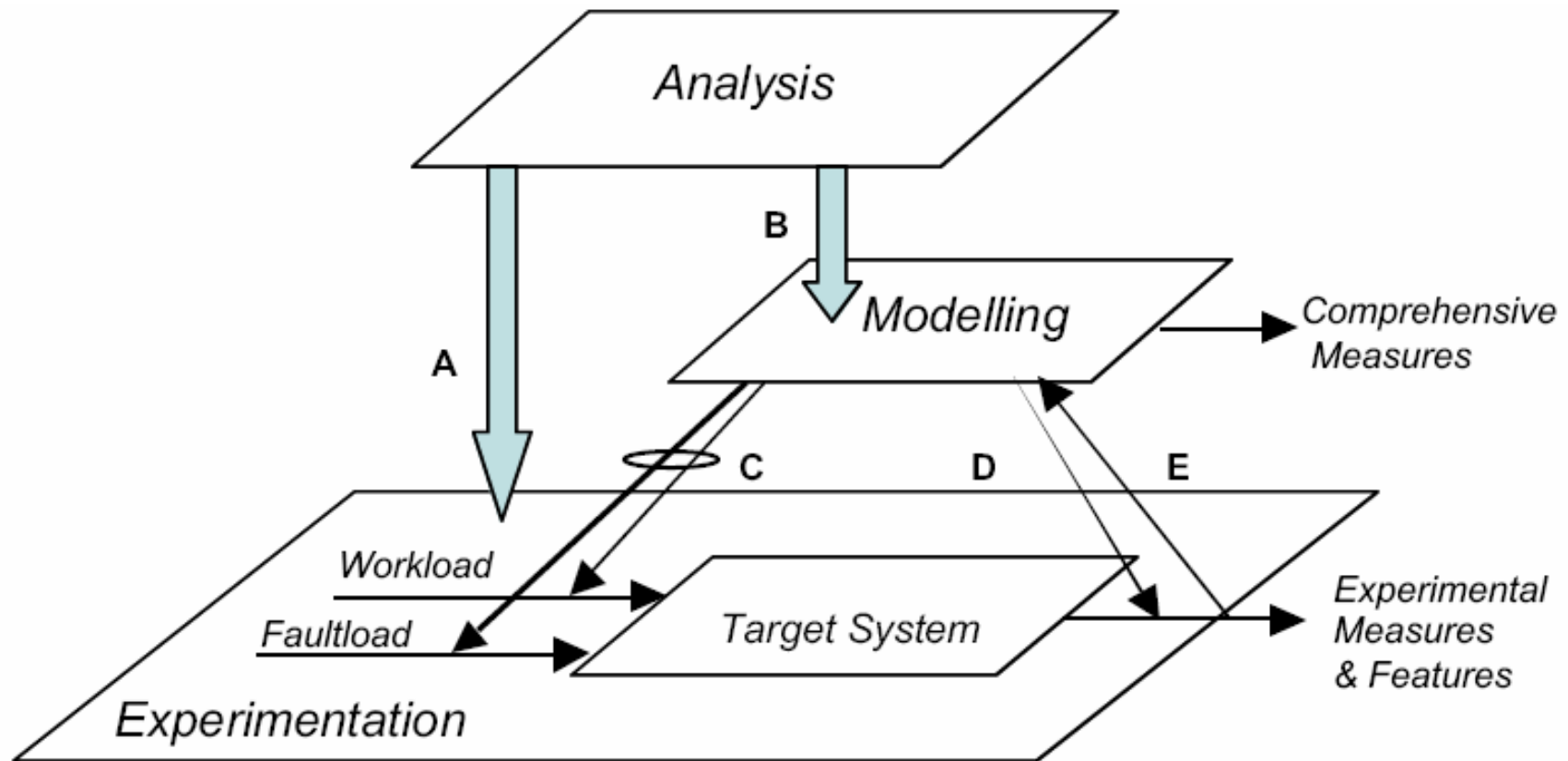
- § globale Maßeinheiten (Gesamtergebnis für ein Attribut)
- § spezifische Maßeinheiten (Maßeinheiten für bestimmte Tests)

§ Testumgebung

- § „System Under Benchmarking“ (Hard- und Softwarekonfiguration)
- § Workload
- § Faultload

Benchmarkdurchführung

§ Szenario



Benchmarkdurchführung

§ **Benchmark-Management-System**

- § Interaktion zwischen SUB und Benchmarking-Target
- § Kontrollieren und Automatisieren des Benchmarking-Prozesses

§ **Prozeduren und Regeln:**

- § Übersetzung des Work- und Faultloads
- § Skalierungsregeln
- § Regeln für die Messungen
- § Formeln und Fehlerberechnungen
- § Offenlegung der Systemkonfiguration
- § Verhindern, dass Entwickler ihr System speziell für die Benchmark optimieren.

Eigenschaften der Benchmark

§ Repräsentativität

§ Maßeinheiten müssen die zu untersuchende Systemeigenschaft widerspiegeln. Workload und Faultload müssen realistisch sein.

§ Wiederholbarkeit und Reproduzierbarkeit:

§ Eine Mehrfachausführung der Benchmark muss statistisch vergleichbare Werte liefern.

§ Übertragbarkeit:

§ Die Spezifikation muss allgemein gehalten werden um die Benchmark auf andere Systeme übertragen zu können.

§ Eindringung:

§ Die Benchmark soll so wenig wie möglich in das System eingreifen.

§ Dauer und Kosten:

§ Kosten dürfen den Nutzen nicht übersteigen.

Zusammenfassung

