

Projekte. Beratung. Spezialisten.

Qualität als Treiber: *Wie Qualitätsanforderungen die Architektur steuern*

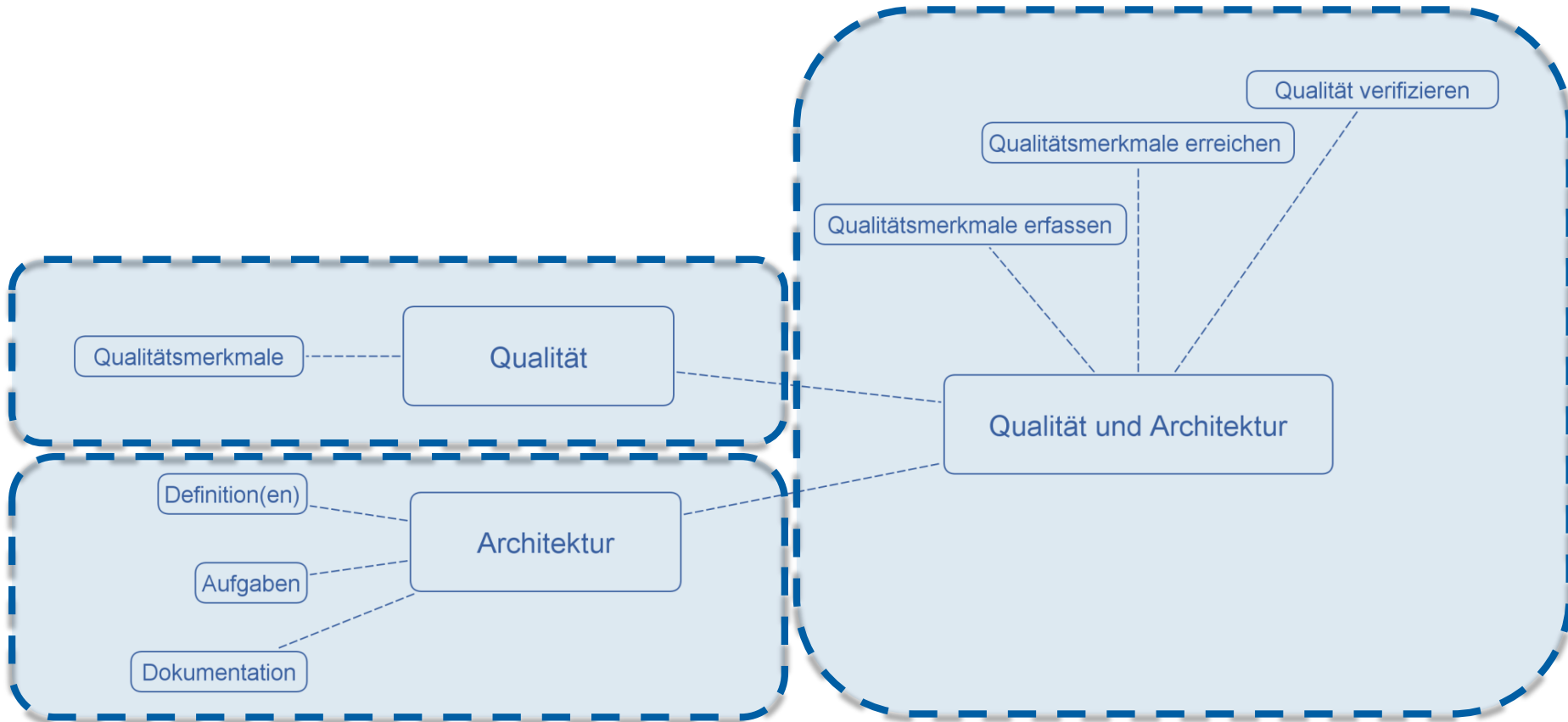
IKS-Thementag

Autor: Christoph Schmidt-Casdorff

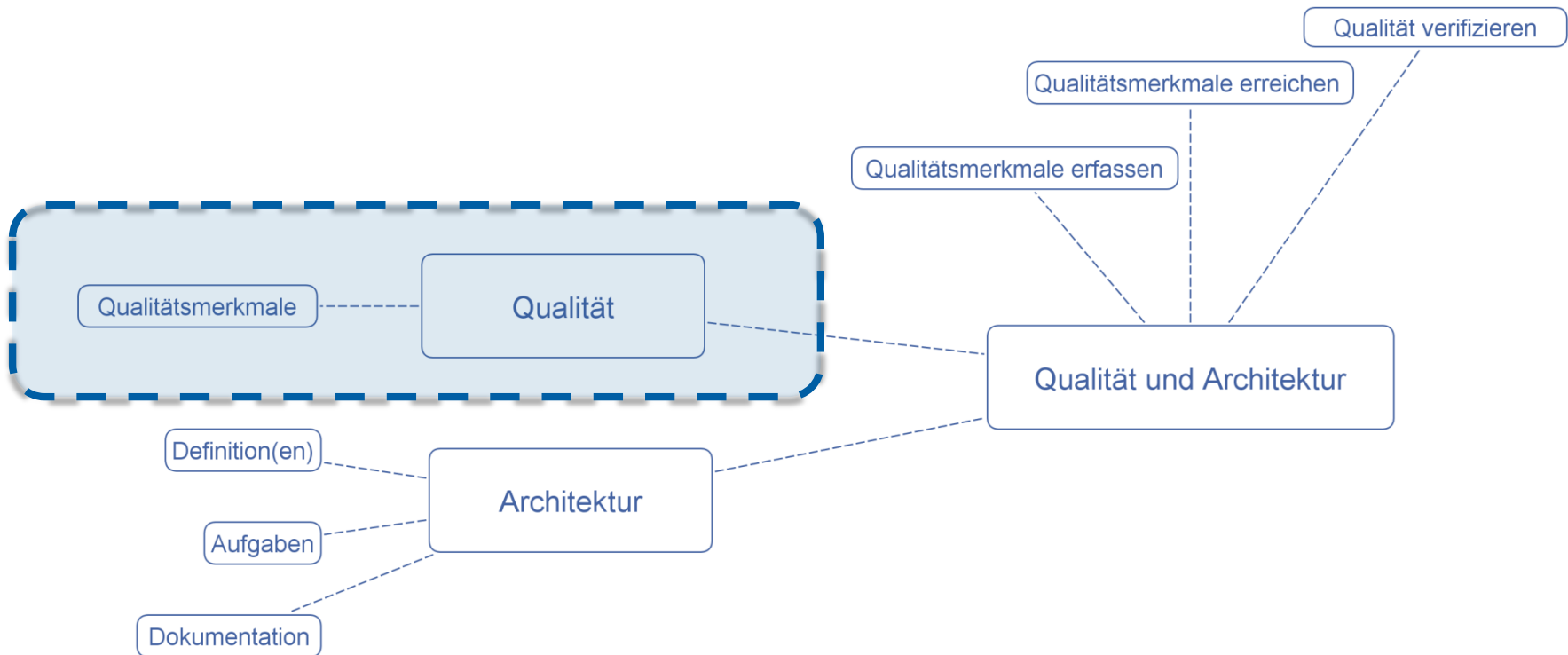
05.05.2015



Agenda



Qualität und Qualitätsmerkmale



Qualität als Treiber

Qualitätsmerkmale | Architektur | Qualitätsmerkmale erfassen | Qualitätsmerkmale erreichen |
Qualität verifizieren | Abschluss

Qualitätsmerkmale

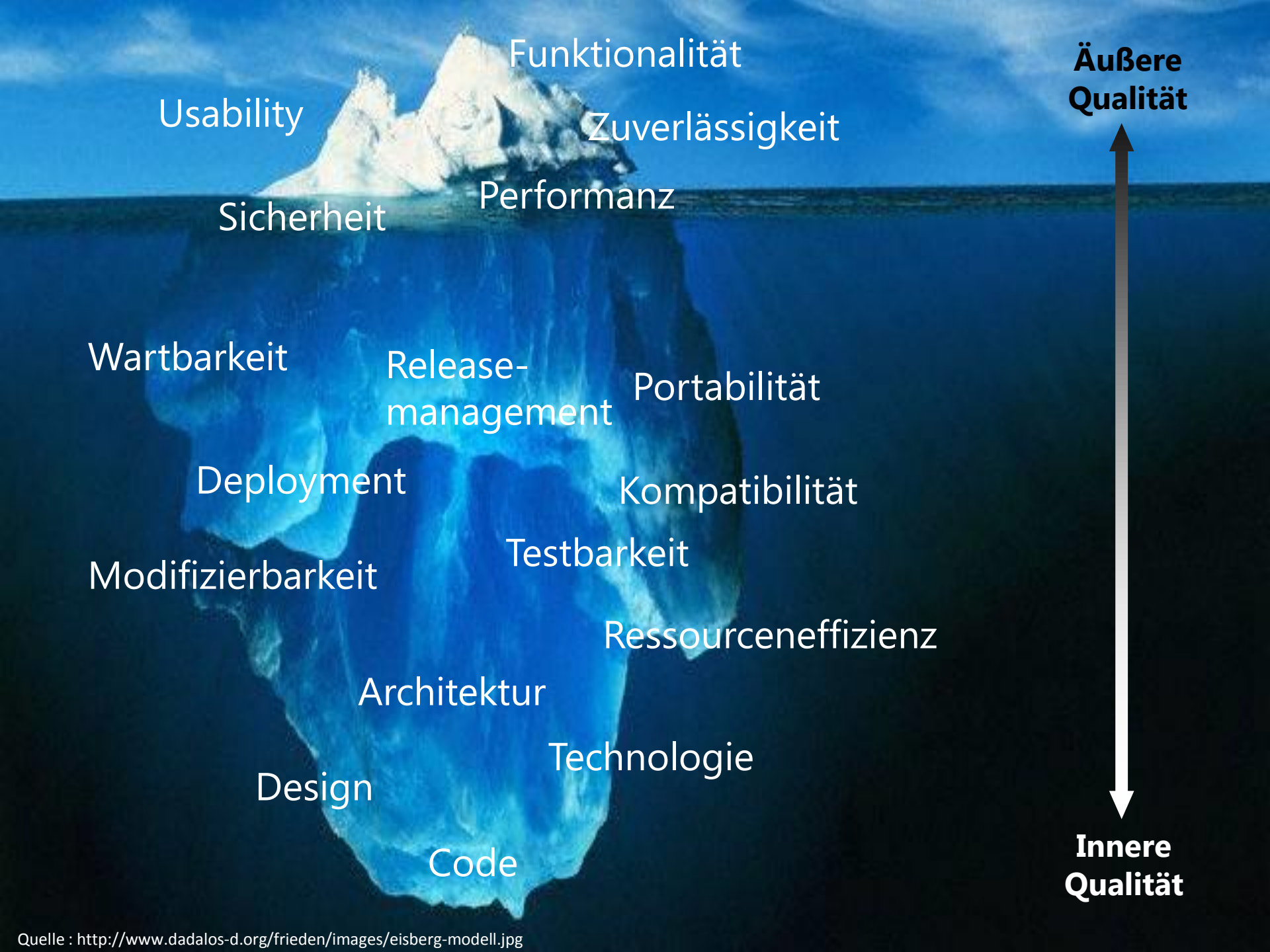
❖ Qualität ist als solche nicht ermittelbar

- ◆ Es können nur Eigenschaften eines Produktes/Prozesses bewertet werden

❖ *Qualitätsmerkmale* definieren

- ◆ Objektiv bestimmbare (und qualitätsrelevante) Eigenschaften
 - des Produkts/Software
- ◆ Alle Qualitätsmerkmale zusammen machen Qualität aus
- ◆ ISO 9126 * definiert eine Hierarchie von Qualitätsmerkmalen
 - Es gibt auch andere Hierarchien

* es existiert Nachfolger ist ISO/EIC 25010:211 (SQuaRE)



**Äußere
Qualität**

Funktionalität

Usability

Zuverlässigkeit

Sicherheit

Performanz

Wartbarkeit

Release-
management

Portabilität

Deployment

Kompatibilität

Modifizierbarkeit

Testbarkeit

Ressourceneffizienz

Architektur

Technologie

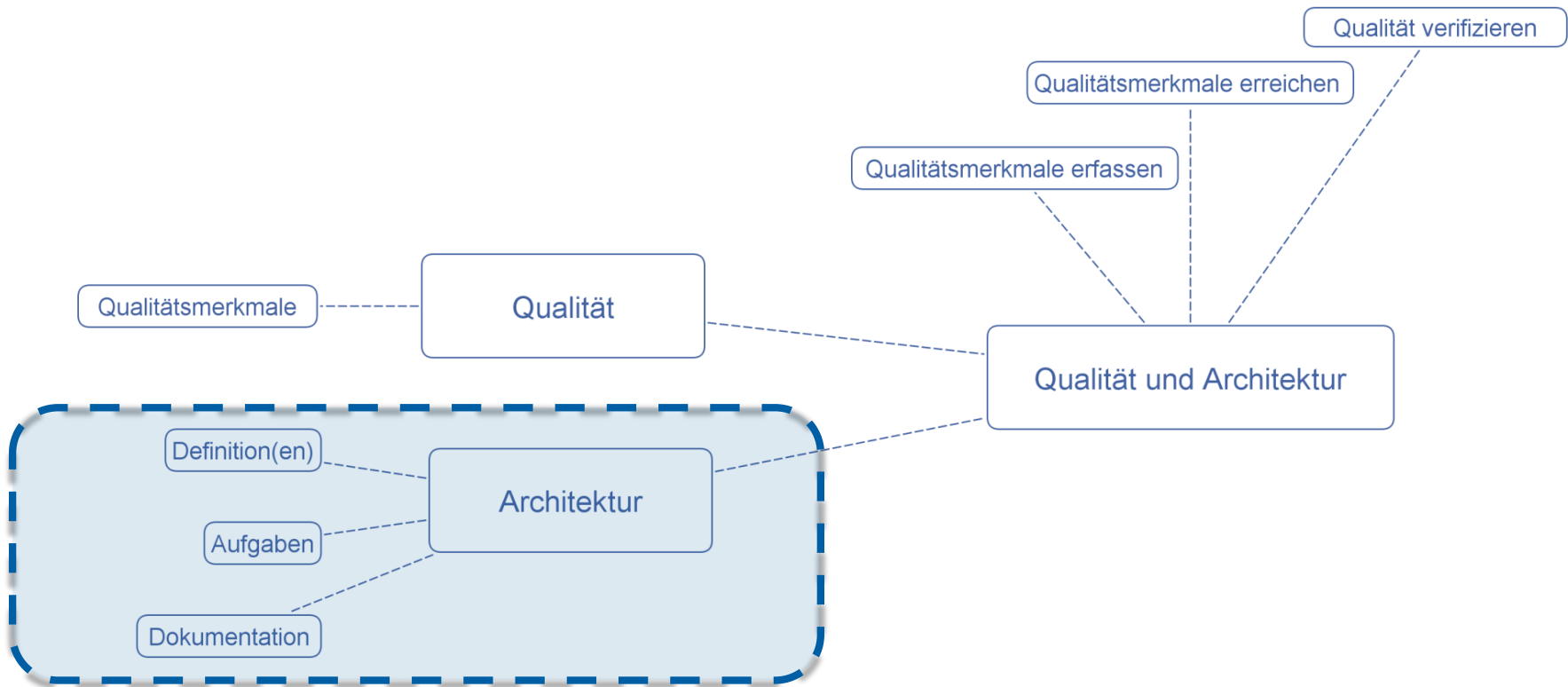
Design

Code

**Innere
Qualität**

Qualitätsmerkmale

- * Mit Qualitätsmerkmalen lässt sich Qualität beschreiben
- * Die Architektur muss
 - ◆ die Anforderungen an Qualitätsmerkmale erfassen
 - ◆ diese Anforderungen in Architekturentscheidungen umsetzen



Definitionen von Architektur

„Softwarearchitektur ist die Menge der hauptsächlichen/wesentlichen Architekturentscheidung eines Systems.

gemäß [Jacobson99]

„Die Softwarearchitektur eines Systems ist die Menge an Strukturen, die benötigt wird, um das System beurteilen zu können.

Sie umfasst Softwareelemente, die sichtbaren Eigenschaften dieser Elemente und deren Beziehungen untereinander.“

gemäß [Brass12]

Strukturen

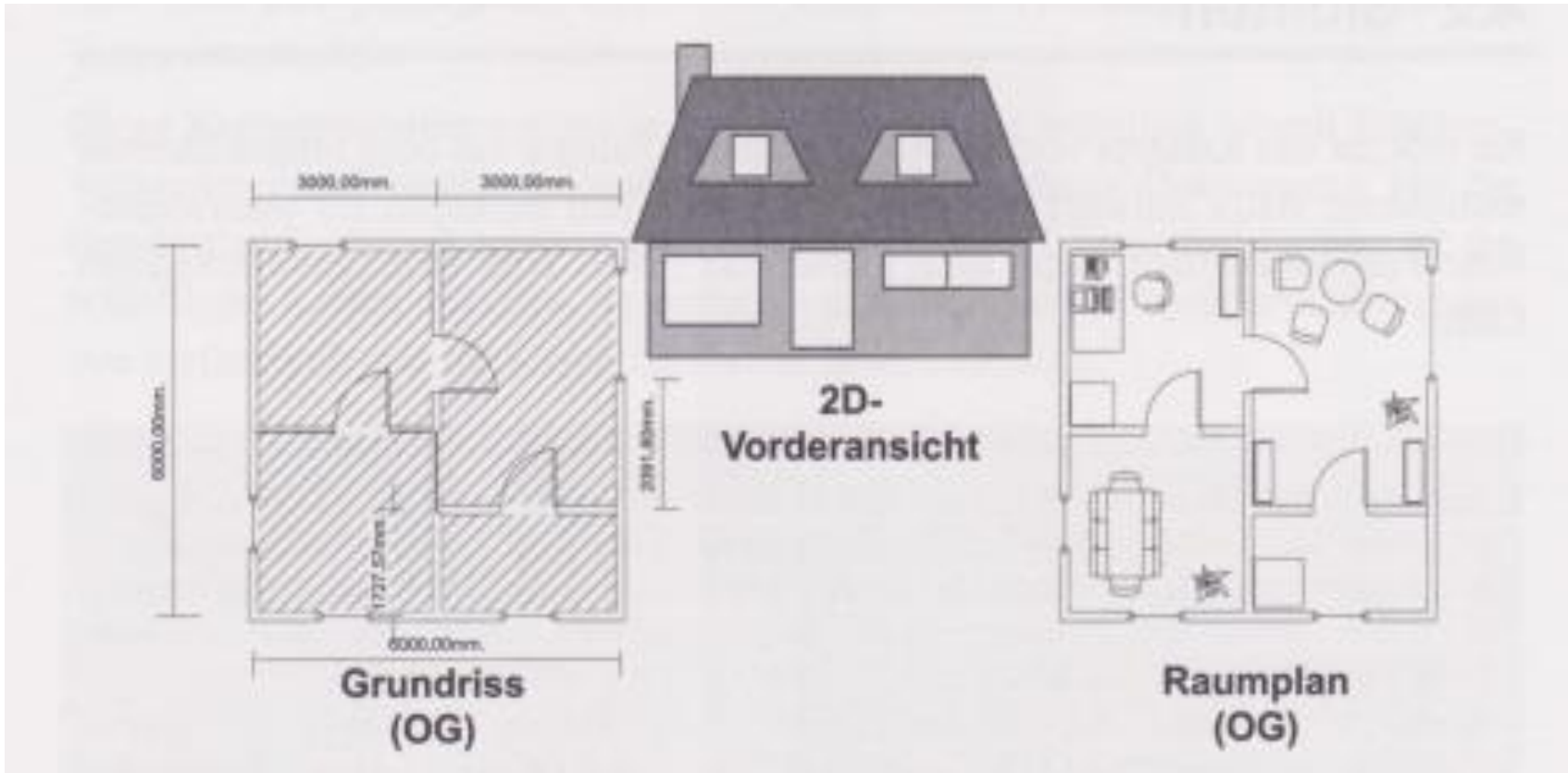
- ✿ Ein System kann unterschiedliche Strukturen haben
 - ◆ Strukturen können sich auf das statische Design, das Laufzeitverhalten usw. beziehen
 - ◆ Keine einzelne Struktur kann die Architektur definieren

- ✿ Strukturen der Architektur werden als Sichten (*View*) repräsentiert
 - ◆ Alle modernen Architekturansätze unterstützen das Konzept von *Views*

Architektursichten/-perspektiven

- * Perspektive definiert die Anforderung an den Informationsgehalt
- * Perspektiven helfen, Komplexität aufzuteilen
- * Perspektiven helfen, Aspekte auszublenden (*perspektivische Abstraktion*)
- * Perspektiven helfen, den Standpunkt des Adressaten einzunehmen
- * Summe der Perspektiven vermittelt eine Gesamtsicht
- * Sicht ist Ausprägung/Anwendung einer Perspektive in der aktuellen Architektur

Architektursichten-/perspektiven



Qualität als Treiber

Qualitätsmerkmale | **Architektur** | Qualitätsmerkmale erfassen | Qualitätsmerkmale erreichen |

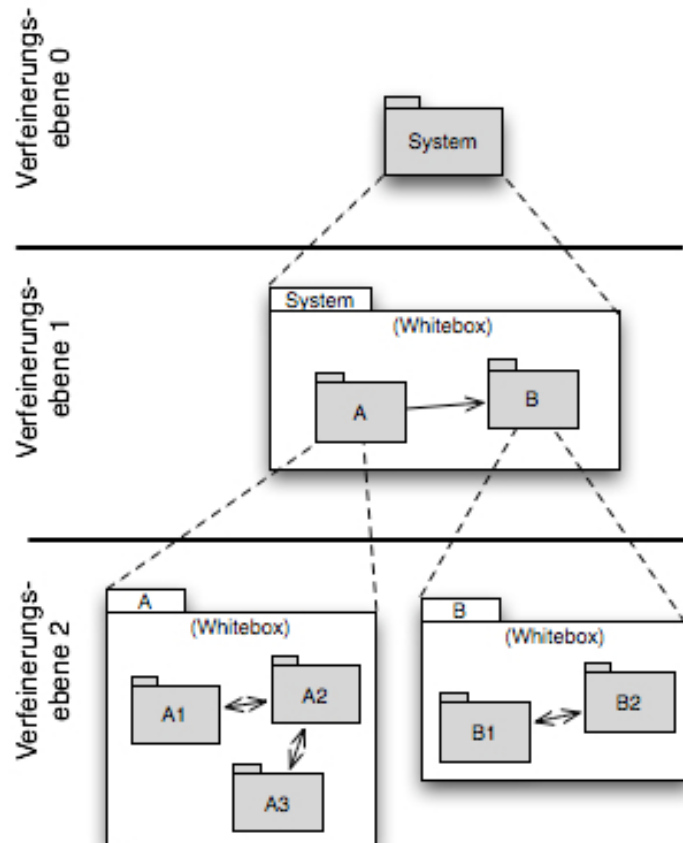
Qualität verifizieren | Abschluss

Sichten der Softwarearchitektur

- ✿ [Arc42] definiert Standardsichten
 - ◆ Kontextsicht, Bausteinsicht, Laufzeitsicht, Verteilungssicht
 - ◆ Notation auf Basis von UML
 - ◆ Weitere Sichten siehe [Brass12], [FMC]

- ✿ Es dürfen auch eigene Sichten entworfen werden

Verfeinerung innerhalb von Sichten



3. Kontextabgrenzung

4. Bausteinsicht

4.1 Verfeinerungsebene 1

4.1.1 Blackbox "A"

4.1.2 Blackbox "B"

4.2 Verfeinerungsebene 2

4.2.1 Whitebox "A"

4.2.1.1 Blackbox "A1"

4.2.1.2 Blackbox "A2"

4.2.1.3 Blackbox "A3"

4.2.2 Whitebox "B"

4.2.2.1 Blackbox "B1"

4.2.2.2 Blackbox "B2"

siehe [Arc42]

Qualität als Treiber

Qualitätsmerkmale | **Architektur** | Qualitätsmerkmale erfassen | Qualitätsmerkmale erreichen |

Qualität verifizieren | Abschluss

Architekturdokumentation

- * ist Basis jeder Kommunikation zwischen *Stakeholdern*
- * ist Mittel, um in das System einzuführen
- * ist Referenz, um das System in Richtung Implementierung zu überführen

- * befasst sich mit der Dokumentation der relevanten Views
 - ◆ fügt Informationen hinzu, welche die Beziehung der Views dokumentiert
 - ◆ Architekturentscheidungen bestimmen Views

- * hält alle relevanten Entscheidungen nach
 - ◆ Alle wesentlichen Entscheidungen müssen nachvollziehbar sein
 - ◆ Die Grundlagen aller Entscheidungen müssen dokumentiert sein

- * [Arc42] stellt ein Template bereit
 - ◆ *IKS* nimmt dieses als Basis eigener Architekturdokumentationen

Architektur

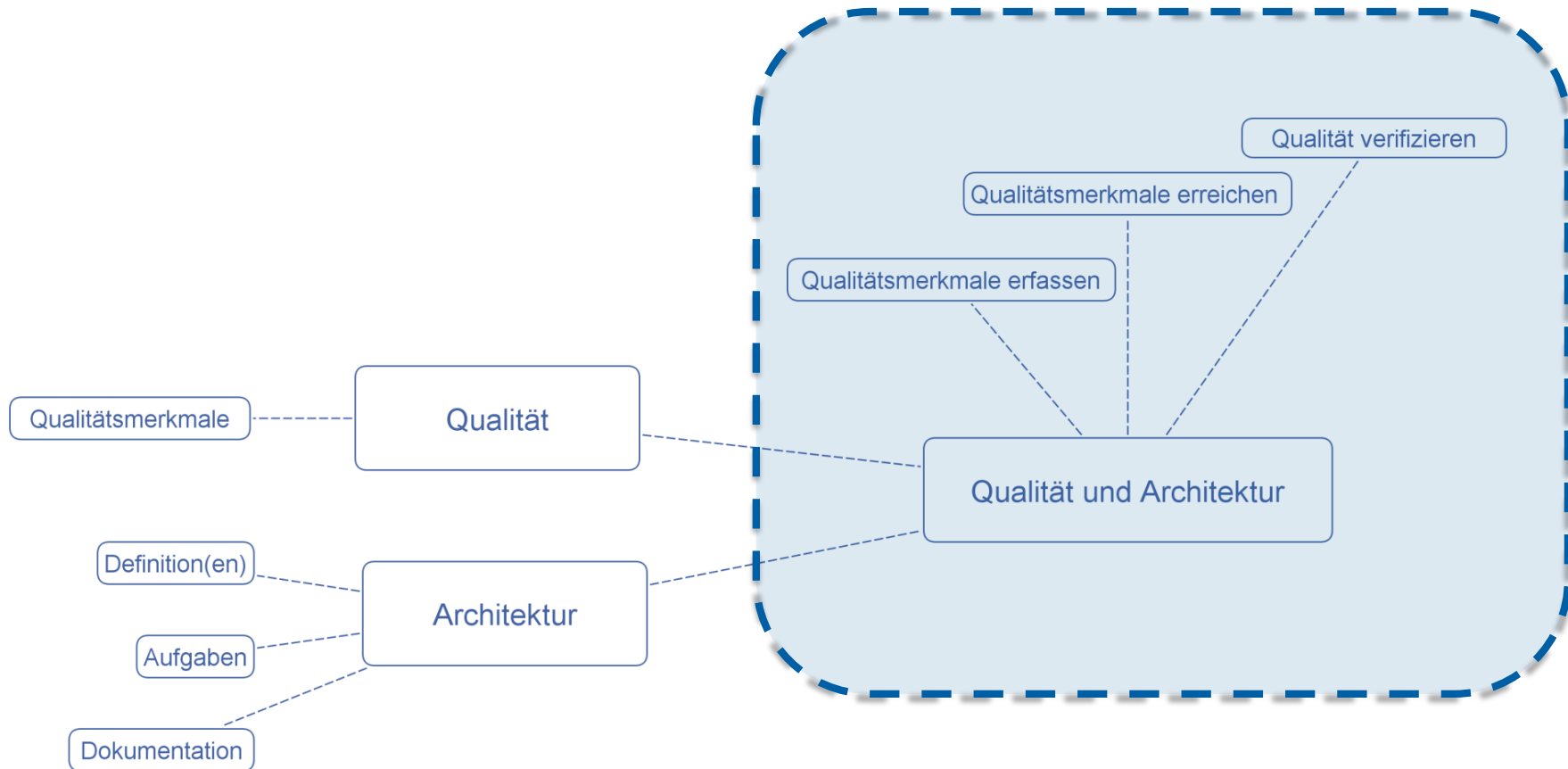
* Architektur ist Dekomposition

- ◆ Iterative Verfeinerung der Strukturen

* Kunst ist

- ◆ die richtige Zerlegung,
- ◆ den richtigen Abstraktionsgrad und
- ◆ die richtige Form der Dokumentation

zu finden.



Die Softwarearchitektur ist verantwortlich, dass ein System die geforderten Qualitätsmerkmale erfüllt.

Warum?

Wo kommen die QMe her?

Wie kann das gelingen ?

Wie wird dies sichergestellt und geprüft?

Softwarearchitektur und Qualitätsmerkmale

Warum ist Softwarearchitektur für Qualitätsmerkmale verantwortlich?

- ✿ Erreichen von Qualitätsmerkmalen bedingt (i.d.R.) weitreichende Architekturentscheidungen
 - ◆ Qualitätsmerkmale beeinflussen daher entscheidend Softwarestrukturen
- ✿ Qualitätsmerkmale sind (die) entscheidende(n) Einflussfaktoren der Architektur
 - ◆ Architektur hat daher Eigeninteresse, diese so exakt wie möglich zu kennen

Softwarearchitektur und Qualitätsmerkmale

- ✿ **Architektur hat aus Sicht der fachlichen Funktionalität keine Qualitätsmerkmale im Fokus**
 - ◆ Struktur des Systems nur aus Sicht der Fachlichkeit adressiert keine Qualitätsmerkmale
 - ◆ Qualitätsmerkmale der Architektur müssen explizit behandelt werden

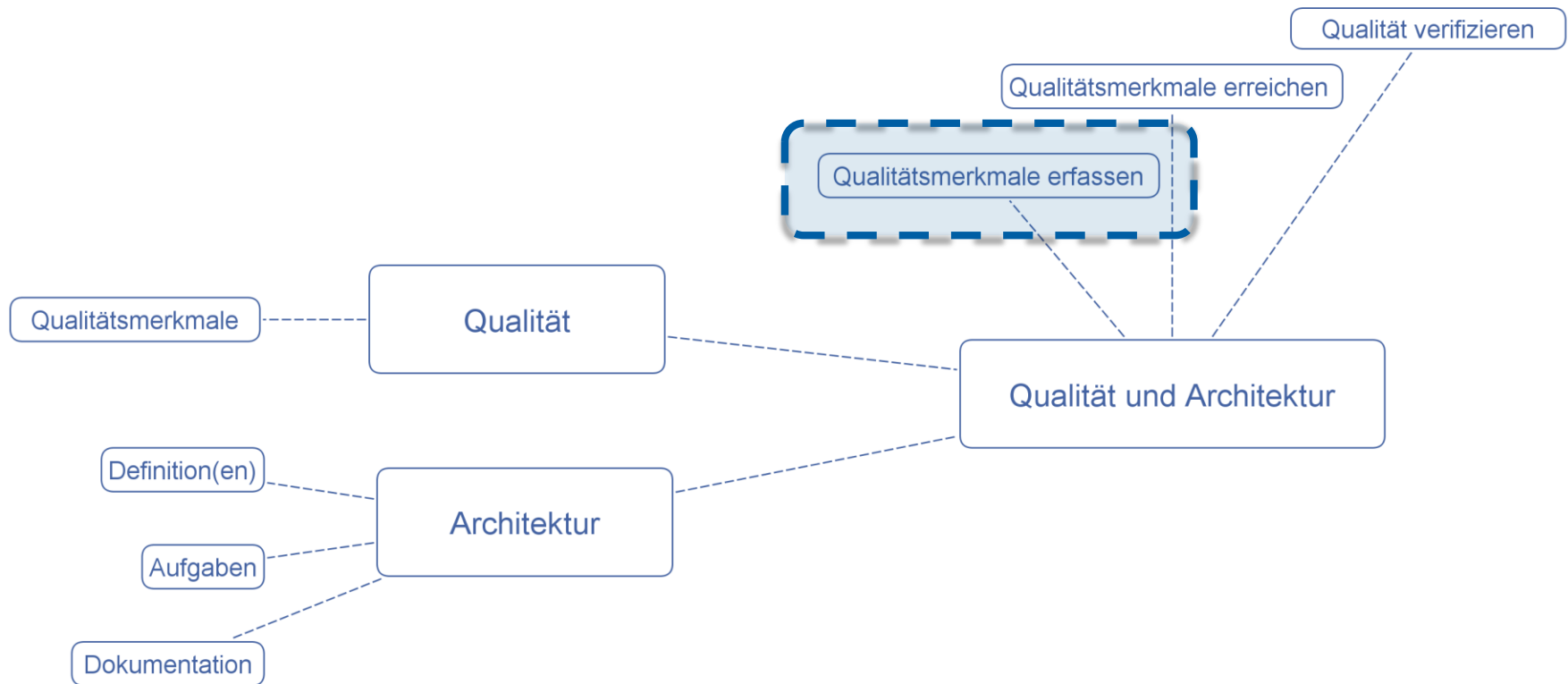
- ✿ **Wir setzen im Weiteren nicht-funktionale Anforderungen (NFA) mit Qualitätsmerkmalen gleich**
 - ◆ Funktionalität ist (formal) auch eine Qualitätsmerkmal

Softwarearchitektur und Qualitätsmerkmale

Softwarearchitektur ist nur bedingt für Erfassung und Prüfung von Qualitätsmerkmalen zuständig

- * Diese Zuständigkeiten führen immer zu Diskussionen
- * Dennoch sollte die Architektur alle Tätigkeiten rund um Qualitätsmerkmale koordinieren
 - ◆ Nicht erfüllte Qualitätsmerkmale werden i.d.R. der Architektur zur Last gelegt

Qualitätsmerkmale erfassen



Anforderungen an Qualitätsmerkmale

Das System soll fehlertolerant sein

Das System muss performant sein

Das System muss leicht zu ändern sein

Das System muss stabil sein

Wie müssen Qualitätsmerkmale gut formuliert sein ?

Qualität als Treiber

Qualitätsmerkmale | Architektur | **Qualitätsmerkmale erfassen** | Qualitätsmerkmale erreichen |

Qualität verifizieren | Abschluss

Beispiel einer Anforderung an Qualität

Unter welchen Umständen?

Was wird getan?

Welcher Teil des Systems ist betroffen?

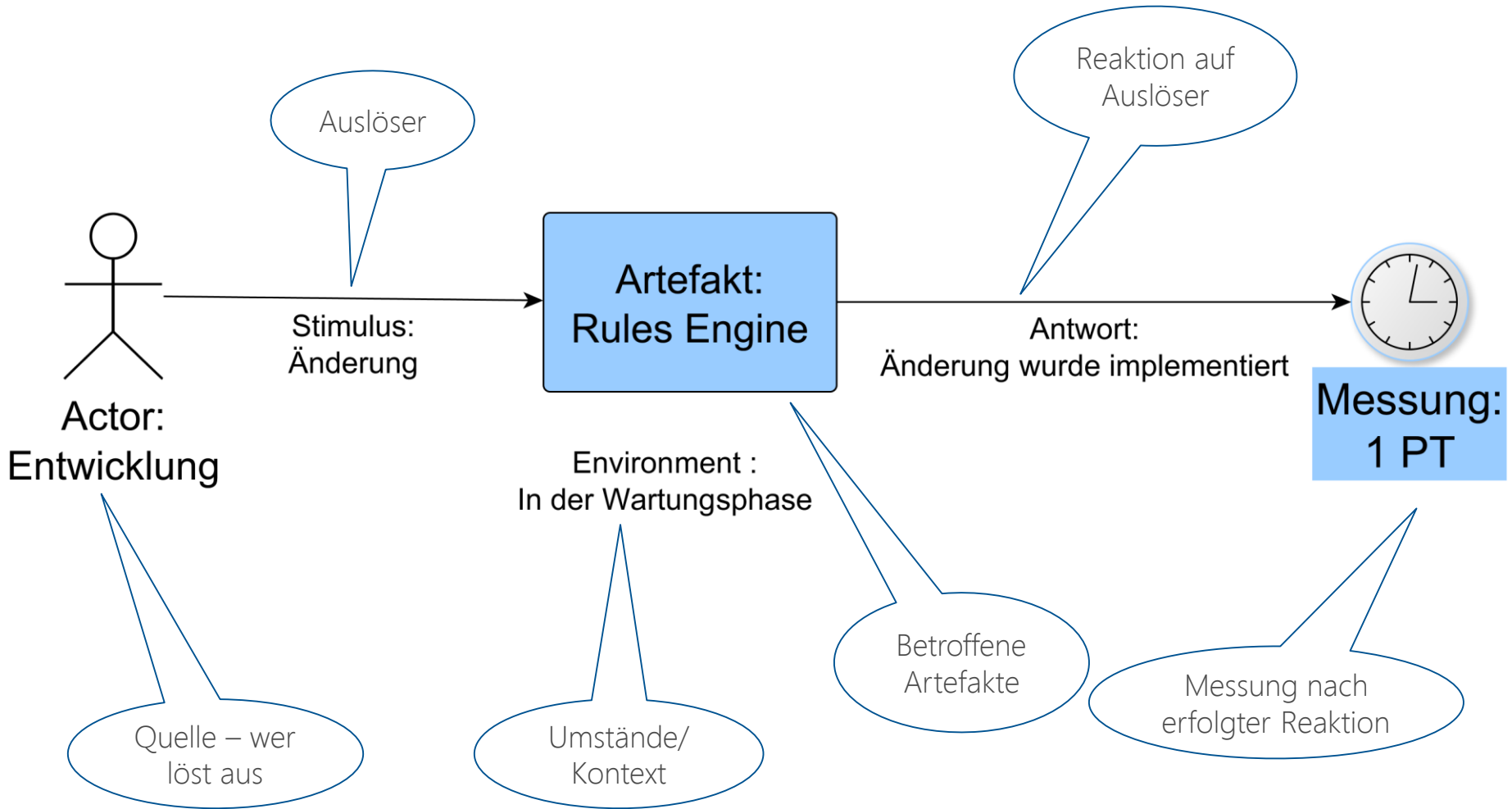
In der **Wartungsphase** des Systems wird eine **Änderung** an der **Rules Engine** des Systems vorgenommen.

Diese Änderungen muss **innerhalb eines Tages fertig implementiert** sein.

Wie ist das Ergebnis zu messen/zu bewerten?

Mit welcher Reaktion des Systems?

Beispiel eines Qualitätsszenarios



Schablone eines Qualitätsszenarios

	Beschreibung
Auslöser (<i>stimulus</i>)	Beschreibt eine spezifische Zusammenarbeit des auslösenden Aktors/Stakeholder mit dem System
Quelle des Auslösers (<i>source</i>)	beschreibt, woher der Auslöser kommt
Systembestandteil (<i>artifact</i>)	beschreibt, welcher Bestandteil des Systems vom Auslöser betroffen ist
Umgebung (<i>environment</i>)	beschreibt die Bedingung, unter der der Auslöser auftritt
Antwort (<i>response</i>)	beschreibt die Reaktion des Systems auf den Auslöser
Antwortmetrik (<i>response measure</i>)	beschreibt, wie die Antwort gemessen oder bewertet werden kann.

siehe [Brass12]

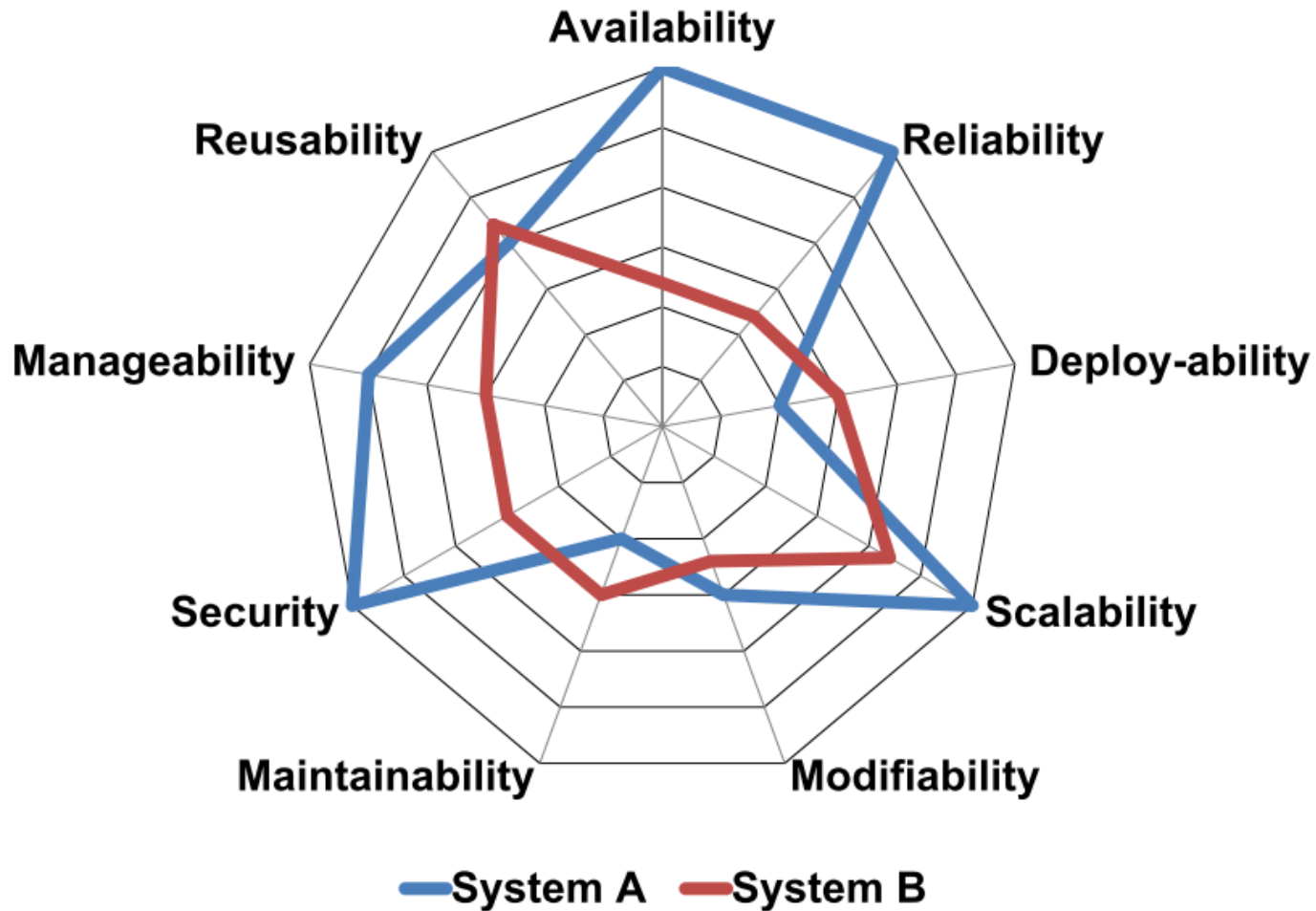


Wie erhalte ich Qualitätsszenarien?

Qualitätsszenarien sind Schablonen/Muster, um Anforderungen an Qualitätsmerkmale zu dokumentieren.

Wie finde ich die Anforderungen an Qualitätsmerkmale?

Qualitätsmerkmale sind systemspezifisch



18

System Properties Web siehe [IBM14]

Qualität als Treiber

Qualitätsmerkmale | Architektur | **Qualitätsmerkmale erfassen** | Qualitätsmerkmale erreichen |

Qualität verifizieren | Abschluss

Wie erhalte ich Qualitätsszenarien?

* Workshop, um Qualitätsmerkmale des Systems zu finden

* Teilnehmer sind Stakeholder

- ◆ Alle, die Anforderungen an Qualitätsmerkmale haben
- ◆ I.d.R. bevor die Architektur erstellt wird

* Quellen für Qualitätsmerkmale

- ◆ Geschäftsziele bilden Grundlage für Qualitätsmerkmale
- ◆ Fachliche Anforderungen enthalten häufig Qualitätsmerkmale
- ◆ Rahmenbedingungen beeinflussen häufig Qualitätsmerkmale
- ◆ Architekturplan/Architekturskizze bilden Diskussionsbasis

Wie erhalte ich Qualitätsszenarien?

Besonderheiten:

- ✿ Teilnehmer haben sich i.d.R. nicht mit der Thematik beschäftigt
 - ◆ und müssen daher besonders eingeführt werden
- ✿ Problembereich ist umrissen
 - ◆ Qualitätsmerkmale sind z.B. durch ISO 9126 definiert
 - ◆ „*Walking the System Properties Web*“

„Walking the System Properties Web“



siehe [IBM14]

Qualität als Treiber
 Qualitätsmerkmale | Architektur | **Qualitätsmerkmale erfassen** | Qualitätsmerkmale erreichen |
 Qualität verifizieren | Abschluss

Ergebnis eines Workshops

* Liste der Qualitätsszenarien

- ◆ Priorisiert
- ◆ Abgestimmt
- ◆ Allgemein akzeptiert

* I.d.R. Folgetermine, um

- ◆ Zielkonflikte aufzulösen (*tradeoffs*)
- ◆ Lösungen zu bewerten

Quality Attribute Workshop

- * Formales Verfahren um Qualitätsmerkmale zu finden
 - ◆ [Brass12], in leichtgewichtiger Variante in [IBM14]
- * Beschreibt im Wesentlichen einen Findungsworkshops

Dokumentation von Anforderungen an Qualitätsmerkmale

* Formal nach Schablone eines Qualitätsszenarios

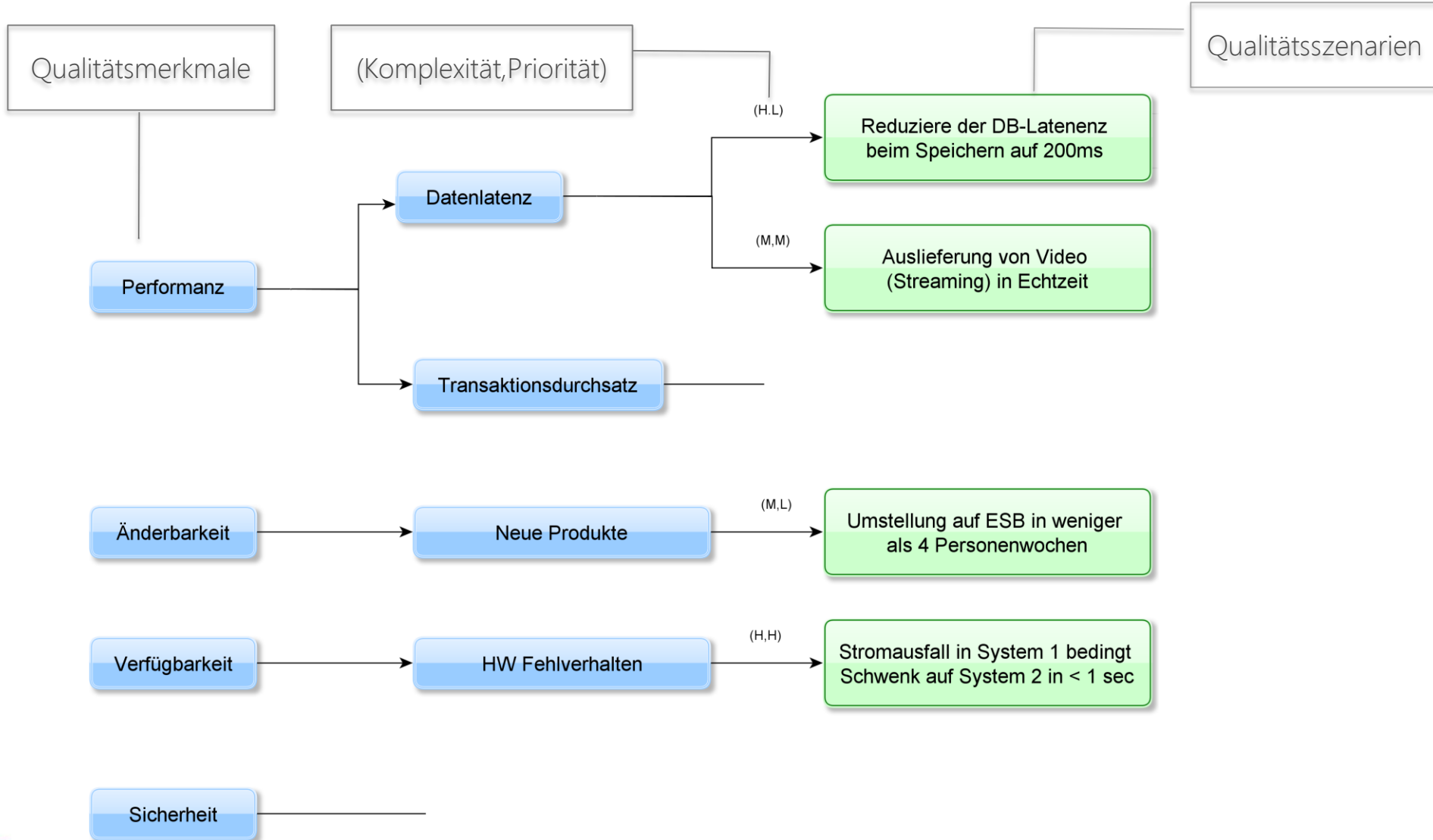
* Was darf nicht fehlen?

- ◆ Exakte Abnahmekriterien
 - Definition des Abnahmesystems
 - Wer nimmt ab?
 - Je höher das Qualitätsmerkmal priorisiert ist, desto wichtiger sind exakte Kriterien
- ◆ Auswirkungen der Abnahme
 - Wer ist für das Abnahmesystem verantwortlich?
 - Frühzeitig in den Projektplan aufnehmen

* Qualitätsbaum

- ◆ Priorisiert die Qualitätsmerkmale und -szenarien
 - bewerte die Komplexität
- ◆ Liefert Überblick über Qualitätsmerkmale und -szenarien

Qualitätsbaum



Qualität als Treiber

Qualitätsmerkmale | Architektur | **Qualitätsmerkmale erfassen** | Qualitätsmerkmale erreichen |
Qualität verifizieren | Abschluss

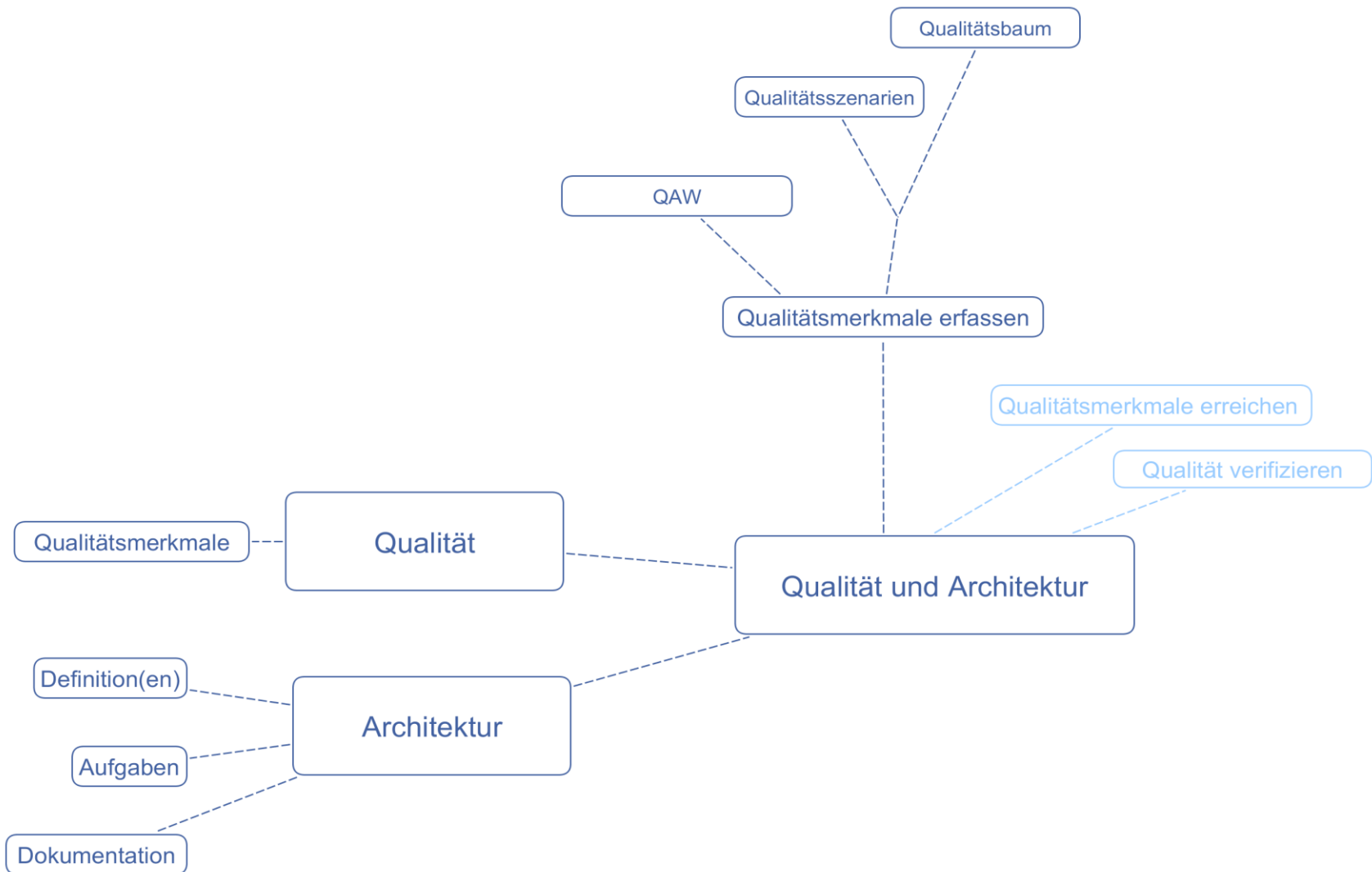
Dokumentation von Qualitätsanforderungen

- * Qualitätsszenarien müssen dokumentiert sein
 - ◆ [Arc42] sieht separates Kapitel vor

- * Priorisierung der Qualitätsszenarien muss dokumentiert sein
 - ◆ Qualitätsbaum

- * Evtl. Kompromisse müssen als Architekturentscheidung dokumentiert sein

Qualitätsmerkmale erfassen

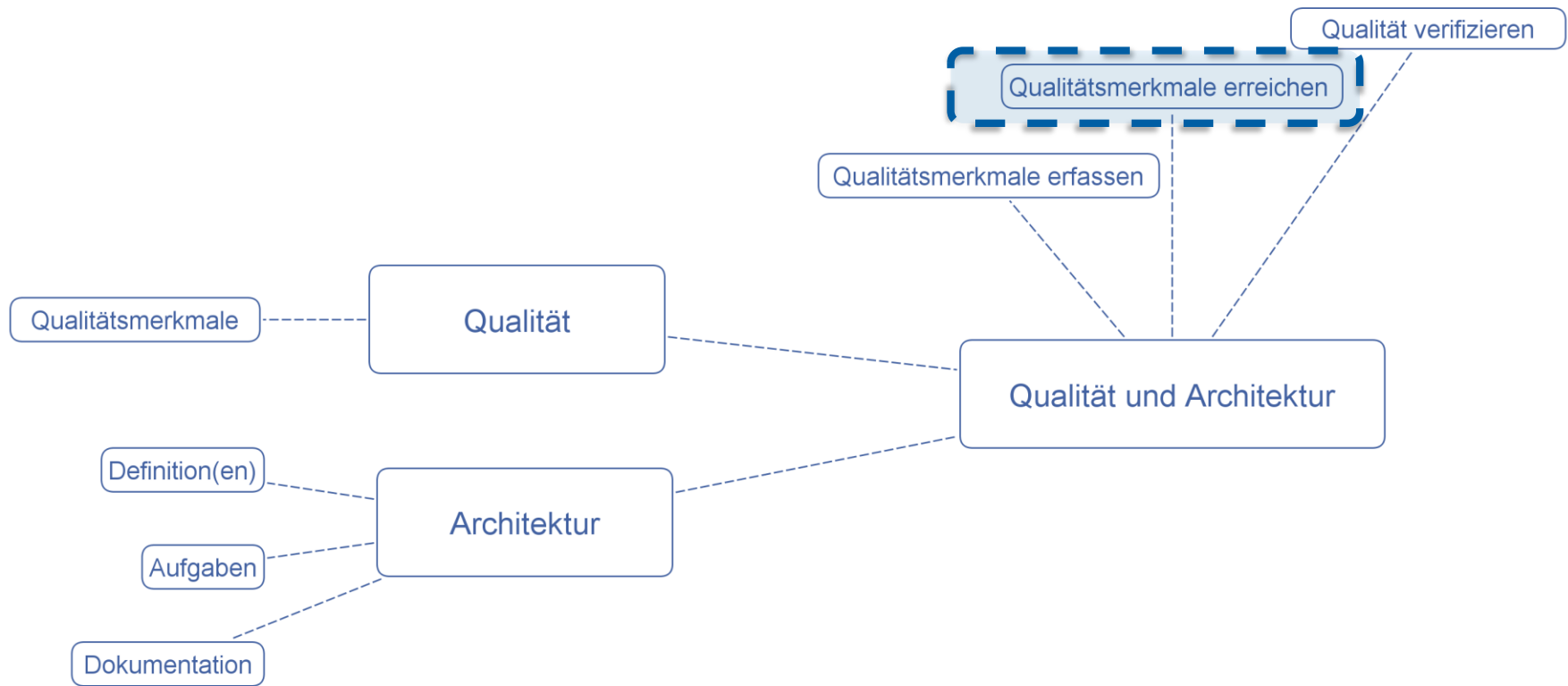


Qualität als Treiber

Qualitätsmerkmale | Architektur | **Qualitätsmerkmale erfassen** | Qualitätsmerkmale erreichen |

Qualität verifizieren | Abschluss

Qualitätsmerkmale erreichen



Umsetzung der Qualitätsmerkmale

* Voraussetzung

- ◆ Abgestimmte, priorisierte Liste an Qualitätsszenarien
 - inkl. Qualitätsbaum
- ◆ Architekturelevante funktionale Anforderungen
- ◆ Rahmenbedingungen

* Aufgabe

- ◆ Architekturentscheidungen zu treffen, so dass das System die geforderten Qualitätsmerkmale erfüllt
 - Das ist eine technologisch sehr anspruchsvolle Aufgabe

Wir wollen Ihnen einen Eindruck möglicher Lösungsansätze bieten.

Um eine Architektur zu erstellen, ist es sinnvoll, bestehende Lösungen zu nutzen. [Brass12]

Wo sind die Lösungen, die uns helfen, Qualitätsmerkmale umzusetzen?

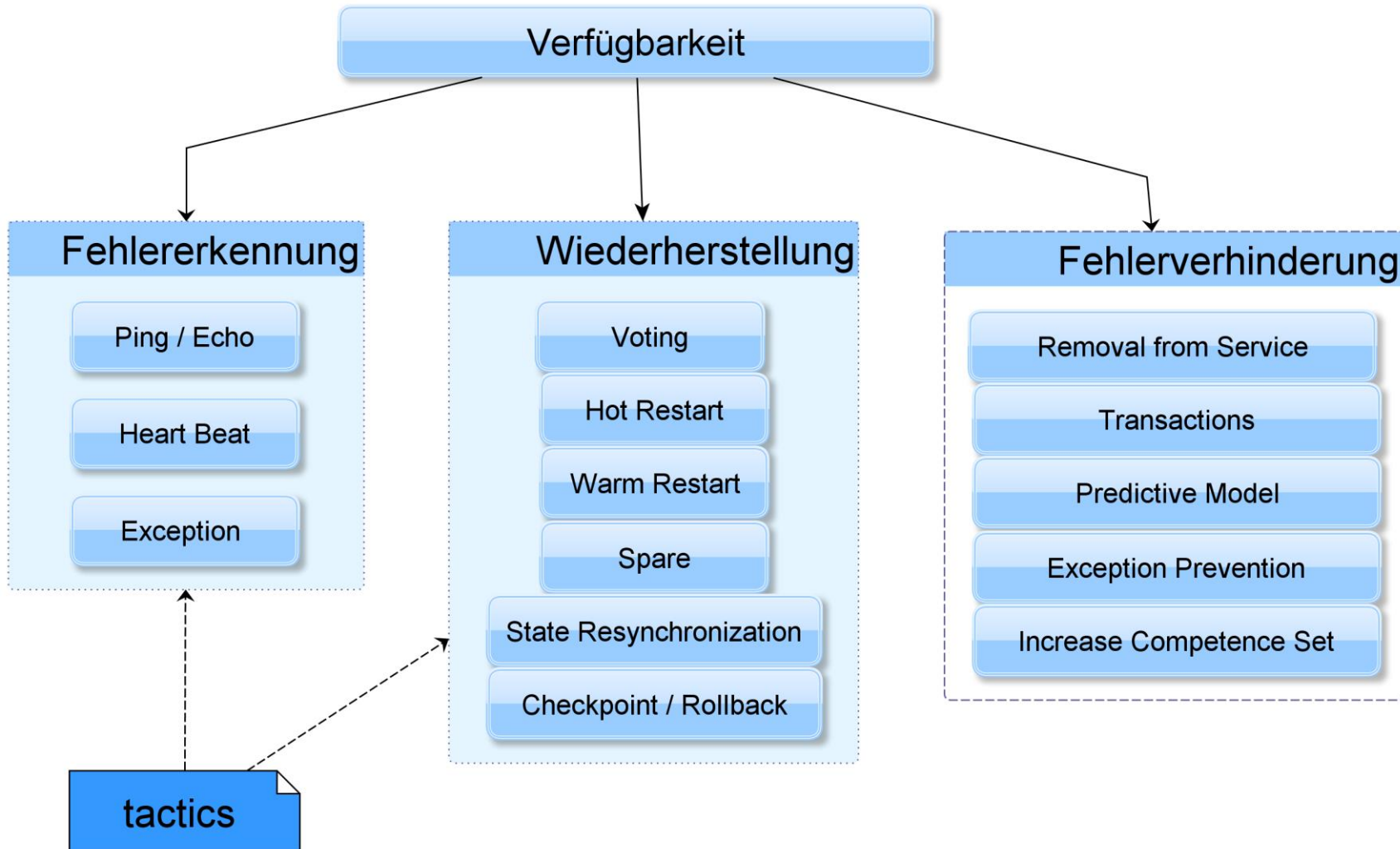
Was würde helfen?

- * Katalog von möglichen Architekturentscheidungen
 - ◆ Architekturentscheidungen, welche spezielle Qualitätsmerkmale unterstützen
- * Vorgehensweise, um aus den getroffenen Architekturentscheidungen eine Gesamtarchitektur zu schaffen
 - ◆ In welcher Reihenfolge geht man die Qualitätsszenarien an?
 - ◆ Wie löst man Zielkonflikte zwischen den einzelnen Qualitätsmerkmalen?
 - ◆ Wie fügen sich die einzelnen Lösungen in die Gesamtarchitektur ein?

tactics

- ✿ tactics sind Architekturentscheidungen, welche auf ein spezielles Qualitätsmerkmal wirken
- ✿ tactics sind bewährte Design-Entscheidungen
 - ◆ ausgerichtet an einzelnen Qualitätsmerkmalen
 - ◆ beinhalten nicht die Wahl der technischen Umsetzung

tactics für Verfügbarkeit



Qualität als Treiber

Qualitätsmerkmale | Architektur | Qualitätsmerkmale erfassen | **Qualitätsmerkmale erreichen** |

Qualität verifizieren | Abschluss

tactics

✿ tactics sind in [Brass12] gut erläutert und es ist beschrieben, unter welchen Umständen sie anzuwenden sind

- ◆ Welche Qualitätsmerkmale sie adressieren
- ◆ Welche Modelle zugrunde liegen
 - *Queuing Model, Scheduling Model* etc.

✿ [Brass12] hat Kataloge von tactics erarbeitet für

- ◆ Testbarkeit
- ◆ Verfügbarkeit (aka Fehlertoleranz)
- ◆ Interoperabilität
- ◆ Performance (aka Effizienz)
- ◆ Sicherheit
- ◆ Benutzbarkeit
- ◆ u.v.m

Architekturmuster

* Architekturmuster sind allgemeine Architekturstrukturen

- ◆ Abgestimmt
- ◆ Allgemeingültige Lösungen

* Architekturmuster

- ◆ sind nicht hinsichtlich ihrer Wirkung auf Qualitätsmerkmale strukturiert
- ◆ adressieren mehrere Qualitätsmerkmale
- ◆ die bekanntesten sind *Layer Pattern, Client-Server Pattern*
 - *Reflection, Microkernel, Broker, ...**

* weitere siehe [Buschmann 1996], [Brass12]

Bewährte Lösungen - tactics und Architekturmuster

* Architekturmuster umfassen *tactics*

- ◆ tactics sind die Bausteine des Designs, auf denen Architekturmuster aufsetzen
- ◆ Welche tactics werden in welchen Architekturmuster in welcher Form berücksichtigt?
- ◆ Den Zusammenhang kann man nachlesen [Brass12]

* *tactics* verfeinern Architekturmuster

- ◆ Architekturmuster ist gewählt
- ◆ tactics können eingesetzt werden, um nachträglich Qualitätsmerkmale zu erreichen

* Architekturmuster und *tactics* repräsentieren konzeptuelle Werkzeuge im „Werkzeugkasten des Architekten“.

- ◆ Gute Handwerker halten ihren Werkzeugkasten aufgeräumt und *up to date*

Attribute Driven Design (ADD)

Ausgangslage für eine Gesamtarchitektur

- * Architekturelevante funktionalen Anforderungen (*[Brass12]*)
- * (Architekturelevanten) Qualitätsszenarien
- * Rahmenbedingungen

siehe [Brass12]

Qualität als Treiber

Qualitätsmerkmale | Architektur | Qualitätsmerkmale erfassen | **Qualitätsmerkmale erreichen** |


Qualität verifizieren | Abschluss

Attribute Driven Design (ADD)

* Methodische Verfeinerung der Architektur unter Berücksichtigung der Qualitätsmerkmale

- ◆ Siehe [Brass12]

* Vorgehen im *Attribute Driven Design*

- 
- ◆ Systemteil wird bestimmt
 - ◆ Relevante Anforderungen werden bestimmt
 - ◆ Architekturentscheidungen werden getroffen
 - Priorisierung wird ausgewertet
 - ◆ Neue Strukturen werden dokumentiert
 - Architektursichten¹⁾ werden erstellt/ergänzt
 - Hier sollte ein Review durchgeführt werden

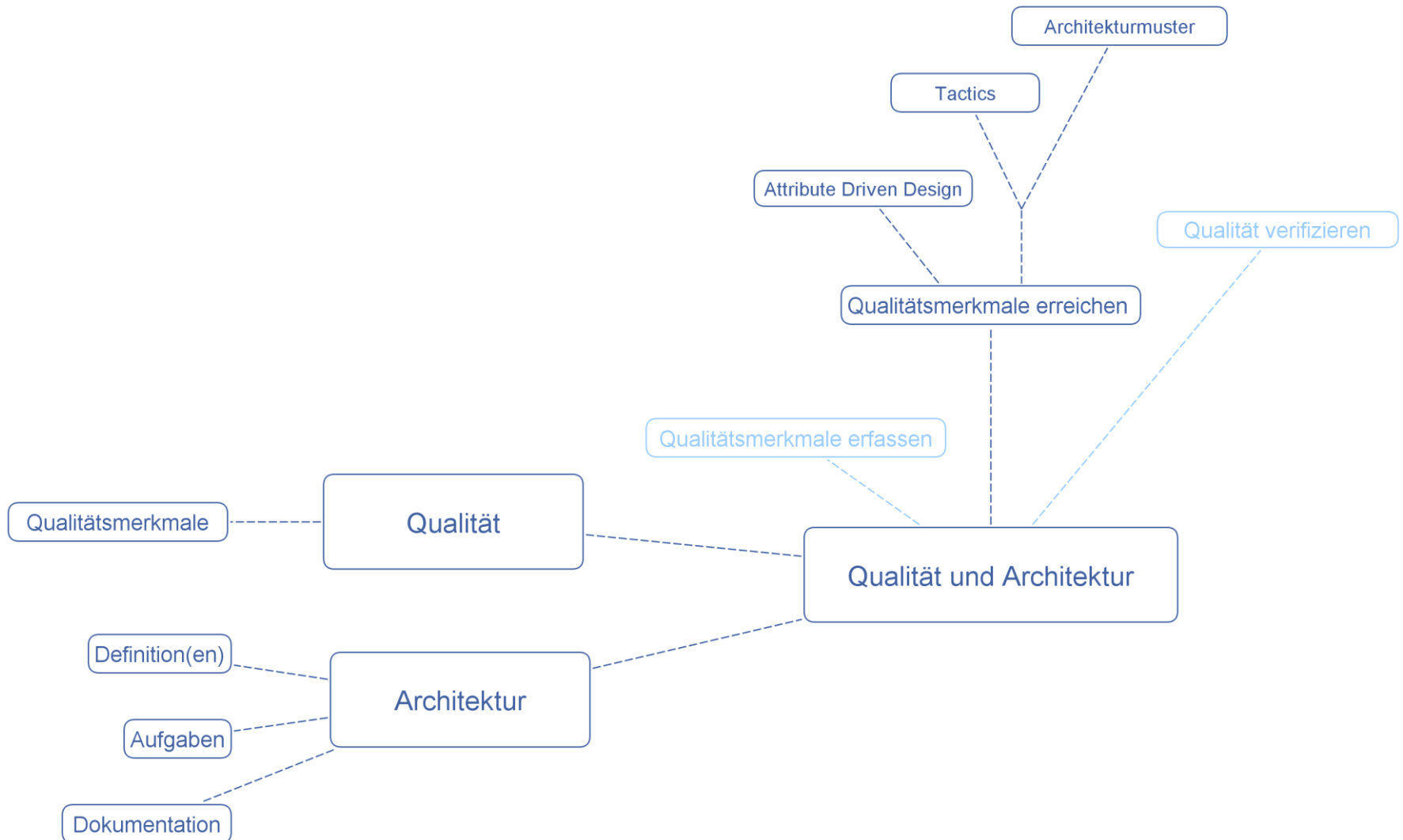
1) [Brass12] trennt zwischen Architekturentwürfen in ADD und der nachgelagerten, eigentlichen Dokumentation

Attribute Driven Design (ADD)

- * Architektur ist iterative Dekomposition

- * Kunst ist
 - ◆ die richtige Zerlegung (in der richtigen Reihenfolge) und
 - ◆ den richtigen Abstraktionsgradzu finden.

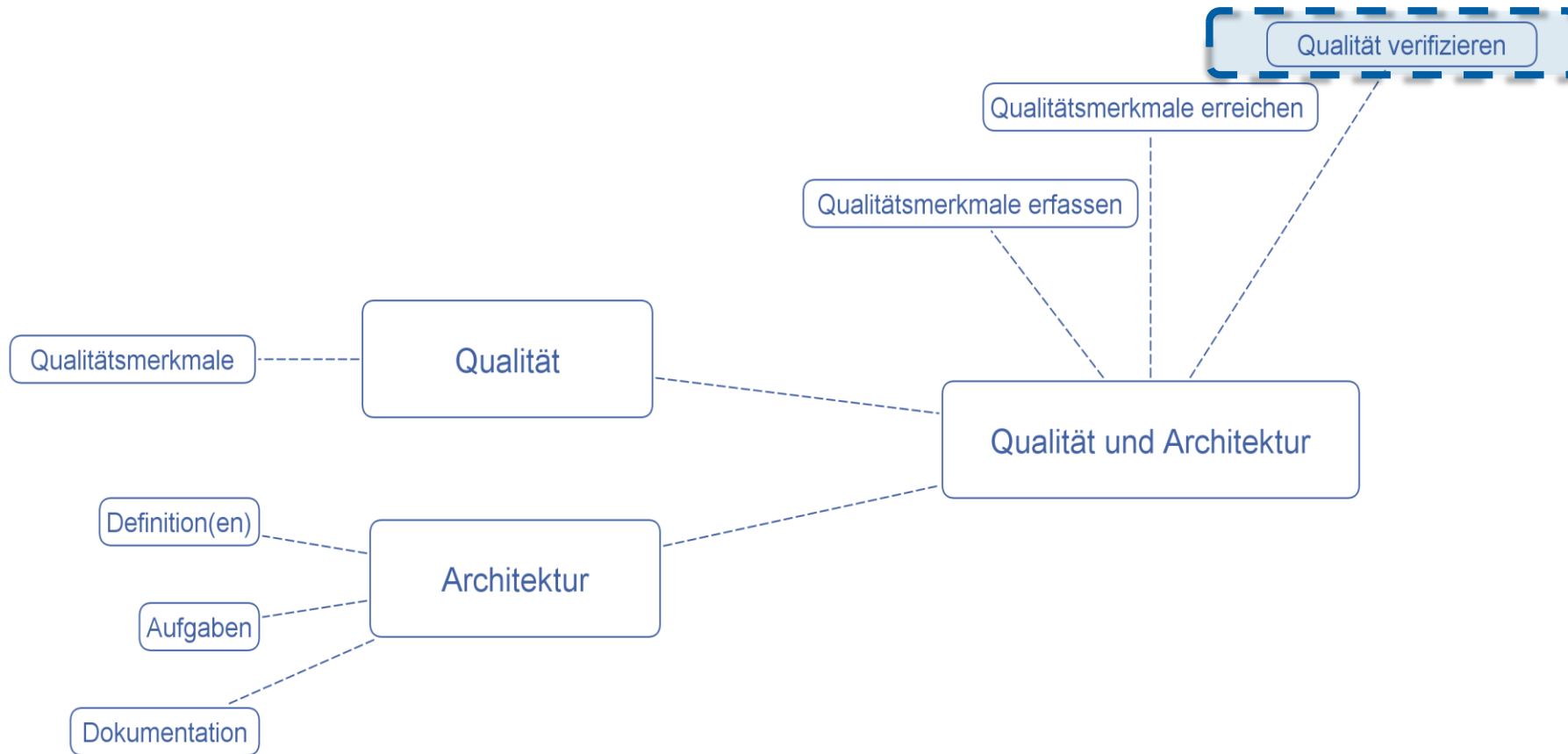
Attribute Driven Design unterstützt Sie dabei.



Qualität als Treiber

Qualitätsmerkmale | Architektur | Qualitätsmerkmale erfassen | **Qualitätsmerkmale erreichen** |
Qualität verifizieren | Abschluss

Qualität verifizieren



Architektur bewerten

* Architekturbewertung

- ◆ Während der Erstellung der Architektur
 - *Attribute Driven Design (ADD)* enthält Reviewphase
- ◆ Auf Basis einer bestehenden Architektur

* Ziel: Bewertung der Architekturentscheidungen vor dem Hintergrund

- ◆ der Qualitätsszenarien
- ◆ der Geschäftsziele

ATAM

* *Architecture Tradeoff Analysis Method*[®] (ATAM)

- ◆ setzt auf bestehender Architektur auf
 - oft auch auf bestehendem System
- ◆ Das Evaluationsteam ist (i.d.R.) extern (bzgl. des Projektteams)

* soll Risiken der Architekturentscheidungen aufdecken

- ◆ aber auch sogenannte *non-risks*

* soll negative Trends des Systems Architekturentscheidungen zuordnen

* soll *tradeoffs* erkennen

- ◆ Entscheidungen, die mehr als eine Qualitätsmerkmal betreffen
- ◆ Konsequenzen aus diesem Zielkonflikt

siehe [Brass12], [Kazman00], [Northrop11]

ATAM

* ATAM ist ein Review-Verfahren, dass

- ◆ auf die Entdeckung von Risiken ausgelegt ist
- ◆ relativ kurzläufig, relativ leichtgewichtig ist
 - gibt es auch als „*lightweight*“ Variante

* Voraussetzungen

- ◆ Architektur muss vorhanden sein
 - Fehlende Qualitätsszenarien werden im Prozess ermittelt
- ◆ Architekt muss mitarbeiten und bereit sein, die Architektur zu präsentieren
- ◆ Stakeholder müssen die Geschäftsziele vorstellen können

* In ATAM werden die Architekturartefakte und Präsentationen einem Review unterzogen

Ergebnisse von ATAM

* ATAM ermittelt

- ◆ Architekturansätze
- ◆ Qualitätsszenarien
- ◆ Qualitätsbaum
- ◆ Risiken
- ◆ *tradeoffs*

* ATAM stellt dar

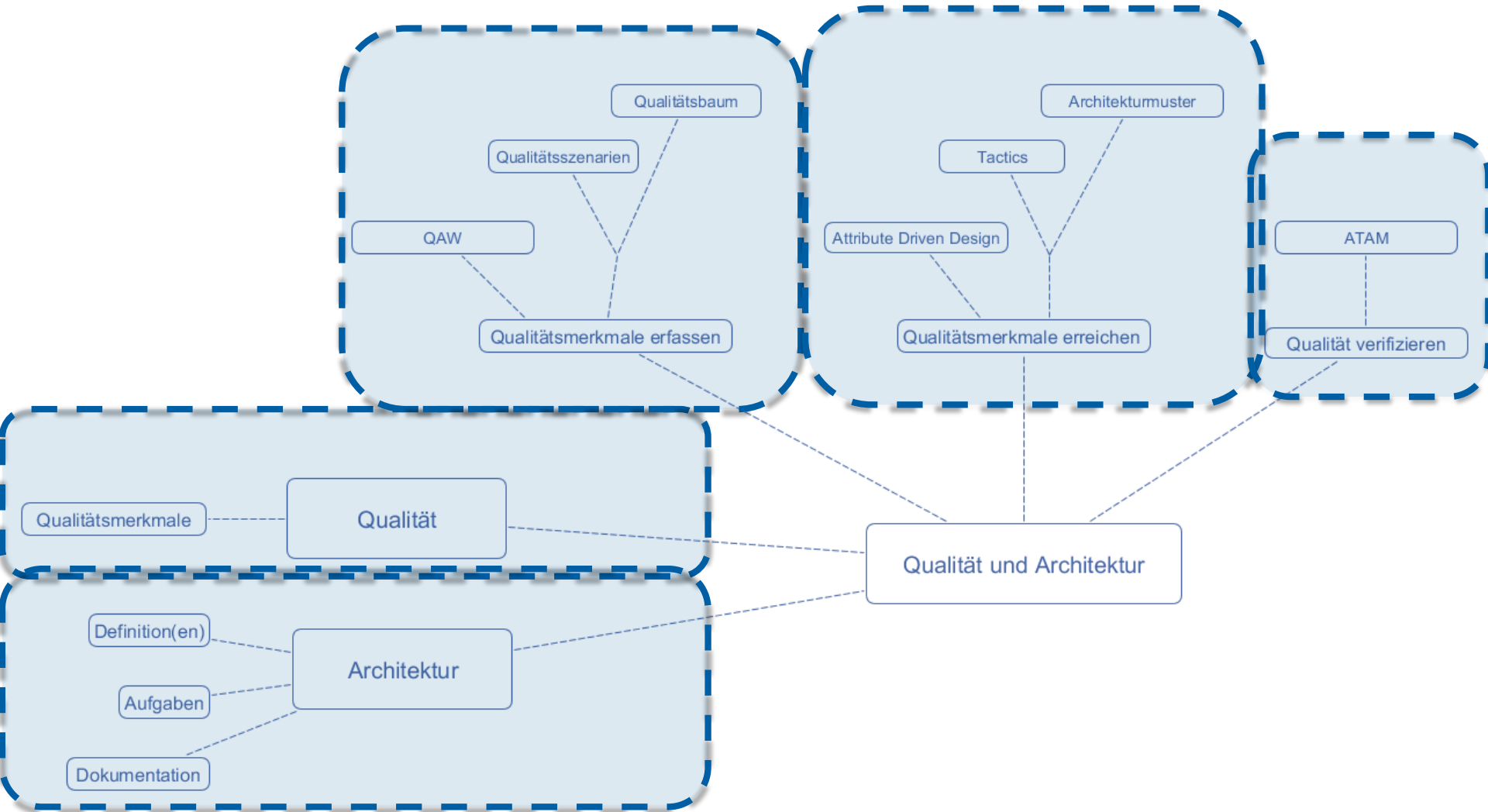
- ◆ inwieweit die vorliegende Architektur die Anforderungen an (Qualitätsmerkmale) erfüllt

Wann wird ATAM eingesetzt?

- ✿ Theoretisch ist der beste Moment für ATAM direkt nachdem die Architektur entworfen wurde
 - ◆ Es gibt keinen oder wenig Code

- ✿ Tatsächlich wird ATAM auch häufig in folgenden Situationen eingesetzt:
 - ◆ Bewertung der Architektur eines bestehenden Systems
 - ◆ Bewertung von Architekturalternativen
 - ◆ Bewertung der Architektur vor großen Updates
 - ◆ Entscheidung, ob ein System neuentwickelt oder angepasst wird
 - Entscheidung *“make or buy”*

Überblick



Qualität als Treiber

Qualitätsmerkmale | Architektur | Qualitätsmerkmale erfassen | Qualitätsmerkmale erreichen |
Qualität verifizieren | **Abschluss**

Für alle vorgestellte Verfahren gilt :

- * Es zeigt eine Idee für die Vorgehensweise
- * Es bringt auch Erfolg, wenn es nicht-formal angewendet werden
 - ◆ Das Ziel dahinter ist wichtig
- * Versuchen Sie es
 - ◆ Es hilft, die Qualitätsmerkmale Ihres Systems besser in den Griff zu bekommen

Architektur ist Schlüssel zur Qualität eines Systems

- * Unterstützen Sie den Architekten in seinem Bemühen um Qualität
 - ◆ Integrieren Sie die Architektur in Ihren Entwicklungsprozess

- * Verorten Sie die (System-)Qualität in der Architektur
 - ◆ Nehmen Sie diese in die Pflicht

- * Es gibt Unterstützung
 - ◆ Qualitätsmerkmale erfassen – Qualitätsszenarios und QAW
 - ◆ Qualitätsmerkmale erreichen – *tactics*, Architekturmuster und ADD
 - ◆ Qualität verifizieren – ATAM

Referenzen

- * [Arc42]
<http://www.arc42.de/>

- * [Brass12]
Software Architecture in Practice (3rd Edition), Brass, Clements, Kazman
Addison-Wesley ISBN-13: 000-0321815734

- * [Buschmann 1996]
Buschmann, F.; Meunier, R.; Rohnert, H.; Sommerlad, P.; & Stal, M. Pattern-Oriented Software Architecture: A System of Patterns. Chichester, NY: Wiley, 1996 (ISBN: 978-0-471-95869-7).

Referenzen

* [FMC]

<http://www.fmc-modeling.org/>

* [IBM14]

Facilitating the Mini-Quality Attributes Workshop

http://resources.sei.cmu.edu/asset_files/Presentation/2014_017_101_89563.pdf

Weiterführende Literatur

- * [Bachmann et al]
Understanding Architectural Patterns in Terms of Tactics and Models
<http://www.sei.cmu.edu/library/abstracts/news-at-sei/architect200708.cfm>
- * [Bachmann et al]
Modifiability Tactics
<http://www.sei.cmu.edu/reports/07tr002.pdf>
- * [Brown]
Software Architecture for Developers – Simon Brown
<https://leanpub.com/software-architecture-for-developers>
- * [Clement10]
Relating Business Goals to Architecturally Significant Requirements for Software Systems
<http://www.sei.cmu.edu/reports/10tn018.pdf>

Weiterführende Literatur

- * [Kazman00]
ATAM :_Method for Architecture Evaluation
<http://resources.sei.cmu.edu/library/asset-view.cfm?assetid=5177>
- * [MS-ArchGuide]
MircoSoft Architecute Guide
<http://apparchguide.codeplex.com/wikipage?title=Chapter%207%20-%20Quality%20Attributes>
- * [Nord09]
A Structured Approach for Reviewing Architecture Documentation
<http://repository.cmu.edu/sei/280>

Weiterführende Literatur

- * [Nothop11]
Identifying Architectural Risks using the Architecture Tradeoff Analysis Method (ATAM)
Linda Nothop OOP 2011
http://www.sigs.de/download/oop_2011/downloads/files/Mi6-4_Northrop_ATAM%20OOP.pdf
- * [STARKE12]
Quality Driven Software Architecture ; Gernot Starke
<https://www.innoq.com/de/articles/2012/04/quality-driven-software-architecture/>
- * [WIKIPEDIA]
<http://de.wikipedia.org/wiki/Softwarearchitektur>

Weiterführende Literatur

✿ [Wojcik 2006]

Wojcik, R.; Bachmann, F.; Bass, L.; Clements, P.; Merson, P.; Nord, R.; Wood, B.
Attribute-Driven Design (ADD), Version 2.0,
<http://www.sei.cmu.edu/library/abstracts/reports/06tr023.c>

✿ [Wood]

A Practical Example for ADD

<http://www.sei.cmu.edu/reports/07tr005.pdf>

WWW.IKS-GMBH.COM



Projekte. Beratung. Spezialisten.



IKS Gesellschaft für Informations-und Kommunikationssysteme mbH

T. +49 2103-5872-0 | www.iks-gmbh.com