

4 Das Requirements-Engineering-Rahmenwerk

Dieses Kapitel gibt Ihnen einen Überblick über unser Requirements-Engineering-Rahmenwerk. Das Rahmenwerk besteht aus:



- *Vier Kontextfacetten: Gegenstands-, Nutzungs-, IT-System- und Entwicklungsfacette*
- *Fünf Requirements-Engineering-Aktivitäten: drei Kernaktivitäten Dokumentation, Gewinnung und Übereinstimmung sowie zwei Querschnittsaktivitäten Validierung und Management*
- *Drei Arten von Anforderungsartefakten: Ziele, Szenarien und lösungsorientierte Anforderungen*

Am Ende des Kapitels vermitteln wir Ihnen die aus dem Rahmenwerk abgeleitete Struktur dieses Buches.

4.1 Überblick über das Rahmenwerk

Jeder Requirements-Engineering-Prozess wird mit dem Ziel begonnen, die aktuelle Realität zu verändern. Auch bei komplexen Vorhaben lässt sich die Essenz der angestrebten Veränderung kurz und prägnant formulieren. Eine solche Formulierung der gewünschten Veränderung bezeichnen wir als „Vision“. Ein prominentes Beispiel ist die Vision von John F. Kennedy aus dem Jahr 1961: „First, I believe that this nation should commit itself to achieving the goal, before this decade is out, of landing a man on the moon and returning him safely to the earth“ (Rede von J. F. Kennedy 1961 in [Duduley 1995]).

*Vision definiert
angestrebte Veränderung*

Visionen können auch kleinere Veränderungen der aktuellen Realität ausdrücken, wie etwa die Integration einer innovativen Funktionalität in ein Multimediasystem oder die Verbesserung der Sicherheit eines Online-Banking-Systems. Eine Vision definiert, was erreicht werden soll, ohne festzulegen, wie die gewünschte Veränderung herbeigeführt werden kann. Mit anderen Worten, die Vision gibt ein Ziel vor, definiert jedoch nicht, wie dieses Ziel konkret erreicht werden soll. In dem Beispiel von J. F. Kennedy definiert die Vision, dass ein Mensch auf den Mond gesandt und sicher zurückgebracht werden soll, aber nicht wie dies zu geschehen hat

*Vision beschreibt ein Ziel,
keine Realisierung*

oder welche konkreten Transportmittel für den Hinflug zum Mond, die Landung auf dem Mond sowie den Rückflug zur Erde verwendet werden sollen.

Vision als Leitgedanke

Die Vision ist kein unerreichbares Wunschbild, sondern ein klar definiertes, überprüfbares und zumeist an eine Frist gebundenes Ziel. Die Vision dient als Leitgedanke für alle an der Entwicklung beteiligten Stakeholder, d. h., die Stakeholder richten ihr Handeln auf die Etablierung der Vision aus. Die in der Vision enthaltenen Informationen sind bei weitem nicht ausreichend, um die Anforderungen an das geplante System in dem erforderlichen Detailgrad zu spezifizieren. Folglich benötigen die Stakeholder zusätzliche Informationen. Quellen zusätzlicher Informationen sind bspw. Kunden, Systemnutzer, Domänenexperten, aber auch Dokumente (z. B. Gesetze, Richtlinien und Standards) und existierende Systeme (z. B. zu ersetzende Altsysteme und Konkurrenzsysteme).

*Etablierung einer Vision
im Kontext*

Wie wir in Teil II erläutern werden, hat jedes System eine Umgebung, die die Anforderungen an das System maßgeblich beeinflusst. Wir bezeichnen diese Umgebung als den „Systemkontext“ (siehe Glossar). Die Vision und ein meist unzureichend bekannter Systemkontext sind somit der Ausgangspunkt des Requirements-Engineering-Prozesses. Das Ziel des Requirements-Engineering-Prozesses kann unter diesen Voraussetzungen als „Etablierung einer Vision im relevanten Systemkontext“ formuliert werden (siehe u. a. [Jarke und Pohl 1993; Pohl 1997]).

Bestandteile des Rahmenwerks

Unser Requirements-Engineering-Rahmenwerk (siehe Abb. 4–1) definiert die strukturellen Bestandteile eines Requirements-Engineering-Prozesses, die zur Schließung der Lücke zwischen der Vision und dem gesetzten Ziel, d. h. der Etablierung der Vision im relevanten Systemkontext, erforderlich sind. Unser Rahmenwerk konsolidiert Erkenntnisse, die ausgehend von Problemstellungen aus der Industrie in Forschungsprojekten gewonnen und in der industriellen Anwendung evaluiert und verfeinert wurden.

Vier Kontextfacetten

Das Rahmenwerk strukturiert den Systemkontext in die vier Kontextfacetten Gegenstandsfacette, Nutzungsfacette, IT-Systemfacette und Entwicklungsfacette (siehe Abschnitt 4.2).

Drei Kernaktivitäten

Ausgehend von der Vision und dem zugehörigen Systemkontext ist das Requirements Engineering im Kern ein iterativ-inkrementeller Prozess, mit drei Kernaktivitäten Dokumentation, Gewinnung und Übereinstimmung (siehe Abschnitt 4.3.1).

Zwei Querschnittsaktivitäten

Die mit dem Requirements-Engineering-Prozess verbundenen Herausforderungen und Risiken begründen zudem die Notwendigkeit der zwei Querschnittsaktivitäten Validierung und Management (siehe Abschnitt 4.3.2).

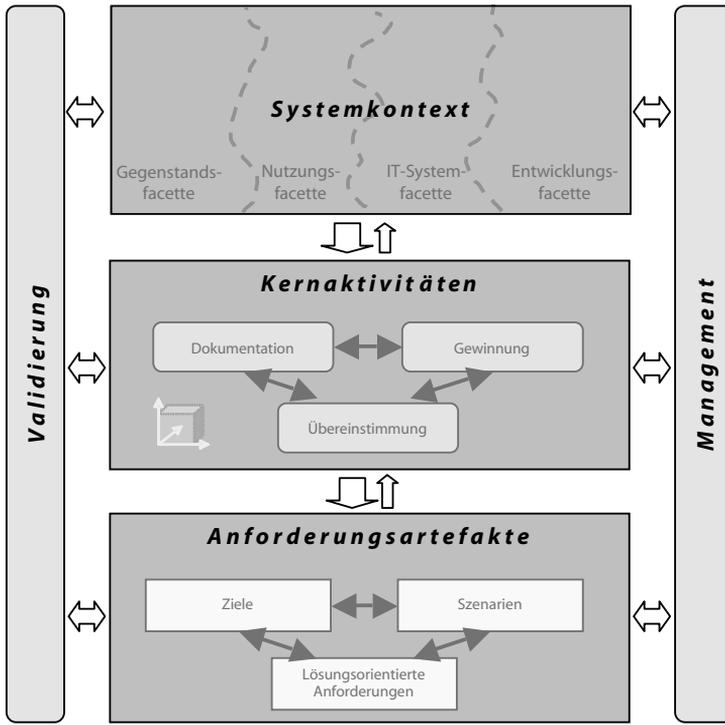


Abb. 4-1
Das Requirements-
Engineering-Rahmenwerk

Wichtige Ergebnisse des Requirements-Engineering-Prozesses sind drei Arten von Anforderungsartefakten, Ziele, Szenarien, lösungsorientierte Anforderungen (siehe Abschnitt 4.4).

*Drei Arten von
Anforderungsartefakten*

In den Abschnitten 4.2 bis 4.4 erläutern wir die einzelnen Bestandteile des Rahmenwerks und ihre Zusammenhänge. In den Teilen II bis VI stellen wir die Grundlagen, Prinzipien und Techniken für die einzelnen Bestandteile des Rahmenwerks im Detail vor. Abschnitt 4.5 enthält einen Überblick, welche Teile des Rahmenwerks in welchen Buchteilen detailliert erläutert werden.

4.2 Vier Kontextfacetten

Der Kontext eines softwareintensiven Systems setzt sich aus einer Vielzahl von Komponenten zusammen. Aufgrund der zunehmenden Vernetzung von Systemen, der steigenden Anzahl von Systemfunktionen, des zunehmenden Qualitätsanspruchs in Verbindung mit immer engeren Zeit- und Kostenvorgaben (siehe Abschnitt 1.1.2) wird es für die Entwickler eines softwareintensiven Systems immer schwieriger, die relevanten Kontextaspekte (d. h. die einzelnen Bestandteile des Systemkontexts; Definition 5-2, Seite 56) vollständig und korrekt zu erfassen bzw. zu berücksichtigen. Fehlende, fehlerhaft erfasste und unzureichend berücksichtigte Kontextas-

*Erfassung und
Berücksichtigung aller
Kontextaspekte*

pekte führen jedoch wiederum zu fehlerhaften und unvollständigen Anforderungen. Die vier Kontextfacetten definieren vier Kategorien von Kontextaspekten, die im Requirements-Engineering-Prozess adäquat berücksichtigt werden müssen¹:

- Gegenstände und Ereignisse der Systemumgebung*

 - **Gegenstandsfacette:** Jedes softwareintensive System bildet Gegenstände und Ereignisse aus seiner Umgebung ab, indem es Informationen über diese Gegenstände und Ereignisse speichert und verarbeitet. Gegenstände müssen nicht notwendigerweise materielle Dinge sein. Beispielsweise bildet eine Softwarekomponente zur Messung der Geschwindigkeit eines Automobils den immateriellen Gegenstand „Geschwindigkeit“ ab. Die Gegenstandsfacette umfasst alle Gegenstände und Ereignisse, die ein System berücksichtigen muss, sowie alle Aspekte, die diese Abbildung beeinflussen, wie z.B. Gesetze oder Standards.
- Systemnutzung*

 - **Nutzungsfacette:** Ein softwareintensives System wird von Menschen oder anderen softwareintensiven Systemen verwendet, um eine bestimmte Aufgabe zu erfüllen und so einen Mehrwert zu realisieren. Die Nutzungsfacette umfasst alle für das geplante System relevanten Aspekte der Systemverwendung, wie bspw. die Nutzungsabläufe, sowie Aspekte, die die Systemverwendung einschränken oder beeinflussen.
- IT-Systemumgebung*

 - **IT-Systemfacette:** Das geplante System wird in ein umgebendes System, d.h. eine IT-Systemumgebung, integriert, die weitere bereits existierende oder geplante softwareintensive Systeme bzw. Hardware- und Softwarekomponenten beinhaltet (bspw. Kommunikationsnetzwerke oder Betriebssysteme). Das geplante System muss also mit anderen Systemen (Hardware und Software) interagieren. Die Eigenschaften und Einschränkungen der IT-Systemumgebung und ihrer Komponenten beeinflussen die Definition der Anforderungen an das geplante System (z.B. Anforderungen an Schnittstellen des geplanten Systems). Die IT-Systemfacette umfasst alle Kontextaspekte, die sich aus der existierenden oder zukünftigen IT-Systemumgebung ergeben, einschließlich der existierenden IT-Strategien.
- Entwicklungsprozessaspekte*

 - **Entwicklungsfacette:** Die Entwicklungsfacette umfasst alle Kontextaspekte, die die Entwicklung des geplanten Systems betreffen bzw. beeinflussen. Hierzu gehören Aspekte des Entwicklungsprozesses, computergestützte Entwicklungswerkzeuge, Reifegradmodelle, Qualitätszertifizierungen, aber auch Aspekte wie Vertrauenswürdigkeit sowie die Sicherheit und die Zuverlässigkeit von softwareintensiven Systemen.

1. Die vier Facetten basieren auf den in [Mylopoulos et al. 1990; Jarke und Pohl 1993] vorgeschlagenen vier Welten des Requirements Engineering.

Beziehungen zwischen den Kontextfacetten

Die Beziehungen zwischen den vier Kontextfacetten lassen sich aus einer abstrakten Betrachtung der Funktionsweise softwareintensiver Systeme ableiten. Jedes softwareintensive System bildet Informationen über einen Gegenstandsbereich ab, z. B. die Geschwindigkeit eines Fahrzeugs oder den Namen eines Kunden. Es verarbeitet diese Informationen gemäß den definierten Funktionen und stellt die Ergebnisse der Verarbeitung in geeigneter Form dem Systemnutzer (Mensch oder System) zur Verfügung. Der Systemnutzer interpretiert die zur Verfügung gestellten Informationen und assoziiert diese wiederum mit der Realität, d. h. dem Gegenstandsbereich. Abbildung 4–2 stellt die Beziehungen zwischen den Kontextfacetten schematisch dar.

Das softwareintensive System selbst ist das Produkt eines Entwicklungsprozesses, dessen Aufgabe es ist, neben den relevanten Aspekten der Entwicklungsfacetten alle relevanten Aspekte der anderen drei Facetten sowie deren Beziehungen entsprechend zu berücksichtigen (siehe Abb. 4–2). Die vier Kontextfacetten sowie deren Beziehungen werden in Teil II weiter detailliert.

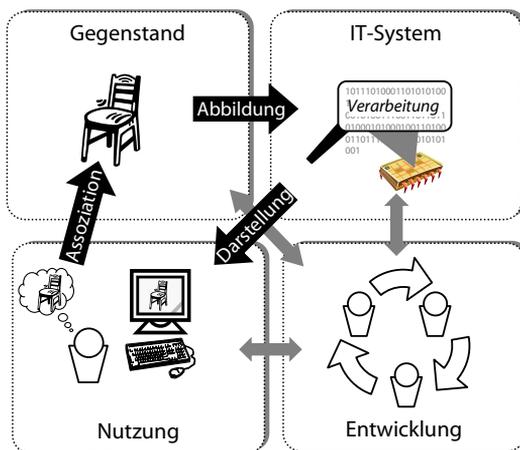


Abb. 4–2

Beziehungen zwischen den vier Kontextfacetten

Verwendung der vier Kontextfacetten

Die Strukturierung des Systemkontexts in die vier Kontextfacetten ist eine Heuristik, die sich u. a. bei der Gewinnung von Anforderungen, der Übereinstimmung über Anforderungen sowie der Validierung von Anforderungen bewährt hat. Die strukturierte Betrachtung des Kontexts anhand der vier Kontextfacetten führt zu einer deutlichen Verbesserung der Qualität einer Anforderungsspezifikation im Hinblick auf die Vollständigkeit und Korrektheit der Anforderungen. Die Kontextstrukturierung ermöglicht eine fokussierte Betrachtung der einzelnen Kontextfacetten und der Bezie-

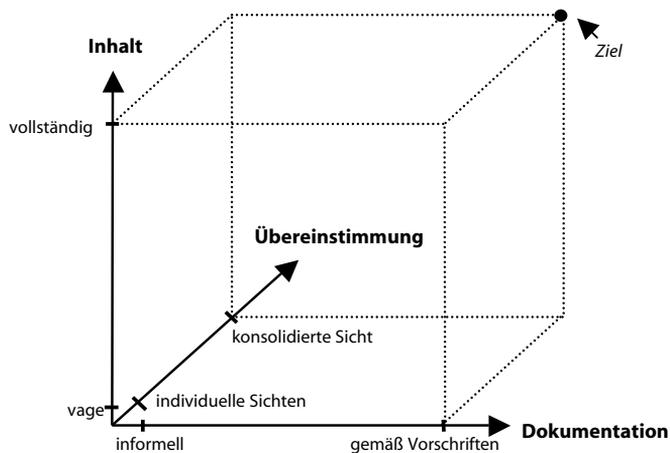
Verbesserte Qualität von Anforderungsdokumenten

hungen zwischen den Kontextfacetten. Unserer Erfahrung nach führt schon die Etablierung von Checklisten für die einzelnen Kontextfacetten zu einer deutlich systematischeren Kontextbetrachtung und in Folge zu einer höheren Qualität der Anforderungsspezifikationen.

4.3 Die fünf Requirements-Engineering-Aktivitäten

In der Entwicklungsfacette findet der Requirements-Engineering-Prozess statt. Die Ziele dieses Prozesses lassen sich anhand der drei Dimensionen des Requirements Engineering charakterisieren (vgl. [Pohl 1993; Pohl 1994]). Die drei Dimensionen sind in Abbildung 4–3 schematisch dargestellt.

Abb. 4–3
Die drei Dimensionen des
Requirements Engineering
(in Anlehnung an
[Pohl 1994])



Im Folgenden charakterisieren wir die drei Dimensionen und definieren das Ziel des Requirements Engineering in jeder Dimension:

*Alle Anforderungen
bekannt und im Detail
verstanden*

- **Inhaltsdimension:** Gegenstand der Inhaltsdimension ist das erreichte inhaltliche Verständnis über die Anforderungen an das geplante System. Zu Beginn eines Requirements-Engineering-Prozesses sind, neben der Vision, typischerweise nur einige der relevanten Anforderungen an das zukünftige System bekannt. Das Verständnis bzw. der Detaillierungsgrad bekannter Anforderungen ist zudem meist sehr vage. Am Ende des Prozesses sollten hingegen möglichst alle Anforderungen bekannt sein und den erforderlichen Detaillierungsgrad aufweisen. Während des Prozesses müssen somit existierende Anforderung explizit gemacht sowie neue Anforderungen kreativ entwickelt werden. Hierdurch ergibt sich das erste essenzielle Teilziel des Requirements-Engineering-Prozesses: Alle relevanten Anforderungen sind in dem erforderlichen Detaillierungsgrad bekannt und verstanden.

■ *Übereinstimmungsdimension:* Gegenstand der Übereinstimmungsdimension ist es, eine Übereinstimmung aller am Prozess beteiligten Stakeholder über die bekannten Anforderungen zu erzielen. Die verschiedenen Stakeholder können über die Anforderungen unterschiedliche Positionen vertreten. Ziel ist es, Konflikte über die bekannten Anforderungen möglichst frühzeitig zu erkennen und diese aufzulösen – sei es durch Konsensfindung oder durch (begründete) Entscheidungen. Werden die Anforderungen an das geplante System definiert, ohne dass eine Konsolidierung der verschiedenen Positionen stattfindet, treten die bis dahin nicht aufgelösten Konflikte spätestens bei der Einführung des Systems auf. Unaufgelöste Konflikte über Anforderungen gefährden die Akzeptanz des Systems und somit die Realisierung der Vision. Das zweite Teilziel des Requirements-Engineering-Prozesses ist daher: die Etablierung einer ausreichenden Übereinstimmung über die bekannten Anforderungen.

Ausreichende
Übereinstimmung

■ *Dokumentationsdimension:* Gegenstand der Dokumentationsdimension ist die Dokumentation und Spezifikation von Anforderungen unter Verwendung verschiedener Formate. Die im Requirements Engineering gewonnenen Informationen werden typischerweise zunächst informell dokumentiert, z.B. in Form von Notizen, Protokollen oder Skizzen. Auf der Grundlage der dokumentierten Informationen werden die Anforderungen gemäß Dokumentationsvorschriften dokumentiert. Dokumentationsvorschriften legen u.a. fest, welche Dokumentationsformate verwendet werden oder welche Schablonen (engl. *templates*) zum Einsatz kommen. Spezifizierte Anforderungen bilden eine Teilmenge der dokumentierten Anforderungen. Sie genügen den Spezifikationsvorschriften, die z.B. die Verwendung von Normsprachen sowie weitere Regeln zur Sicherstellung von Qualitätseigenschaften (z.B. die Konsistenz zwischen verschiedenen Formaten) vorschreiben. Das dritte wesentliche Teilziel des Requirements-Engineering-Prozesses ist somit: die Dokumentation und Spezifikation aller Anforderungen konform zu den vorgegebenen Dokumentations- und Spezifikationsvorschriften.

Anforderungen konform
zu Vorschriften
dokumentiert

Zusammenfassend definieren wir das Ziel des Requirements Engineering daher wie folgt:

Definition 4–1: *Requirements Engineering*

Das Requirements Engineering ist ein kooperativer, iterativer, inkrementeller Prozess, dessen Ziel es ist zu gewährleisten, dass

- (1) alle relevanten Anforderungen bekannt und in dem erforderlichen Detaillierungsgrad verstanden sind,
- (2) die involvierten Stakeholder eine ausreichende Übereinstimmung über die bekannten Anforderungen erzielen,
- (3) alle Anforderungen konform zu den Dokumentationsvorschriften dokumentiert bzw. konform zu den Spezifikationsvorschriften spezifiziert sind.



4.3.1 Die drei Kernaktivitäten

Ableitung der Kernaktivitäten aus den drei Dimensionen

Aus den drei Dimensionen des Requirements Engineering leiten sich die drei Kernaktivitäten des Requirements Engineering ab. Jede Kernaktivität trägt wesentlich zur Erreichung eines der drei Teilziele des Requirements Engineering bei. Die Aktivitäten weisen jedoch auch zahlreiche Wechselwirkungen untereinander auf. Die drei Kernaktivitäten werden im Folgenden erläutert. Die Wechselwirkungen werden in Abschnitt 4.3.3 beschrieben.

Dokumentation

Einhaltung von Dokumentations- und Spezifikationsregeln

Das Ziel der Dokumentationsaktivität ist es, Anforderungen gemäß der Dokumentationsvorschriften zu dokumentieren oder gemäß der Spezifikationsvorschriften zu spezifizieren. Ein Fortschritt auf der Dokumentationsdimension beinhaltet die Dokumentation von Informationen, die im Verlauf des Requirements-Engineering-Prozesses gewonnen oder erarbeitet werden, die Dokumentation von Anforderungen oder die Spezifikation von Anforderungen. Ziel ist es hierbei, die für das Projekt definierten Dokumentations- und Spezifikationsvorschriften zu erfüllen, die sowohl die Dokumentation der Anforderungen während des Prozesses als auch die finale Spezifikation der Anforderungen im Anforderungsdokument regeln.

Geeignete Dokumentation

Im Zentrum steht dabei die Dokumentation von Anforderungen. Es sollten jedoch auch Begründungen, Entscheidungen und weitere wichtige Informationen dokumentiert werden. Abhängig vom jeweiligen Verwendungszweck werden die Anforderungen in geeigneten Dokumentationsformaten festgehalten. Anforderungen können in natürlicher Sprache oder durch Modelle dokumentiert werden. Da verschiedene Stakeholder unterschiedliche Dokumentationsformate bevorzugen und verschiedene (Entwicklungs-)Aktivitäten unterschiedliche Dokumentationsformate erfordern, müssen Anforderungen ggf. von einem Dokumentationsformat in andere Dokumentationsformate überführt werden (z.B. aus einer natürlichsprachlichen Dokumentation in eine modellbasierte Dokumentation). Darüber hinaus muss die Konsistenz zwischen den verschiedenen verwendeten Dokumentationsformaten sichergestellt werden. Die Dokumentationsaktivität wird in Teil IV.a im Detail betrachtet.

Gewinnung

Fortschritt auf der Inhaltsdimension

Das Ziel der Gewinnungsaktivität ist es, einen Fortschritt in der Inhaltsdimension zu erreichen. Ein Fortschritt in der Inhaltsdimension ist gleichbedeutend mit einer Verbesserung des inhaltlichen Verständnisses über die Anforderungen an das geplante System. Im Rahmen der Gewinnungsaktivität werden existierende Anforderungen von Stakeholdern und anderen Anforderungsquellen gewonnen sowie innovative Anforderungen entwickelt.

Da nicht immer offensichtlich ist, welche Anforderungsquellen für das geplante System relevant sind, ist eine gezielte Identifikation von Anforderungsquellen unerlässlich. Existierende Anforderungen werden durch die Befragung von Stakeholdern sowie durch die Analyse von Dokumenten und existierenden Systemen gewonnen. Um innovative Anforderungen zu entwickeln, ist eine bloße Befragung von Stakeholdern bzw. die Analyse von Dokumenten nicht ausreichend. Innovative Anforderungen werden in einem kooperativen, kreativen Prozess mit den Stakeholdern gewonnen. Die Gewinnung innovativer Anforderungen wird durch so genannte Kreativitätstechniken unterstützt. Die Gewinnungsaktivität wird in Teil IV.b detailliert erläutert.

Existierende und innovative Anforderungen

Übereinstimmung

Das geplante System muss die Bedürfnisse und Wünsche verschiedener Stakeholder berücksichtigen. Die Wünsche der Stakeholder an das geplante System müssen derart konsolidiert werden, dass möglichst keine Konflikte mehr existieren und die Anforderungen von möglichst vielen Beteiligten akzeptiert werden. Ziel der Übereinstimmungsaktivität ist es, Konflikte unter den Stakeholdern über die bekannten Anforderungen aufzudecken und aufzulösen. Abhängig von der Ursache eines Konflikts können verschiedene Strategien angewandt werden, um einen Konflikt aufzulösen. Die Übereinstimmungsaktivität wird in Teil IV.c detailliert.

Konsolidierte Stakeholdersicht

4.3.2 Die zwei Querschnittsaktivitäten

Neben den drei Kernaktivitäten des Requirements Engineering existieren zwei Querschnittsaktivitäten, die einen entscheidenden Einfluss auf den Erfolg des Requirements-Engineering-Prozesses haben.

Validierung

Der Querschnittscharakter der Validierung ist durch die folgenden drei Teilaktivitäten der Validierung begründet (siehe Teil V):

- *Validierung der Anforderungsartefakte*: Die Validierung der Anforderungsartefakte hat das Ziel, Fehler in Anforderungen aufzudecken. Nur Anforderungsartefakte mit einer nachweislich hohen Qualität stellen eine geeignete Grundlage für Architekturentwurf, Implementierung und die Entwicklung von Testartefakten dar. Fehler in den Anforderungen haben weitere Fehler in der Architektur, in der Implementierung sowie in Testartefakten zur Folge. Fehler in Anforderungsartefakten sind selten ein reines Dokumentationsproblem. Sie sind oftmals auf ein oder mehrere nicht erreichte Teilziele in den drei Dimensionen zurückzuführen (siehe Abschnitt 4.3.1). Wir definieren daher drei Qualitätstore für die Validierung der Anforderungsartefakte, d. h. ein Tor für die Inhalts-

Validierung der Artefakte

dimension, ein weiteres für die Dokumentationsdimension und ein drittes Tor für die Übereinstimmungsdimension. Jedes Anforderungsartefakt muss alle drei Tore passieren. Erst danach sollte das Artefakt als Referenz für die weitere Entwicklung bzw. als Bestandteil eines Vertrags zwischen Auftraggeber und Auftragnehmer freigegeben werden.

Validierung der
Aktivitäten

■ *Validierung der (Kern-)Aktivitäten:* Die Validierung der Aktivitäten hat das Ziel, die Konformität der durchgeführten Aktivitäten zu Prozessvorgaben und Aktivitätsbeschreibungen zu überprüfen. Zum Beispiel wird geprüft, ob alle definierten Teilaktivitäten tatsächlich durchgeführt wurden.

Validierung der
Kontextbetrachtung

■ *Validierung der Kontextbetrachtung und -berücksichtigung:* Die Validierung der Kontextbetrachtung und Kontextberücksichtigung dient der Überprüfung, ob die relevanten Kontextaspekte der vier Kontextfacetten vollständig und korrekt erfasst sowie zu den entsprechenden Zeitpunkten im Requirements-Engineering-Prozess berücksichtigt wurden. Beispielsweise muss überprüft werden, ob alle geforderten Interaktionen mit Nutzern und anderen Systemen spezifiziert wurden.

Die Ziele, Prinzipien, Techniken und Assistenztechniken der Validierungsaktivität werden in Teil V erläutert.

Management

Das Management im Requirements Engineering umfasst drei Teilaktivitäten:

Management der
Artefakte

■ *Management der Anforderungsartefakte:* Der Gegenstand des Managements der Anforderungsartefakte ist die Verwaltung der Anforderungsartefakte über den gesamten Lebenszyklus des zu entwickelnden Systems hinweg. Das Management umfasst dabei u. a. die Ablage der Anforderungen (z. B. in einer Datenbank), die Priorisierung von Anforderungen, das Konfigurationsmanagement sowie die Gestaltung der Nachvollziehbarkeit und das Änderungsmanagement für Anforderungen.

Management der
Aktivitäten

■ *Management der Aktivitäten:* Das Management der Requirements-Engineering-Aktivitäten zielt auf die Planung, Steuerung und Kontrolle der durchzuführenden Aktivitäten im Requirements Engineering ab, um eine effektive und effiziente Durchführung dieser Aktivitäten sicherzustellen.

Management von
Kontextänderungen

■ *Beobachtung des Systemkontexts:* Das Beobachten des Systemkontexts zielt darauf ab, etwaige Veränderungen des Kontexts zu identifizieren und wenn notwendig zusätzliche Aktivitäten anzustoßen (z. B. Gewinnungsaktivitäten) bzw. die Reihenfolge der durchzuführenden Aktivitäten zu verändern, um die Anforderungen an den veränderten Systemkontext anzupassen.

Die verschiedenen Techniken im Management werden in Teil VI erläutert.

4.3.3 Wechselwirkungen zwischen den Aktivitäten

Die fünf Aktivitäten des Rahmenwerks beeinflussen sich gegenseitig. Fortschritte in einer Dimension können dabei in einer anderen Dimension neue offene To-Dos erzeugen und somit den Fortschritt in einer anderen Dimension mindern. Daher können die einzelnen Aktivitäten im Requirements Engineering für gewöhnlich nicht vollkommen isoliert voneinander durchgeführt werden. Die Wechselwirkungen zwischen Aktivitäten sowie die Auswirkungen der Durchführung einer Aktivität auf verschiedene Dimensionen (Inhalt und/oder Übereinstimmung und/oder Dokumentation) erläutern wir im Folgenden anhand einiger Beispiele.

Beispiel 4–1: Gewinnung zusätzlicher Anforderungen

B

In einer Gewinnungsaktivität werden neue Anforderungen gewonnen, die zunächst informell und nicht gemäß den projektspezifischen Dokumentationsvorschriften dokumentiert werden. Die Gewinnungsaktivität erzielt zwar einen Fortschritt in der Inhaltsdimension, gleichzeitig werden für die Dokumentationsdimension neue To-Dos erzeugt, da die gewonnenen Anforderungen erst durch eine Dokumentationsaktivität in die geforderten Dokumentationsformen überführt werden müssen. Gleichzeitig kann auch ein neues To-Do in der Übereinstimmungsdimension erzeugt werden, falls bspw. die neu gewonnenen Anforderungen nicht von allen Stakeholdern Zustimmung erhalten und somit ein Konflikt über die neuen Anforderungen besteht.

Beispiel 4–2: Aufdeckung einer fehlenden Anforderung

B

Während der Validierung wird festgestellt, dass eine relevante Anforderung fehlt. Die Beteiligten skizzieren die neue Anforderung. Die Skizze entspricht weder den geltenden Dokumentationsvorschriften noch ist die Anforderung mit allen Beteiligten abgestimmt. In diesem Fall führt die Validierung zu einem Fortschritt in der Inhaltsdimension und gleichzeitig zu neuen To-Dos in der Dokumentations- und der Übereinstimmungsdimension.

Beispiel 4–3: Entfernung einer Anforderung aus der Spezifikation

B

Verhandlungen von Kunden und Systemnutzern über eine Anforderung führen aufgrund unterschiedlicher Positionen zu dem Ergebnis, dass die Anforderung nicht Bestandteil der Spezifikation wird. Das Entfernen der Anforderung aus der Spezifikation hat jedoch Auswirkungen auf andere mit der Anforderung verknüpfte Anforderungsartefakte. Diese Artefakte müssen einer Überarbeitung unterzogen werden. Die Auflösung des Konflikts führt zwar zu einem Fortschritt in der Übereinstimmungsdimension, aber zu neuen To-Dos in der Inhalts- und der Dokumentationsdimension, da bspw. durch die Entfernung der Anforderung Inkonsistenzen in anderen Anforderungsartefakten verursacht werden.

4.4 Die drei Arten von Anforderungsartefakten

Mit dem Begriff „Anforderungsartefakt“ bezeichnen wir die dokumentierte Form einer Anforderung (siehe Definition 2–2, Seite 14). Ein Anforderungsartefakt dokumentiert somit das erreichte inhaltliche Verständnis über eine Anforderung unter Verwendung eines bestimmten Dokumenta-

tionsformats. Die verschiedenen Dokumentationsformate für Anforderungen erläutern wir in Teil III. Wir unterscheiden drei Arten von Anforderungen und somit auch drei Arten von Anforderungsartefakten, und zwar Ziele, Szenarien und dokumentierte lösungsorientierte Anforderungen.

Ziele

Intentionen von Stakeholdern Ziele (oder genauer: Zielartefakte) dokumentieren die Intentionen der Stakeholder und abstrahieren dabei sowohl von der Nutzung des Systems als auch von Aspekten der Systemrealisierung. Ziele verfeinern die Vision auf die Ebene einzelner, charakteristischer Merkmale des geplanten Systems. Häufig sind Ziele hierarchisch strukturiert, wobei an der Wurzel der Hierarchie die Vision steht.

Ziele schränken einerseits den Lösungsraum ein, da nur solche Lösungen zulässig sind, die die definierten Ziele bzw. Subziele erfüllen. Andererseits räumen Ziele den Stakeholdern aber auch einen Gestaltungsspielraum bei der Festlegung der detaillierten Anforderungen ein, da sie in der Regel eine Vielzahl von konkreten Lösungen erlauben.

Nutzen von Zielen Die explizite Definition von Zielen unterstützt die Auflösung von Konflikten im Requirements Engineering, führt zu einem besseren Systemverständnis der Beteiligten und fördert somit die Akzeptanz des Systems. Die Verwendung sowie die Dokumentation von Zielen beschreiben wir detailliert in Teil III.a.

Szenarien

Konkrete Beispiele für Zielerfüllung Szenarien beschreiben exemplarisch konkrete Beispiele für Interaktionsfolgen zur Erfüllung (oder Nichterfüllung) eines definierten Ziels bzw. zur Erreichung eines angestrebten Mehrwerts. Szenarien werden u. a. in Form von Use Cases dokumentiert. Szenarien können eine Interaktionsfolge auf unterschiedlichen Abstraktions- bzw. Konkretisierungsgraden definieren. Das Spektrum reicht dabei von Szenarien, deren Inhalte nahe an der Realität sind, bis hin zu Szenarien, die von vielen Aspekten der betrachteten Realität abstrahieren.

Positive Wechselwirkung mit Zielen Ziele und Szenarien verhalten sich komplementär zueinander. Beispielsweise besteht eine Wechselwirkung zwischen der Gewinnung von Zielen und der Gewinnung von Szenarien. Ziele stimulieren die Gewinnung von Szenarien, und Szenarien stimulieren wiederum die Gewinnung von Zielen. Die Eigenschaften von Szenarien, ihre Dokumentation sowie die Wechselwirkung von Szenarien mit Zielen werden in Teil III.b detailliert dargestellt.

Lösungsorientierte Anforderungen

Lösungsorientierte Anforderungen definieren die Daten-/Struktur-, die Funktions- und die Verhaltenssicht eines softwareintensiven Systems. Darüber hinaus beinhalten lösungsorientierte Anforderungen auch Qualitätsanforderungen (siehe Abschnitt 2.2).

Daten, Funktion und Verhalten

Im Gegensatz zu Zielen und Szenarien, die im Wesentlichen unabhängig von einer intendierten Lösung definiert werden können, setzt die Definition von lösungsorientierten Anforderungen die Annahme einer bestimmten angestrebten Lösung voraus (siehe Kapitel 13). Datenmodelle definieren bspw. Entitäten (genauer Entitätstypen), Attribute sowie Beziehungen zwischen Entitäten. Sie legen somit fest, welche Daten und teilweise auch wie diese Daten im System abgebildet werden. Ebenso bedingt die Definition des Systemverhaltens die Festlegung der erlaubten Zustände des Systems. In Teil III.c geben wir einen Überblick über lösungsorientierte Anforderungsmodelle.

Verwendung der drei Arten von Anforderungsartefakten

Die Verwendung von Zielen und Szenarien zusätzlich zu lösungsorientierten Anforderungen hat sich bewährt, um ausgehend von einer Vision detaillierte Anforderungen an das geplante System zu definieren (vgl. bspw. [Jarke und Pohl 1993; Van Lamsweerde 2001; Antón 1996; Antón und Potts 1998; Yu 1997]). Durch ein ziel- und szenariobasiertes Vorgehen wird eine Verbesserung der Qualität der Anforderungsspezifikation erzielt. Beispielsweise verbessert ein ziel- und szenariobasiertes Vorgehen die Vollständigkeit einer Anforderungsspezifikation (vgl. u. a. [Antón und Potts 1998]; siehe auch Abschnitt 7.1). Ziele und Szenarien sind komplementär und ergänzen sich zudem gegenseitig (siehe Kapitel 12).

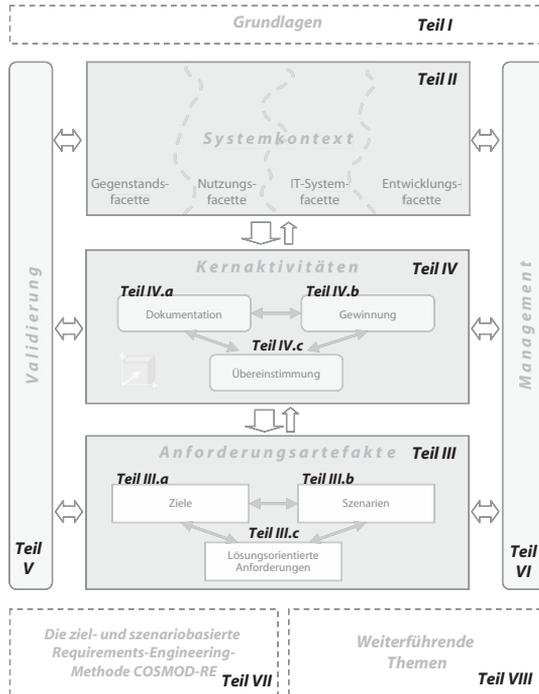
Komplementäre Eigenschaften der Anforderungsartefakte

Durch den inhärenten Kontextbezug von Szenarien bilden diese zudem eine geeignete Basis für die Ableitung von detaillierten lösungsorientierten Anforderungen. Beispielsweise dokumentieren Szenarien explizit, welche Stakeholder das System mit welcher Intention verwenden (siehe Teil III.b für eine detaillierte Betrachtung). Die Verwendung von Zielen und Szenarien zur sukzessiven Kontextverfeinerung und Berücksichtigung wird u. a. bei der Anwendung unserer ziel- und szenariobasierten COSMOD-RE-Methode verdeutlicht (siehe Teil VII).

4.5 Überblick über das Buch

Die Struktur dieses Buches basiert auf unserem Rahmenwerk für das Requirements Engineering. Abbildung 4–4 illustriert, welche Teile des Rahmenwerks in welchen Buchteilen detailliert beschrieben werden.

Abb. 4-4
 Detaillierung des
 Rahmenwerks in diesem
 Buch



- Teil II** ■ Die Strukturierung des Systemkontexts sowie die Bedeutung der vier Kontextfacetten und deren Strukturierung stellen wir Ihnen in Teil II vor.
- Teil III** ■ Die drei Arten von Anforderungsartefakten, deren Dokumentation sowie deren Verwendung erläutern wir in Teil III. Teil III ist in drei Subteile unterteilt. In Teil III.a stellen wir Ziele vor, in Teil III.b Szenarien und in Teil III.c lösungsorientierte Anforderungen.
- Teil IV** ■ Die drei Kernaktivitäten des Requirements Engineering beschreiben wir detailliert in Teil IV, der ebenfalls aus drei Subteilen besteht. In Teil IV.a stellen wir die Dokumentationsaktivität vor, in Teil IV.b die Gewinnungsaktivität und in Teil IV.c die Übereinstimmungsaktivität.
- Teil V und Teil VI** ■ Die beiden Querschnittsaktivitäten werden in den Teilen V und VI vorgestellt. Teil V erläutert die Validierungsaktivität und Teil VI die Managementaktivität.
- Teil VII** ■ In Teil VII stellen wir unsere ziel- und szenariobasierte Requirements-Engineering-Methode COSMOD-RE vor. Zudem illustrieren wir die Verwendung von COSMOD-RE anhand eines Beispiels.
- Teil VIII** ■ Teil VIII beschließt dieses Buch mit vier weiterführenden Themen:
- Anforderungsbasiertes Testen
 - Requirements Engineering und CMMI
 - Requirements Engineering in der Produktlinienentwicklung
 - Ziel- und szenariobasierte Werkzeugauswahl