

## 2 Methodendesign

In diesem Kapitel wird das Method Engineering als wissenschaftliche Erkenntnismethode für den Entwurf eines neuen Artefakts (Österle et al. 2010, S. 5) dargestellt und kritisch betrachtet. Im Speziellen soll diese als Rahmen für das Design einer Methode eingesetzt werden. DRECHSLER ET AL. (2010, S. 18) stellen in ihrer Ausarbeitung fest, dass COCKBURN (2003) in seinen Ausführungen zum Design agiler Methoden bereits implizit einige systemische Aspekte berücksichtigt. Aufgrund dessen und da keine weiteren konkreten Ansätze zum Methodendesign in der aktuellen Literatur identifiziert werden können, werden folgend die Ansätze von COCKBURN (2003) und HIGHSMITH (2002) als zwei Autoren des agilen Manifests vorgestellt. Deren konkrete Anforderungen an eine zu designende agile Methode werden dargestellt und erläutert. Zudem werden diese um zusätzliche Anforderungen erweitert. IT-Projekte zeichnen sich unter anderem durch eine dynamische Umwelt und neben direkt beobachtbaren Einflussfaktoren oft auch durch sogenannte Soft Factors, die nicht direkt beobachtbar sind, aus. Diese spezifischen Eigenschaften stellen weitere Anforderungen an die Projektmanagementmethode dar.

### 2.1 Method Engineering

Method Engineering ist die Disziplin, nach der anhand von Engineering-Praktiken neue Methoden für Konstruktion, Abwägung, Evaluation und Management zur Erstellung von **Information Systems Development Methods (ISDM)**<sup>3</sup> erforscht werden (Rolland 2007, S. 6).

Somit kann das Method Engineering als Meta-Methode angesehen werden, die bei der Auswahl und Integration von Teilstücken von ISDMs hilft, um daraus eine situationsspezifische Methode zu erstellen (Rolland 2007, S. 6).

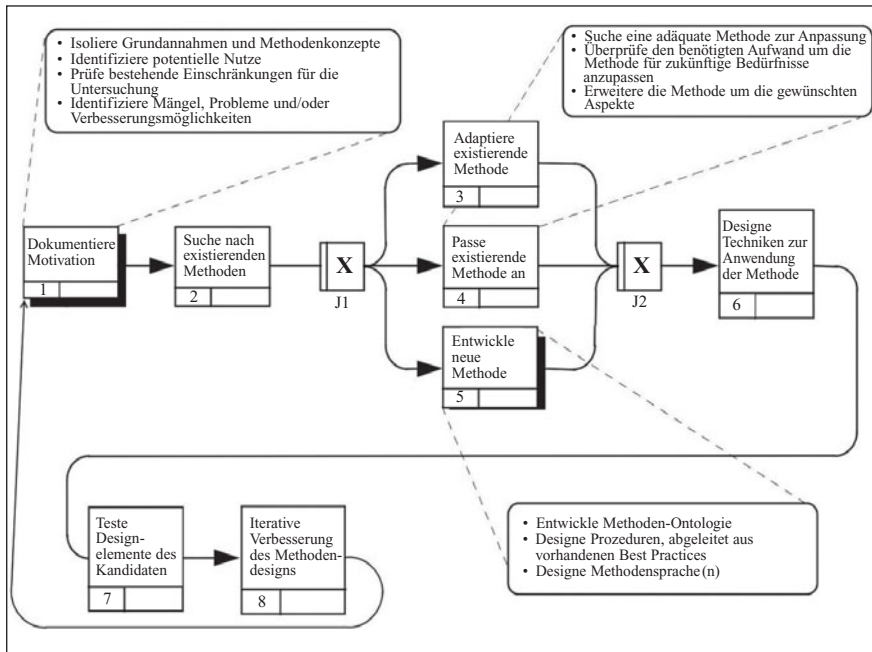
Das Vorgehen des Method Engineering zur Entwicklung einer Methode wird in Abbildung 4 (s. S. 10) in Form eines IDEF 3-Modell dargestellt<sup>4</sup>.

Ausgangspunkt der Methodenentwicklung ist die **Dokumentation der Motivation** (1). Dies umfasst das Ausfindig machen von grundlegenden Ahnungen und Methodenkonzepten sowie die Identifizierung potenzieller Nutzer. Ebenso werden existierende Einschränkungen bei der Ausführung der Analyse untersucht sowie Mängel,

---

<sup>3</sup> Eine erweiterte Auseinandersetzung mit ISDMs liefern Mihailescu und Mihailescu (2010).

<sup>4</sup> Eine Beschreibung der einzelnen Notationselemente ist in Anhang A zu finden.



**Abbildung 4:** Vorgehen des Method Engineering

Quelle: In Anlehnung an Mayer et al. 1995, S. 7–8

nicht benötigte Aspekte aber auch mögliche Verbesserungen identifiziert. Im nächsten Schritt werden mögliche **bereits existierende Methoden gesucht und identifiziert** (2). Nun wird entschieden, ob eine vorhandene Methode direkt unter der Auswahl ihrer relevanten Bestandteile adaptiert werden kann (**Adaption** hier: Method Tailoring) (3), eine vorhandene Methode situationsspezifisch angepasst werden muss (**Anpassung**, hier: Situational Method Engineering) (4) oder aber eine vollkommen neue Methode erstellt werden muss (**Neuerstellung**) (5) (Mayer et al. 1995, S. 7–10). Bei der Adaption wird meist implizit eine vorhandene Methode für die kontextspezifischen Gegebenheiten eines Projektes adaptiert (Patel et al. 2004, S. 1). Bei der Anpassung wird eine vorhandene Methode gesucht, der Anpassungsaufwand festgestellt und anschließend diese um die gewünschten Bestandteile ergänzt (Patel et al. 2004, S. 2). Bei der **Neuerstellung** wird in einem ersten Schritt eine Methoden-Ontologie erstellt. Darauf folgend werden Prozeduren aus vorhandenen Best Practices abgeleitet und letztendlich ein oder mehrere Methodensprachen erstellt. Im nächsten Schritt werden **Praktiken zur Anwendung der Methode design** (6). Darauf folgend wer-

den die **Designelemente der aktuellen Version getestet und evaluiert** (7). Danach erfolgt die **iterative Verbesserung des Methodendesigns**<sup>5</sup> (8) (Mayer et al. 1995, S. 7–10). Im Folgenden werden das Method Tailoring und das Situational Method Engineering genauer betrachtet, da im Rahmen dieser Arbeit die Anpassung einer vorhandenen Methode angestrebt wird. Die komplette Neuerstellung einer Methode ist in diesem Fall keine Option, da von den Vorzügen des vorhandenen Wissens und Erfahrungen im Rahmen einer Wiederverwendung vorhandener Methoden Gebrauch gemacht werden soll (Patel et al. 2004, S. 2). Auch MAYER ET AL. (1995, S. 8) stellen fest, dass eine komplette Neuerstellung erst dann zur Anwendung kommen sollte, wenn sowohl das Method Tailoring als auch das Situational Method Engineering ausgeschlossen wurden. Sie bezeichnen die anderen beiden Vorgehensweisen, wie auch HIGHSMITH (2002, S. 350–354), als wesentlich attraktiver und einfacher. Vor allem in Kapitel 6 wird dargestellt, dass keine Methode existiert, die alle Anforderungen (vgl. Kap. 2.3) erfüllt.

### 2.1.1 Method Tailoring

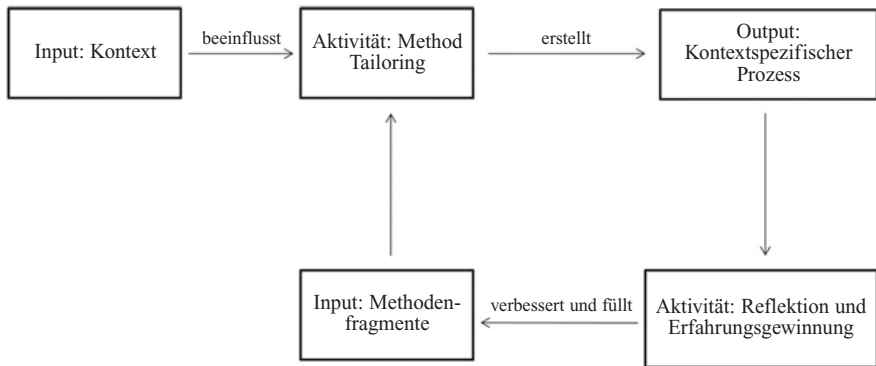
Im Folgenden wird das Method Tailoring (MT) als ein möglicher Ansatz im Rahmen des Method Engineering betrachtet, der im Weiteren die Modellbildung unterstützen und fundieren kann. Es wird herausgestellt, inwiefern eine Eignung als Rahmenwerk für die Erstellung einer neuen agil-systemischen Methode vorhanden ist. Während bei klassischen Softwareentwicklungsmethoden oftmals in der Literatur davon ausgegangen wird, dass sie universell übertragbar sind, trifft dies in der Praxis oft nicht zu (Fitzgerald, Russo und O’Kane 2000, S. 1; Patel et al. 2004, S. 1). Diesem Problem wird durch den Einbezug des Kontexts in dem hier betrachteten Ansatz entgegengewirkt.

Bisweilen findet die Anpassung von Methoden meist implizit während deren Nutzung statt (Patel et al. 2004, S. 3). Dies kann zu Problemen führen. Zum einen ist es schwer festzustellen, wer welche Anpassungen vorgenommen hat und welche Konsequenzen diese mit sich bringen, zum anderen werden die gesammelten Erfahrungen, die sich durch die Anpassung ergeben, nicht dokumentiert und geteilt, so dass diese nur schwierig wiederverwendet werden können (Patel et al. 2004, S. 1–3). Lediglich derjenige, der die Anpassung selbst durchgeführt hat, hat auch Zugriff auf die dabei erzielte Erfahrung. Eine nicht rigide sondern flexible Methode ist nach PATEL ET AL. (2004, S. 2) die Voraussetzung dafür, dass Method Tailoring effektiv und erfolgreich umgesetzt werden kann. Ein Beispiel hierfür ist der Rational Unified

---

<sup>5</sup> Anm. des Autors: Der hier dargestellte Prozess erfolgt iterativ. Um das iterative Vorgehen graphisch besser zu verdeutlichen wurde in der ursprünglichen Grafik nach Mayer et al. (1995, S. 7–8) ein Prozessrückschritt in Form eines Pfeils hinzugefügt, der von Schritt 8 ausgehend zum Anfang des Prozesses verläuft.

Process (RUP), der keinen Prozess vorschreibt, sondern ein Menge von Prozesskomponenten definiert, die unter Berücksichtigung eines Projektes ausgewählt werden können. Eine im Rahmen des Method Tailoring erstellte Methode benötigt somit eine wohldefinierte Menge von frei wählbaren Prozesskomponenten sowie explizite Vorgaben zur Anwendung dieser (Patel et al. 2004, S. 2–3). Der Prozess des Method Tailoring findet beeinflusst vom vorhandenen Kontext statt. Im Rahmen von Method Tailoring wird, abgeleitet von vorhandenen Methodenfragmenten, ein kontextspezifischer Prozess erstellt. Durch Reflexion und Erfahrungsgewinn während des Einsatzes des kontextspezifischen Prozesses wiederum ergeben sich neue und verbesserte Methodenfragmente, mit denen die vorhandenen ergänzt werden. Abbildung 5 visualisiert dieses Vorgehen.



**Abbildung 5:** Vorgehen des Method Tailoring

Quelle: In Anlehnung an Patel et al. 2004, S. 6

In ihren Ausführungen betrachten PATEL ET AL. (2004, S. 4–13) vier unterschiedliche Rahmenwerke zur Erstellung von Methoden unter Berücksichtigung des Method Tailoring. Sie stellen fest, dass zurzeit nur wenige Rahmenwerke vorhanden sind und diese bislang nicht empirisch belegt sind (Patel et al. 2004, S. 5). Trotzdem scheint der vorgestellte Ansatz durch die vorhandenen Freiheitsgrade in der Gestaltung von Methoden Relevanz für die Praxis zu besitzen (Fitzgerald, Russo und O’Kane 2000, S. 1; Patel et al. 2004, S. 1). Es erscheint sinnvoll einem definierten Prozess zu folgen und gewonnene Erkenntnisse zu bewahren, anstatt implizit Anpassungen während des Methodeneinsatzes vorzunehmen und mögliche Erfahrungen nicht für die Zukunft festzuhalten. Ein wichtiger Aspekt in Bezug auf die spätere Methodenerstellung ist zudem, dass eine Vielzahl von agilen Methoden als geeignete Ausgangsmethode im Rahmen des Method Tailoring verwendet werden können, da diese bereits eine Vielzahl von Problemen klassischer Methoden beheben (Abrahamsson et

al. 2003, S. 6–7; Aydin et al. 2004, S. 127). Eine genauere Betrachtung dieser Methoden, sowie deren konkrete Eignung als Ausgangsmethode für die Erstellung einer neuen agil-systemischen Methode, findet in Kapitel 5 statt.

### 2.1.2 *Situational Method Engineering*

Als weiterer möglicher Ansatz des Method Engineering wird das Situational Method Engineering (SME) betrachtet. In diesem Kontext wird ebenfalls aufgezeigt, ob sich das Situational Method Engineering als Rahmenwerk für die Erstellung einer neuen agil-systemischen Methode eignet. HARMSSEN (1996, S. 28) definiert das Situational Method Engineering wie folgt:

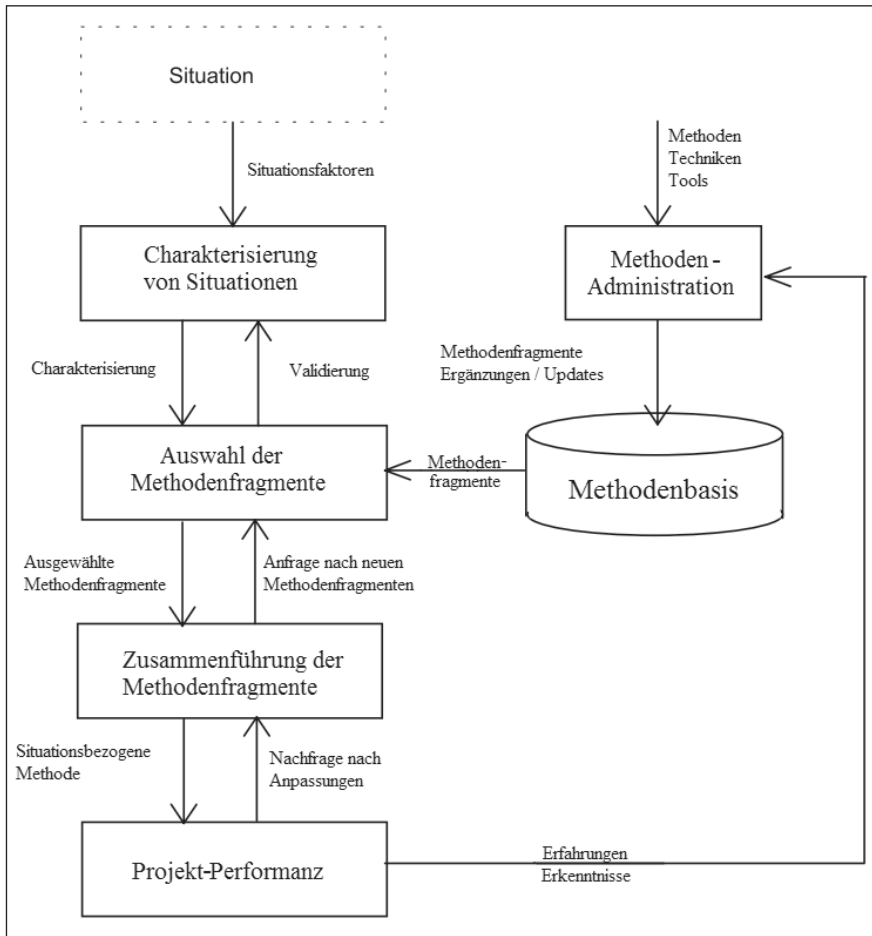
“Situational Method Engineering is the sub-area of Method Engineering directed towards the controlled, formal and computer-assisted construction of situational methods out of method fragments.”

Die Notwendigkeit von Situational Method Engineering ergibt sich aus der Komplexität von Softwareprojekten. Eine Methode allein ist oftmals nicht für alle Situationen ausreichend geeignet (Harmsen 1996, S. 5–6). HARMSSEN stellt die Konstruktion von situationsbezogenen Methoden aufbauend aus ausgewählten Methodenfragmenten als eine mögliche Lösung dar.

Methodenfragmente sind die Schlüsselemente einer Methodenbasis. Diese können zum einen konzeptueller, zum anderen technischer Natur sein. Die konzeptuellen Methodenfragmente betrachten die Produkt- und Prozessspezifikationen, während die technischen Methodenfragmente die unterstützenden Werkzeuge betrachten (Harmsen 1996, S. 52–56). Neben den Methodenfragmenten existieren noch weitere Konzepte, die für die vollständige Abbildung einer Methodenbasis benötigt werden. Dies sind die Erstellung von Verbindungen zwischen Instanzen von Produktfragmenten, Regeln zur Einschränkung der Anwendung von Methodenfragmenten, Produktrollen, die wiedergeben, welche Rolle ein Produktfragmenten in einer Assoziation spielt, Prozessrollen, die darstellen wie ein Prozessfragment ein Produktfragmenten beeinflusst, sowie Teilnehmer des Projektes und deren spezifische Rollen in Bezug auf Produktfragmente (Harmsen 1996, S. 56–58).

Die einzelnen Methodenfragmente, ihre Beziehungen, Eigenschaften und Einschränkungen sind im Rahmen einer Methodenbasis digital verfügbar (Harmsen 1996, S. 52). Die Methodenbasis selbst deckt konzeptuelle, technische und menschliche Aspekte ab. Hierbei gelten Praktiken, Vorgehensschritte, Werkzeuge, konkrete Rollen und Produktbeschreibungen als Instanzen dieser Aspekte. Diese werden nicht in der eigentlichen Methodenbasis hinterlegt, sondern in einem Repository auf das die Methodenbasis referenziert (Harmsen 1996, S. 51–52).

Abbildung 6 gibt eine Übersicht über einen möglichen Prozess einer solchen situationsbezogenen Methodenbildung. Der hier aufgezeigte Prozess umfasst alle rele-



**Abbildung 6:** Der Prozess des Situational Method Engineering

Quelle: In Anlehnung an Harmsen 1996, S. 31

vanten Aspekte, ist aber aufgrund der hohen Dynamik in der Projektumgebung lediglich als ein Beispiel zu sehen (Harmsen 1996, S. 32).

Zentraler Punkt bei der situationsbezogenen Methodenbildung ist die **Methodenbasis**. Diese beinhaltet sämtliche Methodenfragmente, die für die situationsbezogene Methodenerstellung zur Verfügung stehen. Initial gefüllt und im Späteren auch ergänzt wird diese Basis durch die **Methoden-Administration**. Diese wählt aus be-

kannten Methoden, Praktiken und Tools die relevanten aus und fügt sie der Methodenbasis hinzu. Hierbei spielen auch Erfahrungen und Erkenntnisse aus aktuellen sowie vollendeten Projekten eine Rolle. Der Hauptprozess der situationsbezogenen Methodenerstellung findet im Rahmen einer **Situation** statt. Die eine Situation bestimmenden Situationsfaktoren werden in einem ersten Schritt **charakterisiert**, worauf hin eine **Auswahl der relevanten Methodenfragmenten** stattfindet. Rückkoppelnd findet in diesem Schritt ebenfalls die Validierung der Charakterisierung statt. Als Nächstes werden die ausgewählten **Methodenfragmente zusammengeführt** und je nach Bedarf weitere angefragt. Die nun vorhandene situationsbezogene Methode wird im Projekt angewandt und die **Projekt-Performanz** wird gemessen. Hieraus resultiert die Nachfrage nach möglichen Anpassungen, die durch eine Rückkopplung an den Schritt der Zusammenführung geleitet wird. Ebenfalls ergeben sich in diesem Schritt der bereits genannte Output von Erfahrungen und Erkenntnissen aus dem getätigten Projekt, die wiederum in die Methoden-Administration eingehen (Harmsen 1996, S. 31–32).

### *2.1.3 Kritische Betrachtung des Method Tailoring und des Situational Method Engineering*

Im Vergleich des Situational Method Engineering mit dem Method Tailoring fällt die Ähnlichkeit der beiden Ansätze auf. Das Situational Method Engineering als situativer Ansatz unterscheidet sich lediglich in seiner konkreteren Ausformulierung, dem Umstand, dass anstelle des Rückbezugs auf den Kontext an dieser Stelle die Situation als auslösender Faktor gesehen wird und der expliziten Forderung des Situational Method Engineering nach einer digitalen Unterstützung des Prozesses, vom Method Tailoring. Auch in der Literatur werden beide Methoden nicht strikt voneinander getrennt. So spricht HARMSSEN (1996) in seiner Arbeit durchgängig vom Situational Method Engineering wird aber von PATEL ET AL. (2004) im Rahmen des Method Tailoring zitiert. Offensichtlich ist auch in der einschlägigen Literatur eine eindeutige begriffliche Abgrenzung nicht möglich bzw. nicht vorhanden. Aufgrund dessen wird auch im Rahmen dieser Arbeit nicht zwischen den beiden Ansätzen unterschieden und die Bezeichnungen Situational Method Engineering und Method Tailoring synonym verwendet.

Der Fokus dieser Arbeit liegt auf der Erstellung einer neuen agil-systemischen Methode zum Management von Softwareprojekten, die nicht nur theoretisch fundiert, sondern vor allem auch in der Praxis umsetzbar sein soll. Zudem sollen die beiden aufgestellten Hypothesen bestätigt bzw. widerlegt werden. Kritisch am Situational Method Engineering zu betrachten ist die geringe empirische Fundierung des Prozesses (Patel et al. 2004, S. 5). Trotzdem erscheint die grundlegende Idee für Softwareprojekte als durchaus geeignet. In Software-Entwicklungsprojekten angewende-

te Methoden werden selten strikt übernommen, sondern oft dem Projekt implizit angepasst. Dieses Vorgehen führt häufig zu nur schwer vorhersehbaren Ergebnissen und begünstigt den Verlust wichtiger Erkenntnisse, die bei der Methodenanpassung erzielt werden, da diese bei einem impliziten Vorgehen in der Regel nicht dokumentiert werden (Patel et al. 2004, S. 1). Positiv zu bewerten ist die explizite Berücksichtigung des situativen Kontextes eines Projektes im Situational Method Engineering. Jedes Projekt zeichnet sich durch seine Einzigartigkeit aus. Die Einflussfaktoren für den Erfolg eines Projektes und damit der situative Kontext unterscheiden sich von Projekt zu Projekt (Harmsen 1996, S. 5–6). Der von HARMSSEN (1996, S. 31) vorgeschlagene Prozess (vgl. Abbildung 6) bietet hierfür anhand eines flexiblen Vorgehens, einer Methodenbasis und eines expliziten Schrittes zur Sicherung von Erkenntnissen eine angemessene Lösung. Systemisches und agiles Projektmanagement gehen von einer turbulenten, nicht durch klassische Methoden zu beherrschenden Umwelt aus (vgl. Kap. 4.2.1 und Kap. 5.1.2) (Highsmith 2002, S. 310–311). Um der dynamischen und komplexen Struktur solcher Projekte Rechnung zu tragen, erscheinen die Erkenntnisse des Method Tailoring bzw. des Situational Method Engineering als geeignetes Rahmenwerk für die Erstellung einer neuen agil-systemischen Methode. Im Rahmen dieser Arbeit wird in einem ersten Schritt lediglich das Repository für die Methodenbasis mit möglichen Praktiken, Vorgehensschritten, Werkzeugen, konkreten Rollen und Produktbeschreibungen gefüllt (vgl. Kap. 2.1.2). Die Erstellung einer umfangreichen Methodenbasis ist aufgrund des zeitlich beschränkten Rahmens dieser Ausarbeitung nicht möglich. Um eine passende Methode als Grundlage für das Repository zu identifizieren, werden im folgenden Kapitel Anforderungen an eine solche aufgestellt. Daraufhin werden Methoden aus unterschiedlichen Projektmanagementbereichen auf ihre Eignung als Ausgangsmethode anhand des aufgestellten Anforderungskatalogs betrachtet. Anschließend wird die am besten geeignetste Ausgangsmethode ausgewählt und entsprechend angepasst.

## 2.2 Methodendesign nach COCKBURN und HIGHSMITH

Als konkrete Ansätze zum Design einer Methode werden im Folgenden Abschnitt die Ansätze nach COCKBURN (2003 S. 157–230) und HIGHSMITH (2002 S. 321–389) vorgestellt. Beide Autoren haben entscheidend zur Entstehung des agilen Manifests beigetragen. Die Erkenntnisse der beiden Autoren sind sehr eng miteinander verknüpft und beide referenzieren stellenweise aufeinander. Im Hinblick auf die im späteren Verlauf der Arbeit zu erstellende agil-systemische Methode stellt sich ein Methodendesign, das bereits agile Grundzüge aufweist sowie bereits implizit systemische Aspekte berücksichtigt (Drechsler, Kalvelage und Trepper 2010, S. 145), als besonders geeignet dar. Zudem findet sich in der betrachteten Literatur kein Methodendesign,



das insbesondere in einer solch ausgeprägten Form konkrete zu erfüllende Anforderungen an die zu erstellende Methode transparent darstellt und beschreibt.

HIGHSMITH (2002, S. 321) stellt in seinen Ausführungen fest, dass die Bezeichnung Methode für das Design einer Vorgehensweise im Projektmanagement nicht ausreichend ist, da diese lediglich auf das gesammelte Wissen über erfolgreiche Praktiken und Prozesse referenziert. Vielmehr geht er von der Erstellung eines Umweltsystems aus, das neben den einzusetzenden Praktiken auch den Grund für ihren Einsatz mit einbezieht. Hieraus folgert er drei kritische Fragen, die im Rahmen der Suche nach einem geeigneten Methodendesign geklärt werden sollten. Was sind die strategischen Möglichkeiten der Problemdomäne? Wie sieht die Organisationskultur aus? Kann die Problemdomäne in der bestehenden Kultur funktionieren? Somit wird implizit der Forderung des Situational Method Engineering, bezogen auf die Berücksichtigung der Situation, Rechnung getragen (vgl. Kap. 2.1.2). In Bezug auf die Auswahl einer geeigneten Methode stellt HIGHSMITH (2002, S. 332) fest, dass diese erste einmal unter Rücksichtnahme auf die Kultur erfolgen soll und erst danach unter Rücksichtnahme auf das Projekt. Sollte an dieser Stelle eine Inkompatibilität zwischen den ausgewählten Methoden auftreten fordert er dazu auf, die für beide Bereiche am besten geeignete Methode entsprechend zu modifizieren. Das Design oder die Adaption einer agilen Methode, bzw. die Methode selbst, sollte seiner Ansicht nach auf den sieben im Folgenden beschriebenen Merkmalen basieren.

### 1. Eine realistische Erwartung an die Methode

Hiermit ist die Einstellung der Organisation gegenüber der zu verwendenden Methode gemeint. Eine Methode dient in erster Linie zur Unterstützung der Durchführung einer Aufgabe. Ihr Einsatz sollte immer mit angemessenen und realistischen Erwartungen einher gehen. Sie erlaubt beispielsweise durch verschiedene Praktiken eine grobe Approximation zukünftiger Ereignisse, kann allerdings niemals dazu dienen, genaue Vorhersagen zu machen (Highsmith 2002, S. 336–337). COCKBURN (2003, S. 189–195) nennt in seinen Ausführungen außerdem einige kritische Aspekte, die während des Methodendesigns beachtet werden sollten. Eine Methode kann nicht auf alle Projektarten und -größen übertragen werden (vgl. Kap 2.1.2). Das Methodendesign sollte selbstkritisch vorgenommen werden, insbesondere weil die Erfahrungen und Werte des Designers Einfluss auf das Design haben. Eine Methode sollte nur notwendige Elemente enthalten und nicht unnötig ausgeschmückt werden. Zudem sollte eine Methode im Hinblick auf deren Anwendbarkeit in der Praxis evaluiert werden.

### 2. Eine klare Definition der Elemente der Methode inklusive konkreter Praktiken

Eine Methode besteht laut HIGHSMITH (2002, S. 338) aus unterschiedlichsten Elementen. Er verweist an dieser Stelle auf COCKBURN. Nach dessen Auffassung besteht

die Struktur einer Methode aus den folgenden dreizehn Bausteinen: Prozesse, Meilensteine, Aktivitäten, Teams, Produkte, Praktiken, Rollen, Standards, Werkzeuge, Qualität, Fertigkeiten, Persönlichkeit und Teamwerte (Cockburn 2003, S. 159). HIGHSMITH (2002, S. 338) trennt und ergänzt diese Bausteine durch die Perspektive auf das Ökosystem des Teams, welche COCKBURN nur teilweise berücksichtigt. Bereits von COCKBURN berücksichtigte Merkmale, die HIGHSMITH nennt, sind Fertigkeiten, Persönlichkeit und Teamwerte. Von COCKBURN nicht berücksichtigte Merkmale des Ökosystems sind Einstellung, Individuen und kollaborative Praktiken. Praktiken im Besonderen unterteilt HIGHSMITH (2002, S. 339) in die drei Kategorien: Kollaboration, Projektmanagement und Softwareentwicklung. Außerdem stellt er fest, dass für eine agile Methode mindestens in den sechs Schlüsselbereichen Projektvorbereitung, Projektbeginn, Entwicklungszyklus<sup>6</sup>, Feedback, Kundenvertrauen und technische Exzellenz Praktiken vorhanden sein müssen. Schlüsselcharakteristika von Methoden sind nach HIGHSMITH (2002, S. 339) Größe, Anzahl von Ritualen, Gewicht als Produkt aus Größe und Ritualen sowie Toleranz. COCKBURN (2003, S. 168) ergänzt diese Schlüsselcharakteristika um Problemgröße, Projektgröße, Labilität des Systems, Präzision, Genauigkeit, Relevanz, Sichtbarkeit, Maßstab und Stabilität<sup>7</sup>.

### 3. Eine effektive Sammlung von Designprinzipien

HIGHSMITH (2002, S. 342) nennt neun Prinzipien, die im Rahmen einer Methodenerstellung erfüllt werden müssen, wobei alle in engem Zusammenhang mit den Inhalten des agilen Manifests (vgl. Kap. 5.1.2) stehen und teilweise die dort genannten Leitsätze und Prinzipien wiedergeben. Sieben dieser Prinzipien stammen von COCKBURN (2003, S. 199), der diese auch im Rahmen seiner Crystal Methods voraussetzt. Zwei weitere stammen von HIGHSMITH selbst. Die Prinzipien lauten:

- Interaktive, direkte Kommunikation ist der günstigste und schnellste Kanal zum Austausch von Informationen.
- Übermäßiges Gewicht in der Methode ist kostspielig.
- Größere Teams benötigen schwerere Methoden.
- Eine höhere Anzahl von Ritualen eignet sich insbesondere für kritischere Projekte.
- Eine höhere Anzahl an Methodenritualen kann im Sinne von stärkeren Regulationen, Sicherheit und Rechtsbelangen nötig sein.
- Ein erhöhtes Maß an Feedbacks und Kommunikation reduziert das Bedürfnis nach Zwischenergebnissen.
- Disziplin, Fertigkeiten und Verständnis stehen Verfahren, Formalitäten und Dokumentation gegenüber.

---

<sup>6</sup> Dieser sollte kurz, iterativ und funktionsgetrieben sein und feste Zeitvorgaben besitzen.

<sup>7</sup> Eine ausführliche Darstellung dieser Merkmale liefert Cockburn (2003, S. 168–181).

- In Engpasssituation ist Effizienz entbehrlich.
- Vermeide das Entstehen von Engpässen durch effiziente, fließende Prozesse<sup>8</sup> (Cockburn 2003, S. 199; Highsmith 2002, S. 342–343).

4. Ein phasenbezogenes Rahmenwerk, Projekt Templates und Entwicklungsszenarien  
HIGHSMITH fordert (2002, S. 345–347), dass ein Methodendesign ein sehr abstrakt gehaltenes Rahmenwerk beinhalten muss, das auf die einzelnen Projektphasen eingeht, ohne dabei eine strikte Struktur vorzugeben. Außerdem benötigt es einige vorgefertigte Templates, die als Beispiele für die Zusammensetzung von Methodenfragmenten dienen (Highsmith 2002, S. 347) sowie konkrete Szenarien, die sich als Beispiele für den Einsatz der in der Methode enthaltenen Methodenfragmente und deren Zusammenspiel eignen (Highsmith 2002, S. 347–350).

#### 5. Eine Serie von kollaborativen Designschritten

HIGHSMITH (2002, S. 350) nennt drei allgemeine Schritte zur Methodenerstellung:

- Artikulation von Werten und Prinzipien.
- Ausarbeitung der Projektziele und Projektcharakteristika.
- Erstellung eines Methodenrahmenwerks von Templates und Szenarien.

Diese Schritte werden nicht einmalig nacheinander, sondern iterativ durchgeführt. Werte und Prinzipien werden ausgebildet. Projektziele sowie die Projektcharakteristika Teamgröße, Criticality<sup>9</sup>, Risiko und Unsicherheit sowie der Methodenfokus werden erarbeitet und das bereits im vierten Merkmal genannte Rahmenwerk mit Templates und Szenarien erstellt. Hierbei stellt er fest, dass der einfachste Weg die Wiederverwendung eines anderen Ansatzes ist, auf dem als Template aufgebaut werden kann (Highsmith 2002, S. 350–354).

#### 6. Prozesse zur Methodenanpassung

Vor der Erstellung einer Methode muss darüber nachgedacht werden wie die Anpassung der Methode erfolgt. Beispielhaft nennt HIGHSMITH (2002, S. 354–357) folgendes Vorgehen:

- Auswahl eines domänenspezifischen Templates mit einem Minimum an Praktiken und Szenarien.
- Austausch von Praktiken und Szenarien, die besser für das Projektteam geeignet sind.

---

<sup>8</sup> Eine ausführliche Betrachtung der Prinzipien findet im Rahmen der Crystal Methods in Kapitel 5.2.3 statt.

<sup>9</sup> Mit Criticality ist das „Kritisch-sein“ eines Projektes gemeint.

- Vorsichtige und zurückhaltende Ergänzung weiterer benötigter Praktiken und Szenarien.
- Evaluation jeder zukünftigen Praktik gegenüber den vorhandenen Prinzipien.

Außerdem weist er darauf hin, dass eine Anpassung der Methode auch im weiteren Verlauf des Projektes je nach Bedarf möglich und durchaus sinnvoll ist.

## 7. Richtlinien für die Übertragung auf große Projekte

HIGHSMITH (2002, S. 357–364) fordert die Anpassbarkeit der Methode auf große Projekte. Als große Projekte definiert er solche, die 50 und mehr Personen umfassen. Er fordert eine Balance zwischen Agilität und Striktheit, die der Projektgröße gerecht wird. Hierbei spielen Kollaboration und eine Architektur, ausgerichtet auf die Weitergabe von Wissen, eine entscheidende Rolle.

Fasst man die genannten Anforderungen im Hinblick auf die Suche nach einer geeigneten Methode für ein Repository im Rahmen des Method Engineering sowie bezogen auf die Überprüfung der zweiten Hypothese zusammen, lässt sich die Erfüllung dieser Anforderungen anhand folgender Fragestellungen überprüfen.

1. Existiert für die betrachtete Methode eine Domänenabgrenzung?
2. Sind die Methodenelemente und Praktiken klar definiert?
3. Werden die geforderten Designprinzipien erfüllt?
4. Existieren ein phasenbezogenes Rahmenwerk, Projekt Templates und Entwicklungsszenarien?
5. Enthält die Methode Werte und Prinzipien? Sind Projektziele und Projektcharakteristika definiert?
6. Existieren Prozesse zur Methodenanpassung?
7. Existieren Richtlinien für die Übertragung auf große Projekte?

### 2.3 Erweiterung der im Methodendesign nach COCKBURN und HIGHSMITH definierten Anforderungen

Im Folgenden wird der Anforderungskatalog an eine Ausgangsmethode nach COCKBURN und HIGHSMITH um weitere Forderungen ergänzt, die insbesondere im Rahmen von IT-Projekten große Relevanz besitzen.

So existieren nach der Auffassung von HOLZBAUER (2007, S. 95–96) drei Klassen von „Einflussfaktoren erfolgreicher Projektarbeit“. Er unterscheidet zwischen sogenannten **Hard Factors**, **Soft Factors** und **externen Faktoren**. Hierbei stehen vor allem die Soft Factors als besonders ausschlaggebend für den Erfolg eines Projektes im Vordergrund (Fiedler 2005, S. 8). Häufig sind es nicht die technischen Gegebenheiten, die ein Projekt verzögern oder scheitern lassen, sondern vielmehr zwischen-

Agil-systemisches Softwareprojektmanagement

Trepper, T.

2012, XVII, 163 S. 22 Abb., Softcover

ISBN: 978-3-8349-4201-2