



Niklas Spitzczok von Brisinski
 (E-Mail: nspitzczok@adesso.de)
 ist Managing Consultant bei der adesso AG in München. Er ist vom PMI zertifizierter Project Management Professional (PMP) und Certified Scrum Product Owner (CSPO) der Scrum Alliance. Bei adesso ist er verantwortlich für die Projektmanagementberatung.

PRAGMATISCHES IT-PROJEKTMANAGEMENT: SOFTWAREENTWICKLUNGS- PROJEKTE MIT DEM **PMBOK® GUIDE FÜHREN**

In Softwareentwicklungsprojekten muss man das Spannungsfeld zwischen Anforderungen, Technologie, Risiken und Kosten erfolgreich meistern. Eine hohe Dynamik kann sich beispielsweise durch veränderte Anforderungen, unerkannte Risiken oder andere ungeplante Ereignisse ergeben. Die Verwendung einer Projektmanagement-Methodik schützt zwar nicht vor solchen Vorkommnissen, lässt sie aber leichter bewältigen. Dieser Artikel erläutert den Einsatz des *PMBOK®* Guide in Softwareprojekten und zeigt, wie Unternehmen mit wenig Aufwand Projekte wesentlich methodischer realisieren können.

Auch wenn es seit einigen Jahren eine Vielzahl von Vorgehens- und Prozessmodellen für die Softwareentwicklung gibt, die Fehlentwicklungen konstruktiv entgegenwirken sollen, geraten Projekte doch immer wieder in Schieflage. Zwar garantiert allein der Einsatz einer Projektmethodik noch nicht den Projekterfolg, er kann aber die Erfolgswahrscheinlichkeit signifikant verbessern. Aber wie sieht die passende Methode aus?

Dazu ist es hilfreich, sowohl neuere Entwicklungen, wie z.B. Agilität in der Softwareentwicklung, zu betrachten, als auch bewährte internationale Methoden aus dem Bereich des Projektmanagements einzubeziehen. Hierbei fällt der Blick schnell auf den bewährten „Project Management Body of Knowledge“ des Project Management Institutes (PMI), kurz *PMBOK®* Guide (sprich: „Pimbuk Gaid“).

Der *PMBOK®* Guide

Der *PMBOK®* Guide besteht aus einer Sammlung von „Best Practices“, legt dabei den Fokus nicht auf Softwareentwicklungsprojekte und ist so auch nicht software-spezifisch ausgeprägt. Er ist generisch konzipiert und kann theoretisch auch beim Industriebau oder in Organisationsprojekten eingesetzt werden. Folglich lässt er sich nicht ohne Anpassungen in konkreten Softwareentwicklungsprojekten anwenden. Zudem ist der Guide auf den anglo-amerikanischen Raum ausgerichtet, sodass er auch aus kulturellen und rechtlichen Erwägungen heraus nur mittelbar

für Softwareentwicklungsprojekte aus dem deutschsprachigen bzw. mitteleuropäischen Raum geeignet ist.

Der zugehörige *Project Management Professional (PMP)* ist das weltweit am weitesten verbreitete Projektmanagement-Zertifikat. Im Rahmen eines vierstündigen, computergestützten Multiple-Choice-Tests mit 200 Fragen werden die Inhalte des *PMBOK®* Guide abgefragt. Wer den Test bestanden hat, darf anschließend drei Jahre lang den Titel PMP führen. Auch wenn nicht jeder Zertifikatsinhaber automatisch ein brillanter Projektmanager ist – und umgekehrt ein brillanter Projektmanager nicht zwangsläufig ein Zertifikat benötigt: Der PMP belegt ein gewisses Grund-

verständnis seines Inhabers auf dem Gebiet des Projektmanagements.

PMBOK® Guide, PMP und alle damit verbundenen Aktivitäten haben maßgeblich den Erfolg des PMI bestimmt, sodass heute ein einheitlicher Standard die globale Kooperation in Projekten zwischen Unternehmen und Kulturen vereinfacht. Das PMI ist in Größe und Ausrichtung damit weltweit konkurrenzlos. **Abbildung 1** zeigt die Entwicklung der Anzahl zertifizierter PMPs.

Um den *PMBOK®* Guide jedoch auch in Softwareentwicklungsprojekten unmittelbar einsetzen zu können, ist neben einer entsprechenden Adaption die praktische Erprobung in mittleren und großen Soft-

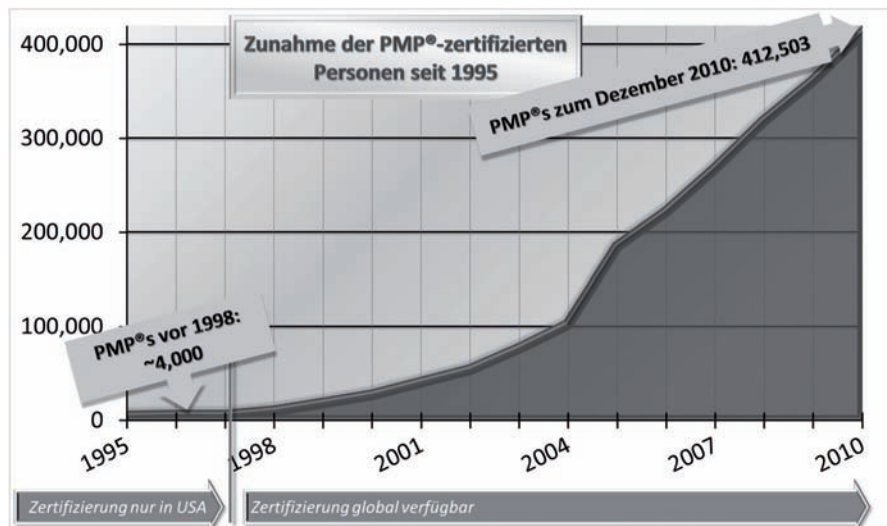


Abb. 1: Anzahl vergebener PMP-Zertifikate von 1995 bis 2010 (Quelle: Oliver Lehmann, PMI Munich Chapter e.V.).

warentwicklungsprojekten nötig. Der schlichte Einsatz des *PMBOK*® Guide als „Maß aller Dinge“ in der täglichen Arbeit ist nicht hilfreich.

Softwareentwicklung mit dem *PMBOK*® Guide

Softwareentwicklungsprojekte sind meist bestimmt durch Technologieeinsatz, Komplexität und Kommunikationsaufwand. Zudem unterliegen sie kontinuierlichen Änderungen während des Entwicklungsprozesses. Anders als bei Projekten mit einem materiellen Produkt, sind die Konsequenzen einer Softwareänderung für Außenstehende häufig nicht nachvollziehbar. Daher ist es verwunderlich, dass das PMI als Spezialisierung für besondere Branchen bisher nur Erweiterungen (so genannte *Extensions*) zum *PMBOK*® Guide für die Domänen *Construction* (vgl. [PMI07]) und *Government* (vgl. [PMI06]) herausgegeben hat. Diese Erweiterungen, die bisher zudem nur zur alten „*PMBOK* Guide Edition 3“ (vgl. [PMI04]) geschrieben wurden und für die „Edition 4“ (vgl. [PMI08]) noch nicht vorliegen, sind außerhalb der USA weitgehend nutzlos. Im Gegensatz dazu könnte eine Softwareentwicklungs-Erweiterung zumindest global ausgerichtet sein.

Teile des *PMBOK*® Guide finden sich aber in verwandten Konzepten wieder. Als eines der Unternehmen mit den meisten PMP-zertifizierten Mitarbeitern weltweit vertritt IBM mit dem *Rational Unified Process* (*RUP*) eine sehr mächtige Softwareentwicklungsmethode, die wesentliche Teile und Prozesse enthält, die auch Bestandteil des *PMBOK*® Guide sind. Meist wird der *RUP* jedoch vor allem als Softwareprozess und weniger als Projektmethode eingesetzt. Der Softwareprozess wird als angepasstes Vorgehensmodell bei der „ingenieurmäßigen“ Anwendungsentwicklung verstanden und dient dazu, die Komplexität der Softwareentwicklung beherrschbar zu machen. Eine Projektmethode hingegen betrachtet die Softwareentwicklung als im Rahmen des Projektmanagements zu integrierendes Wissensgebiet. **Abbildung 2** zeigt die Verwandtschaft der Kernprozesse im *RUP* und *PMBOK*® Guide.

Was fehlt im *PMBOK*® Guide?

Der unmittelbare Einsatz des *PMBOK*® Guide in der Softwareentwicklung ist nicht zielführend. Erst ein davon abgeleitetes Vorgehensmodell mit explizit beschriebenen

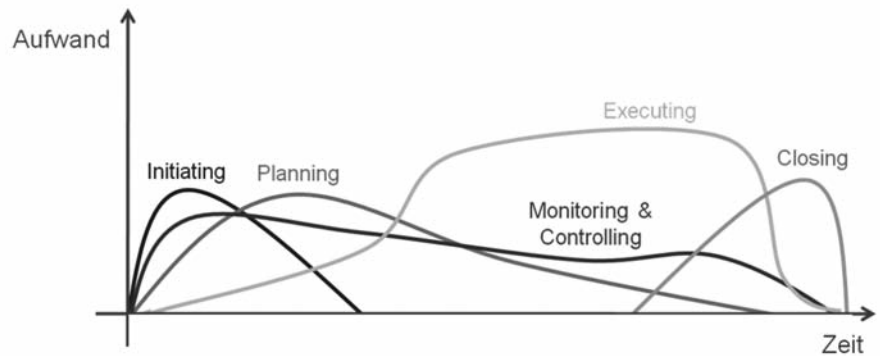
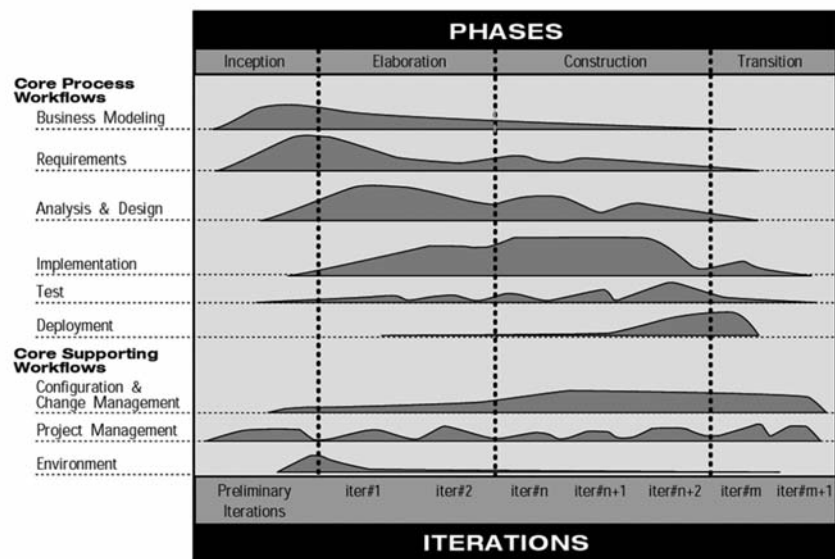


Abb. 2: Vergleich der Prozessgruppen im *Rational Unified Process* (oben, Quelle: IBM) und im *PMBOK*® Guide (unten, eigene Darstellung).

nen Prozessen und Vorlagen ermöglicht dessen Verwendung. So erweisen sich die im PMI-Verlag publizierten Vorlagen im „*Book of Forms*“ (vgl. [Sta09]) in der Praxis häufig als zu sperrig und sind für Softwareprojekte nur mit enormen Anpassungen nutzbar.

Zudem wird im *PMBOK* Guide kein explizites Phasenmodell festgelegt: Die Prozesse sind in Gruppen organisiert, die gemäß dem Guide – je nach Projektphase – mit unterschiedlicher Intensität durchgeführt werden. Was genau eine Phase in einem Softwareprojekt ausmacht und aus welchen Phasen ein Projekt besteht, wird nicht explizit definiert. Das ist zwar im Zusammenhang mit dem restlichen Guide gesehen stringent, erschwert aber auch die direkte projektbezogene Anwendung.

Darüber hinaus fehlen dem *PMBOK*® Guide einige Themenbereiche, die in Softwareprojekten existenzielle Bedeutung haben. Die schwerwiegendste Schwachstelle ist das Anforderungsmanagement,

das zwar im Rahmen der vierten Edition verbessert wurde, jedoch dem in Softwareprojekten erforderlichen Umgang mit Anforderungen bei Weitem nicht gerecht wird. Zudem unterstützt der *PMBOK*® Guide nur mittelbar die Prozesse der Softwareentwicklung, da ihm die dazu notwendigen Bestandteile fehlen.

Für hiesige Softwareentwicklungsprojekte wäre daher eine einfache Adaption des *PMBOK*® Guide für den deutschsprachigen Raum hilfreich. Dazu ist auch ein Seitenblick auf agile Projektmethoden erforderlich, da diese mittlerweile die Projektkultur sowie die Erwartungshaltung der Auftraggeber beeinflussen.

Agilität und *PMBOK*® Guide – ein Widerspruch?

Agile Projektmethoden berücksichtigen stärker als klassische Methoden, dass die fachliche und technische Durchdringung nicht mit dem Projektstart beendet ist, sondern dort häufig erst ihren verwertbaren



Anfang nimmt. Außerdem tragen sie der Erkenntnis Rechnung, dass die Vorstellungskraft der Fachbereiche für das Endprodukt ebenso endlich ist wie die Vorhersagbarkeit technischer Komplikationen im Rahmen der Entwicklung. Das soll kein Freibrief sein, um „einfach mal loszulegen“. Denken und Planen sind auch in agilen Projekten die zwei wichtigsten Tugenden.

Vergleicht man zum Beispiel Scrum mit dem *PMBOK*® Guide, wird schnell deutlich, dass die Rahmenbedingungen sehr unterschiedlich sind. Während Scrum exakte Angaben zu den Zeiträumen, Artefakten und Phasen eines Projekts macht, also sehr spezifisch ist und eher einer Checkliste bzw. einem Fahrplan gleicht, ist der *PMBOK*® Guide systembedingt relativ vage. Ein direkter Vergleich ist daher nur bedingt möglich und sinnvoll – letztlich ist der Großteil der Prozesse auf die eine oder andere Weise auch im Guide beschrieben. Nur wurden die Prozesse im *PMBOK*® Guide nicht unter agilen Aspekten betrachtet, sodass der Kontext völlig fehlt. Bemerkenswert ist, dass der Begriff „agil“ in der aktuellen vierten Ausgabe des englischsprachigen *PMBOK*® Guide nicht vorkommt. Dennoch springt nun das PMI mit auf den „agilen Zug“ und überraschte im Februar 2011 mit der Nachricht, noch im laufenden Jahr eine agile Zertifizierung anzubieten. Es bleibt abzuwarten, ob in Ermangelung agiler Ansätze im *PMBOK*® Guide das neue Zertifikat einen praktischen Wert hat.

Agile Vorgehensweisen unterliegen einem gänzlich anderen Paradigma als klassische Ansätze: Inhalt und Umfang werden nicht initial festgelegt und dann sukzessive umgesetzt, sondern von einem Sprint zum nächsten geplant (vgl. [Ble08]). Es gibt zahlreiche Projekttypen, bei denen dieses Vorgehen wesentlich sinnvoller als die klassische Projektplanung ist. Was aber, wenn rein agiles Projektmanagement nicht eingesetzt werden kann, z.B. in Festpreis-Projekten?

Wo agile Projekte nicht funktionieren, aber dennoch sich ändernden Anforderungen Rechnung getragen werden muss (was an sich in fast jedem Softwareprojekt der Fall sein dürfte), ist eine iterative Vorgehensweise empfehlenswert. Hier wird zu Beginn klassisch verfahren, z.B. mit der Anlage eines Projektauftrags, der Projektumfangsbeschreibung und der möglichst vollständigen Spezifikation der Anforderungen.

Wie setzt sich eine solide Methodik zusammen?

- *PMBOK*® Guide als Basis
- Integration von softwareentwicklungsspezifischem Anforderungsmanagement
- Wissensgebiet Softwareentwicklungsmanagement mit eigenen Prozessen
- Sprachlich klare und einfache Prozesse
- Grafisch gestaltete Prozesse (z. B. in der BPMN)
- Phasenmodell mit Kontrollpunkten oder sinnvolle Kombinationen agiler Ansätze (das muss nicht unbedingt Scrum sein)
- Controlling als wichtigster Steuerungsprozess
- Direkt einsetzbare Vorlagen

Kasten 1: Bestandteile einer soliden IT-Projektmanagement-Methodik.

derungen. Nach der Schätzung sind Zeiträumen und Aufwände weitgehend bekannt. Änderungen können dann über ein *Change-Request*-Verfahren eingesteuert werden, das in der Anwendung nicht komplexer als beispielsweise die Führung des in Scrum enthaltenen *Product Backlog* sein muss. Das alles sieht der *PMBOK*® Guide nicht explizit vor.

Lösungsansatz: *PMBOK*® Guide agil machen

Ein Vorgehensmodell zum Projektmanagement (siehe **Kasten 1**) ist in naturgemäß risikobehafteten Technologieprojekten unerlässlich. Häufig scheitert der Einsatz jedoch an der Komplexität und Praktikabilität der Methodik. Eine zuverlässige Vorgehensweise zeichnet sich mindestens durch folgende Kriterien aus:

- Das Vorgehen sollte auf sicheren und anerkannten Verfahren des Projektmanagements basieren. Der *PMBOK*® Guide bietet sich hier an, da er die gelebte Praxis aus vielen Projekten zusammenführt.
- Im Rahmen der Wissensgebiete *Scope Management* sollte dem Anforderungsmanagement besonderes Augenmerk geschenkt werden. In Softwareentwicklungsprojekten haben die Aufnahme und Verfolgung der Anforderungen das größte Fehlerpotenzial. Die Prozesse und Werkzeuge des *PMBOK*® Guide

reichen nicht aus, um dieses Wissensgebiet strukturiert und nachhaltig zu gestalten.

- Ein Wissensgebiet *Softwareentwicklung* mit ihren Prozessen und Werkzeugen sollte nahtlos ergänzt und integriert werden. Dabei ist darauf zu achten, dass die Prozesse zwar so offen gestaltet werden, dass sie technologieunabhängig sind, aber dennoch spezifisch genug, um sie ohne zu großen Anpassungsaufwand anwenden zu können.
- Aufgrund der Sammlung der „Best Practices“ enthält der Guide auch etliche für Softwareentwicklungsprojekte nicht direkt relevante Prozesse, Methoden und Werkzeuge. Daher ist es hilfreich, eine vereinfachte Interpretation des *PMBOK*® Guide mit klarem Bezug auf die Softwareentwicklung zu erstellen. Dabei sollte darauf geachtet werden, die Wissensgebiete und Prozesse auch für PMI- und *PMBOK*® Guide-Unerfahrene anwendbar zu machen.
- Ein wichtiges Element in diesem Zusammenhang ist die grafische Gestaltung der Prozesse. Der *PMBOK*® Guide bedient sich einer proprietären Darstellungsweise, wohingegen Notationen wie die *Unified Modeling Language (UML)* (vgl. [OMG-b]) oder die *Business Process Modeling Notation (BPMN)* (vgl. [OMG-a]) mittlerweile weit verbreitet und besser lesbar sind.
- Systembedingt fehlt dem *PMBOK*® Guide ein explizites Phasenmodell. Zwar läuft jedes Projekt anders ab, dennoch sollte ein Phasenmodell als Basis vorgeschlagen werden. In der Projektkonfiguration können die Phasen angepasst werden. Um Folgefehler zu vermeiden, sollten Kontrollpunkte Bestandteil des Phasenmodells sein, die nur passiert werden dürfen, wenn die in ihnen definierten Kriterien erfüllt wurden. Ist das nicht der Fall, wird die Folgephase faktisch ohne Genehmigung begonnen. Das kann zwar gut gehen, doch sollen die vorab definierten Kriterien ja einen gewissen Qualitätsstandard sichern. Das ist nachvollziehbar nur möglich, wenn auch tatsächlich alle vorab definierten Kriterien erfüllt wurden.
- Zumindest für den deutschen Wirtschaftsraum ist eine sprachliche und inhaltliche Adaption der Vorgehens- ▶

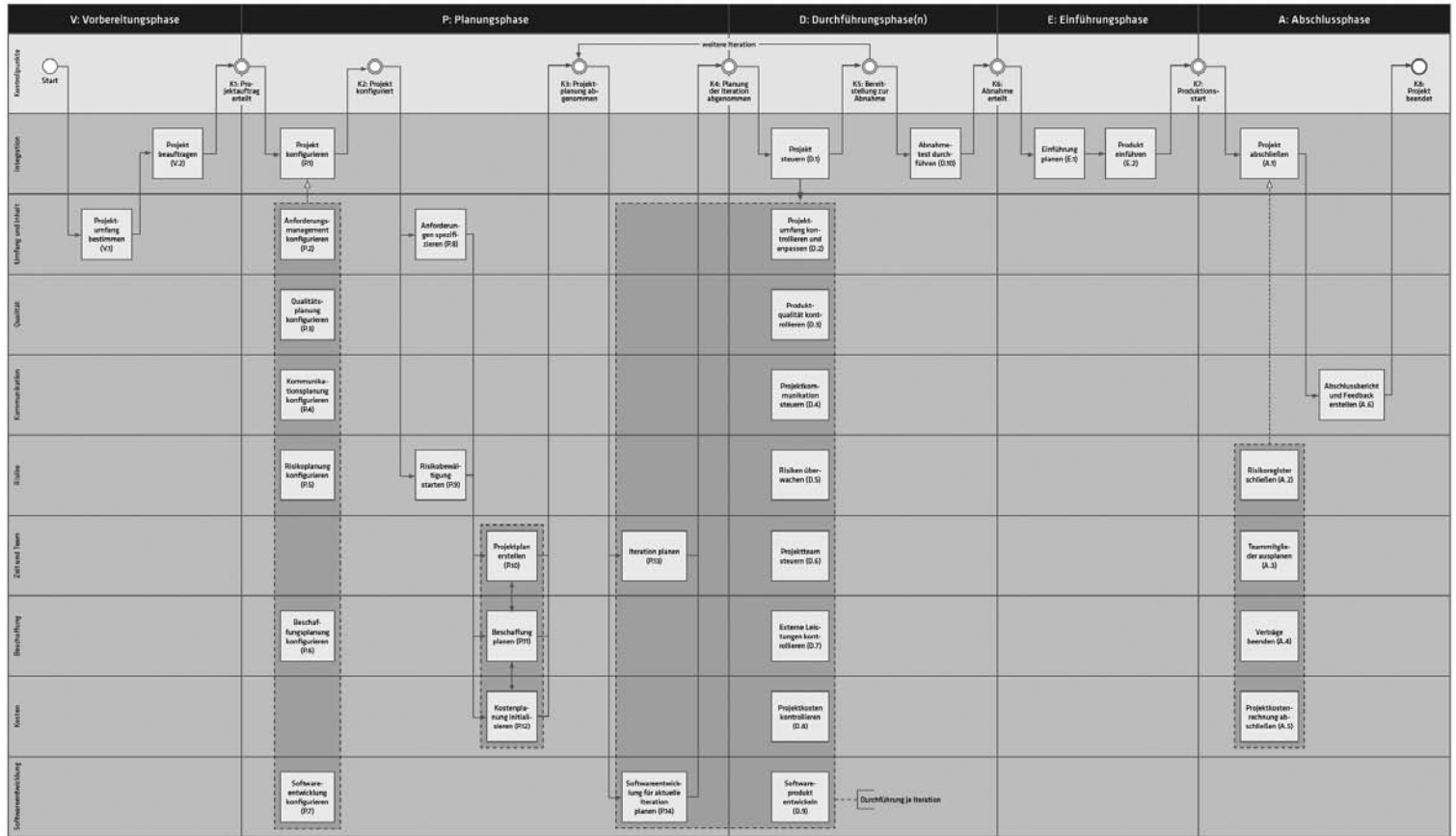


Abb. 3: Phasen, Wissensgebiete und Prozesse im Überblick.

weise erforderlich. So sind z. B. etliche der im *PMBOK*® Guide aufgeführten Vertragsformen irrelevant.

■ Aufgrund der Universalität des *PMBOK*® Guide fehlen ihm einsatzbereite Vorlagen. Das vorab erwähnte Book of Forms ist in der Praxis für Softwareprojekte nicht nutzbar. Verständliche, aber umfassende MS Word, Excel- und Project-Vorlagen sind erforderlich, um möglichst vielen Anwendern den Umgang mit der Vorgehensweise zu ermöglichen.

Die Anpassung der Inhalte des *PMBOK*® Guide an die projektspezifischen Bedürfnisse ist sogar eine Anforderung, die dessen Autoren an die Projektmanager stellen (vgl. [PMI08]). Die aufgelisteten Kriterien sind also streng genommen nichts anderes als die von den Autoren geforderte Anpassung des *PMBOK*® Guide. Ziel ist nicht der Ersatz des *PMBOK*® Guide, sondern eine Anpassung der Inhalte. Der Guide selbst wird damit nicht überflüssig – er bildet die methodische Basis.

Pragmatisches IT-Projektmanagement

Das Ergebnis ist eine sehr pragmatische Vorgehensweise, die auf Schnörkel verzich-

tet und dabei einfache, aber klare Vorgaben macht. Das so entstandene Vorgehen verbindet praxisrelevante Aspekte und Elemente der Softwareentwicklung mit den bewährten Inhalten des Guide und gliedert die Prozesse in ein Phasenmodell, das auch iterativ eingesetzt werden kann. Zudem verfeinert es die teilweise groben Prozessbeschreibungen in Bezug auf die konkrete Softwareentwicklung und macht sie so im betrieblichen Umfeld unmittelbar und praktisch anwendbar. Die hier vorgestellte Version wird im Besonderen dadurch pragmatisch, dass ein Prozess durch die unter <http://www.pitpm.net> zur Verfügung gestellten Vorlagen unmittelbar eingesetzt werden kann. Analog zum Titel „Pragmatisches IT-Projektmanagement“ heißt das bei der adesso AG entwickelte Vorgehen einfach nur PITPM. Mittlerweile wird es in zahlreichen Unternehmen und Projekten eingesetzt.

Wissensgebiete und Projektphasen

PITPM teilt ein Projekt zeitlich in fünf Phasen und organisatorisch in neun „Project Management Knowledge Areas“ bzw. Wissensgebiete auf. Sie entstammen weitgehend dem *PMBOK*® Guide, sind aber nicht vollständig kongruent: „Project Time Management“ und „Project Human Re-

source Management“ wurden in PITPM aus Gründen der Praktikabilität zusammengelegt; das Wissensgebiet „Softwareentwicklungsmanagement“ wurde ergänzt. Sämtliche Gebiete wurden den Erfordernissen von Softwareprojekten angepasst.

Die einzelnen Phasen geben den typischen Ablauf eines Projekts über Vorbereitung, Planung, Durchführung, Einführung und Abschluss wieder (siehe *Abbildung 3*). Der *PMBOK*® Guide kennt diese Phasen nicht, sondern unterteilt die Prozesse lediglich in Gruppen (siehe *Abbildung 2 unten*). Dadurch leidet der Praxiswert erheblich, da die lose Zuordnung die Anwendung deutlich komplizierter gestaltet und die Umsetzung vor allem für angehende Projektmanager oftmals eine erhebliche Herausforderung darstellt. In PITPM wird daher jeder Prozess einer Phase zugeordnet.

Berücksichtigung iterativer Prozesse

Wie bereits erläutert, gibt der *PMBOK*® Guide keine klaren Handlungsanweisungen für agile Softwareprojekte vor. Alles andere wäre systembedingt auch gar nicht sinnvoll: Schließlich ist der Guide nicht auf Softwareprojekte ausgerichtet.

Um jedoch dem grundsätzlichen Gedanken der agilen Entwicklung zu folgen,

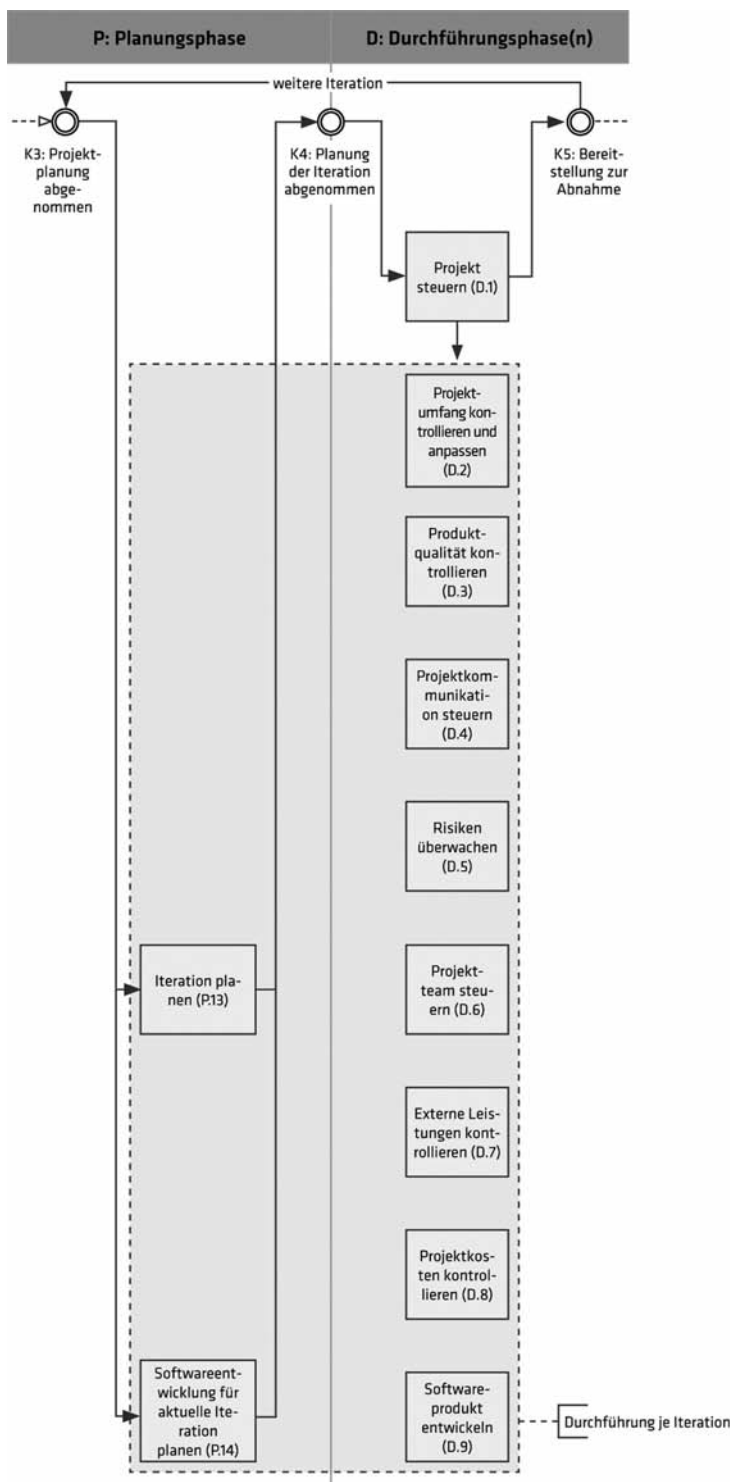


Abb. 4: Ablauf einer Iteration in PITPM.

also zum Beispiel Änderungen in den Anforderungen während der Projektlaufzeit berücksichtigen zu können, sollte jedes Projekt in Iterationen durchgeführt werden. Ein weiterer Vorteil der iterationsbasierten Vorgehensweise ist der häufige und regelmäßige Kundenkontakt: Je häufiger der Kunde – gleich ob intern oder extern – sein künftiges Produkt anfassen und

dadurch beurteilen kann, desto zufriedener werden letztlich alle Beteiligten sein, auch wenn ein iteratives Vorgehen völlig andere Anforderungen an die Kommunikation zwischen Auftraggeber und -nehmer stellt. Jede Iteration durchläuft eine Planungs- und eine Durchführungsphase. In der Planungsphase werden die Inhalte der folgenden Iteration aufgestellt. Auf Basis des

Product Backlog werden die Inhalte für die kommende Iteration gemäß der Anforderungsspezifikation definiert. Das Ergebnis – vergleichbar mit dem *Sprint Backlog* aus Scrum – umfasst nur die Anforderungen für eine Iteration.

Die geplanten Inhalte einer Iteration werden anschließend in der Durchführungsphase implementiert. In jedem Fall erfolgt zum Abschluss der Entwicklung einer Iteration die „Bereitstellung zur Abnahme“, die dem Kunden das Ergebnis zur Verfügung stellt. **Abbildung 4** verdeutlicht die Prozessabhängigkeiten in einer Iteration.

Kontrollpunkte sichern die Qualität

Der *PMBOK® Guide* sieht einen Prozess zur Kontrolle der Projektqualität vor („8.2 Perform Quality Assurance“ in [PMIO8]). In diesem Rahmen wird für gewöhnlich die Einhaltung der Projektprozesse ebenso überprüft wie die Übereinstimmung mit den Vorgaben aus dem Projektmanagement-Plan.

Im Rahmen von PITPM werden Kontrollpunkte eingesetzt. Eine Weiterführung des Projekts ist nur möglich, wenn alle Kriterien des Kontrollpunkts eingehalten werden. Setzt sich ein Projektmanager über die Kontrollpunkte hinweg, bringt er möglicherweise den Projekterfolg in Gefahr. Abweichungen sollten vom Projektsteuerungsgremium genehmigt werden. Durch acht Kontrollpunkte soll sichergestellt werden, dass das Projekt zielgerichtet und sicher durchgeführt wird. **Tabelle 1** gibt einen Überblick über die PITPM-Kontrollpunkte.

Einfache Prozesse

Ein *Prozess* ist eine meist sequenzielle oder nebenläufige Abfolge von Aktivitäten. Jeder Prozess besteht aus einer Anzahl von *Aktivitäten*, die in der jeweiligen Beschreibung des Prozesses abgebildet und beschrieben sind. Zu jeder Aktivität gehören Eingangsartefakte, Methoden, Techniken und Werkzeuge und Ergebnisartefakte:

- *Eingangsartefakte* sind die Dokumente, die für den Start einer Aktivität vorliegen sollten. Mithilfe der darin beschriebenen Informationen kann der Prozess durchlaufen werden.
- Im Rahmen der Aktivität werden die beschriebenen *Methoden*, *Techniken* und *Werkzeuge* eingesetzt, um das Produkt zu erstellen. Dabei kann es sich um Methoden, Softwareplattformen,

Kontrollpunkt	Phase	Bedingung
K1: Projektauftrag erteilt	Übergang Vorbereitung zu Planung	Die Planungsphase kann erst begonnen werden, wenn die Projekthinhalte geklärt und festgelegt wurden. Nur dann kann der Projektauftrag explizit erteilt werden. Wird der Projektauftrag nicht erteilt, wird das Projekt beendet. Dazu sollte die Abschlussphase zumindest rudimentär durchlaufen werden.
K2: Projekt konfiguriert	Planungsphase	Der Projektmanagementplan mit seinen Unterplänen wurde erstellt.
K3: Projektplanung abgenommen	Erstellung der Gesamtplanung vor dem Start der ersten Iteration	Die Gesamtprojektplanung mit Berücksichtigung der Release-Termine wurde erstellt und vom Auftraggeber abgenommen.
K4: Planung der Folge-Iteration abgenommen	Nach Abschluss der Planung der Iteration	Die Planung für die Durchführungsphase der Iteration wurde verabschiedet. Dieser Kontrollpunkt wird in jeder Iteration wiederholt.
K5: Bereitstellung zur Abnahme (BzA)	Durchführung	Das Release oder das Endprodukt ist aus Sicht des Auftragnehmers bereit zur Abnahme mit allen vertraglich vereinbarten Nebenprodukten (etwa Dokumentation, Schulungsunterlagen usw.).
K6: Abnahme erteilt	Übergang Durchführung zu Einführung	Der Liefergegenstand (Release oder Endprodukt) ist frei von abnahmeverhindernden Fehlern und wird vom Auftraggeber abgenommen.
K7: Produktionsstart	Übergang Einführung zu Abschluss	Der Liefergegenstand (Release oder Endprodukt) kann in den Produktivbetrieb übernommen werden.
K8: Projektende	Ende Abschlussphase	Das Projekt wurde gemäß PITPM dokumentiert und vollständig abgeschlossen.

Tabelle 1: Übersicht über die PITPM-Kontrollpunkte.

Verfahrensweisen usw. handeln. Nicht jede Aktivität benötigt spezifische Werkzeuge.

- Die **Ergebnisartefakte** dokumentieren schließlich das Ergebnis einer Aktivität. Sie sind entweder Projektergebnis oder werden in einem nachgelagerten Prozess verwendet oder verfeinert.

Die Darstellung der Prozesse ist an die BPMN angelehnt, die jedoch in einigen

Punkten vereinfacht wurde. **Abbildung 5** zeigt beispielhaft eine Prozessgrafik aus der Vorbereitungsphase, in der der Projektumfang bestimmt wird. Dabei geht es noch nicht um die detaillierte Festlegung des Inhalts und Umfangs in allen Feinheiten, sondern um eine grobe Zusammenstellung und Abgrenzung der Projekthinhalte, auf deren Basis anschließend eine Schätzung möglich sein soll. Dargestellt wird der gesamte Prozess mit seinen drei Aktivitäts-

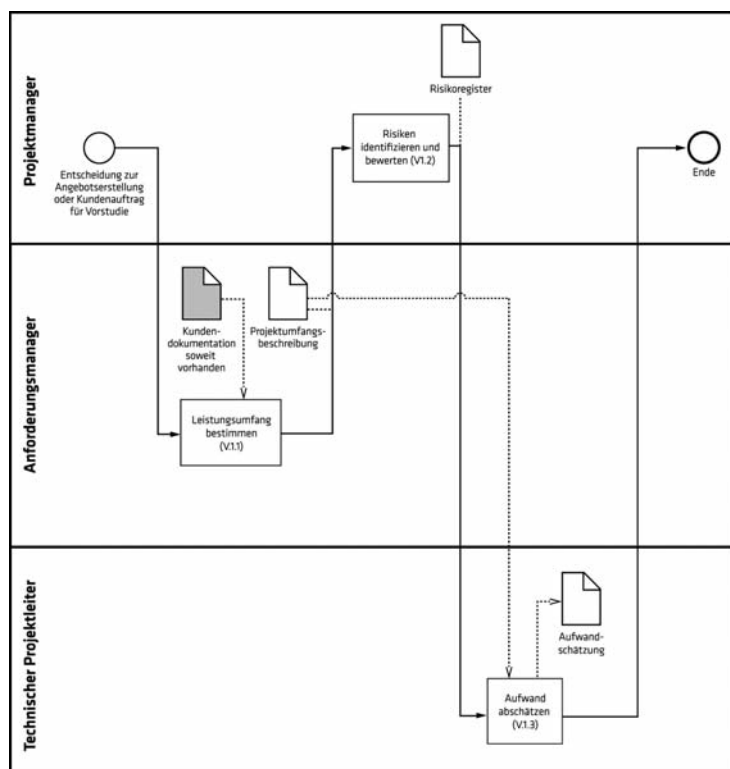


Abb. 5: Beispielprozess „Projektumfang bestimmen“.

ten und den Eingangs- und Ergebnisartefakten.

Fazit

Das auf dem *PMBOK*® Guide basierende Vorgehen PITPM lässt sich direkt im eigenen Projekt einsetzen, ohne dass dafür erst eine umfangreiche Methodik im Unternehmen ausgerollt werden muss. Allein durch die Verfügbarkeit der Vorlagen auf Basis der Prozesse kann ein Projektmanager sein Projekt beschleunigen und es qualitativ verbessern. Die Anwendung von PITPM ist besonders in kleineren bis mittleren Projekten sinnvoll, da die erforderlichen Anpassungen an die Prozesse meist gering sind und der Nutzen sofort sichtbar wird.

PITPM erschien im Juli 2010 als Buch unter dem Titel „Pragmatisches IT-Projektmanagement Softwareentwicklungsprojekte auf Basis des *PMBOK*® Guide leiten“ im dpunkt.verlag (vgl. [Spi10]). Sämtliche Vorlagen stehen im Internet unter <http://www.pitpm.net> kostenlos zum Download bereit. Auf dem Projektmanagement Summit des PMI im November 2010 in München wurde PITPM mit dem „Best Paper Award“ ausgezeichnet.

Literatur & Links

[Ble08] W.-G. Bleek, H. Wolf, Agile Softwareentwicklung: Werte, Konzepte und Methoden, dpunkt.verlag 2008

[OMG-a] Object Management Group (OMG), BPMN Homepage, siehe: <http://www.bpmn.org>

[OMG-b] Object Management Group (OMG), UML Homepage, siehe: www.uml.org

[PMIO4] Project Management Institute (Hrsg.), A Guide to the Project Management Body of Knowledge: *PMBOK*® Guide, 3rd. Ed., Project Management Institute 2004

[PMIO6] Project Management Institute (Hrsg.), Government Extension to the *PMBOK*® Guide, 3rd Ed., Project Management Institute 2006

[PMIO7] Project Management Institute (Hrsg.), Construction Extension to the *PMBOK* Guide, 3rd Ed., PMI 2007

[PMIO8] Project Management Institute (Hrsg.), A Guide to the Project Management Body of Knowledge, 4th Ed., PMI 2008

[Spi10] N. Spitzcok von Brisinski, G. Vollmer, Pragmatisches IT-Projektmanagement: Softwareentwicklungsprojekte auf Basis des *PMBOK*® Guide führen, dpunkt.verlag 2010

[Sta09] C.S. Stackpole, A Project Manager’s Book of Forms, PMI 2009