

# Prozessverbesserung durch fragmentierte Anwendung von Scrum & Co.

Philipp Diebold<sup>1</sup>, Thomas Zehler<sup>1</sup>, Anna Schmitt<sup>1</sup>, Frank Simon<sup>2</sup> und Birger Kruse<sup>2</sup>

**Abstract:** Agile Entwicklung ist in der Softwareindustrie vorherrschend. Meist wird auf Scrum als agile Vorgehensweise gesetzt. Die Erfahrungen haben jedoch gezeigt, dass die in Scrum enthaltenen Einzelbausteine als auch deren Zusammenspiel meist auf den Kontext angepasst werden müssen, insbesondere vor dem Hintergrund einer möglichst smarten Transition. Um die individuelle Anpassung einer „fertigen“ Methode zu vermeiden und eine schrittweise Einführung von einzelnen Bausteine und Bausteingruppen zur sukzessiven Prozessverbesserung herzuleiten, wird in diesem Beitrag eine neue Idee vorgestellt. Sie beschreibt, wie ein Bausteinkatalog, bestehend aus (vorrangig) agilen Bausteinen, zielorientiert aufgebaut und modular genutzt werden kann. Zur exemplarischen Darstellung der Grundidee werden Beispiele aus dem Gesamtkatalog verwendet. Die Anwendungsszenarien adressieren hierbei einerseits das Management und andererseits die Mitglieder der Entwicklungsteams.

**Keywords:** Scrum, Scrum-But, Agile Methoden, Agile Praktiken, Prozessbausteine, Prozessverbesserung, Ziel-orientiert, Kontext-basiert, Bausteinorchester.

## 1 Einleitung & Motivation

Die Mehrzahl an Unternehmen (insbesondere KMUs) sind in der heutigen Zeit in einem globalen, hochdynamischen Umfeld tätig. Sie müssen sehr flexibel auf neue Marktchancen, veränderte Bedingungen [Re07] und eine starke, globale Mitbewerbersituation, gerade im immateriell geprägten Geschäft der IT, reagieren. Immer kürzer werdende Release-Zyklen zwingen Unternehmen, ihre Softwareentwicklung noch iterativer und kundenorientierter zu gestalten. Hierin kann insbesondere für Mittelständler eine große Chance liegen, da sie Änderungen aufgrund ihrer überschaubareren Größe naturgemäß leichter und zügiger umsetzen können. Doch welche Einzelmethoden existieren für ein Umfeld effektiv, um die Time-to-Market der Softwareentwicklung zu verbessern? Welche Verfahren sind möglich, um mittelstandstypische Projekte demokratischer, das heißt unter Berücksichtigung der Meinung aller Mitarbeiter, durchführen zu können und für diese das, heute gerade für die Generation Y [Pa13] relevante, Anforderungsprofil eines guten Arbeitgebers zu erfüllen? Wie kann ein IT-Projekt insgesamt zielorientierter und damit auch änderungsfreundlicher durchgeführt werden, um dem Team und dem Kunden mehr direkte Kommunikation zu ermöglichen und jeweils unmittelbares Mitspracherecht zu geben? Wie kann Nutzerfeedback früher eingeholt werden, das direkt in der nächsten Entwicklungsiteration mit einfließt?

---

<sup>1</sup> Fraunhofer IESE, Process Engineering, Fraunhofer-Platz 1, 67663 Kaiserslautern, {vorname.nachname}@iese.fhg.de

<sup>2</sup> BLUECARAT AG, Albin-Köbis-Str. 4, 51147 Köln, {vorname.nachname}@bluecarat.de

Die häufig von agilen „Evangelisten“ in diesem Spannungsfeld propagierte agile Revolution, entlang derer ein Unternehmen in Form eines Big-Bang-Wechsels des Entwicklungsvorgehens nur noch agil zu arbeiten hat, z.B. [KH14], ist für den Mittelstand häufig nicht praktikabel. Während einer solchen umfassenden Veränderung entstehen oftmals Produktivitätseinbrüche, die vom Mittelstand nicht aufgefangen werden können. Zusätzlich existieren keine Handreichungen in Form feingranularer Baukästen, die angeben, wie diese Revolution Schritt für Schritt angegangen und umgesetzt werden soll. Daher offenbaren sich in der Praxis Nachteile, wie zum Beispiel:

- (1) ein fehlender transparenter Überblick über den Projektfortschritt (inkl. Kostenkontrolle und Ressourcenverwaltung) [VV13],
- (2) Unzulänglichkeiten in der Kommunikation sowohl in den Projektteams als auch mit dem Kunden [KM14][VV13][Di14],
- (3) Missverständnisse im Umgang mit Änderungswünschen des Kunden vor dem Hintergrund eines Zeit- und Budgetplans [Fa14] sowie
- (4) Unklarheiten über die Art und den Umfang der im Projekt anzufertigenden Dokumentation [KM14][Di14].

Viele Unternehmen sehen daher eine unüberwindbare Lücke zwischen gestiegenen Anforderungen und den meist nur disruptiv umzusetzenden, neuen agilen Entwicklungsmethoden. Ein schrittweises, gezieltes und „standardisiertes“ Adressieren spezifischer Probleme, Verbesserungen und konkreter Herausforderungen durch geeignete Elemente und Methoden, z.B. aus der agilen Softwareentwicklung, obliegt mehr einer vorsichtig gestarteten Eigeninitiative der betroffenen Unternehmen. Ansätze wie halb-agil, fragmentiertes agil oder Scrum-But sind in der Literatur negativ besprochen [We15]. Was fehlt ist zum einen eine Übersicht darüber, welche Elemente grundsätzlich für eine solche Verbesserung geeignet sind. Zum anderen stellen sich die Fragen, welche Vorbedingungen für solche anwendbaren Bausteine<sup>3</sup> erfüllt sein müssen, welchen Nutzen und Kosten sie für das Unternehmen haben und wie sie geeignet verbunden und in den bestehenden Entwicklungsprozess integriert werden können.

Um diese Aspekte zu beantworten, wurde die Methodik ProKoB (ProjektKontext spezifische ProzessBaustein-Orchestrierung zur Prozessverbesserung) mit der folgenden Vision entwickelt: Es gibt eine smarte Transition im Entwicklungsvorgehen, die eine fortwährende Produktivität bei nachhaltiger Adressierung von Verbesserungszielen garantiert. Die damit verbundenen und zu adressierenden Forschungsfragen sind: (1) Welche Prozessbausteine gibt es und entlang welcher Attribute lassen sie sich charakterisieren? (2) Was ist der jeweilige konkrete unternehmerische Nutzen? (3) Welche Vorbedingungen sollten jeweils erfüllt sein, um die Kontinuität nicht zu gefährden? (4) Was sind gute Kombinationen solcher Bausteine?

---

<sup>3</sup> Ein Prozessbaustein ist ein Teilprozess, welcher eine Aktivität beschreibt, die bestimmte Eingaben in bestimmte Ausgaben überführt. Der Prozessbaustein selbst ist auf dieser Ebene tool-, prozess- und vorhabens-unabhängig definiert und beschreibt einen konkreten Handlungsleitfaden sowie notwendige Parameter für eine erfolgreiche Anwendung. (angepasst aus [Ha14])

## 2 Stand der Wissenschaft und Technik

Durch die zunehmende Verbreitung agiler Vorgehensweisen (Scrum, XP) in den vergangenen 15 Jahren hat sich der Schwerpunkt der Forschung auf den Bereich agiler Entwicklung fokussiert [Ve15][Ca15]. Die entwickelten Ansätze beschreiben „fest definierte“ Methoden, die oftmals vor einem Einsatz zunächst kontextspezifisch angepasst werden [Di15]. Der Nutzen vollständiger agiler Vorgehen wurde in der jüngeren Vergangenheit häufiger betrachtet. Es fehlte jedoch die Detaillierung darüber, welche der enthaltenen Methoden und Praktiken welchen konkreten Nutzen nach sich ziehen [SY07][HA05] und welche Kombinationen während einer Transition ggfs. hilfreicher sind als andere. Nur zu wenigen agilen Elementen finden sich weitergehende Informationen über deren Zweck, etwaige Rahmenbedingungen (z.B. erforderliche Vorbedingungen für den Einsatz des Elements, z.B. Pair Programming: >1 Person oder Daily StandUp: <10 Personen) und den Nutzen durch deren Einführung (z.B. Pair Programming erhöht die Codequalität, verbessert die Einarbeitung neuer Mitarbeiter) [HA05].

Durch die Fokussierung auf „fest definierte“, atomar aufgefasste, agile Vorgehensweisen wie Scrum wird die schrittweise Einführung einzelner darin enthaltener Fragmente, inklusive der erforderlichen spezifischen Anpassung an den konkreten Unternehmenskontext, vernachlässigt. Das Ziel einer derartigen Transition, die nur vollständig agile Vorgehensweisen kennt, ist gerade für klein- und mittelständische Unternehmen nicht praktikabel. Sie macht – wie im Fall der Einführung von Scrum – eine grundlegende, eher revolutionäre Veränderung des Entwicklungsvorgehens erforderlich [DZ16]. Diese kann ein KMU im Tagesgeschäft nicht leisten. Die Annahme, dass ein mittelständisches Unternehmen auch „halb-agil“ erfolgreich sein kann und die fehlende Agilität durch andere Attribute kompensieren kann, ist bisher nur wenig erforscht [DZ16] oder wird in der Literatur zum Teil abgelehnt [We15].

## 3 Schrittweise Prozessverbesserung durch den Bausteinkatalog

Die ProKoB-Methodik (Abb. 1) schlägt eine systematische Katalogisierung leichtgewichtiger Einzelbausteine zur Verbesserung des gegenwärtigen Entwicklungsvorgehens in Verbindung mit einem geeigneten Einführungsweg (Transition) über wohldefinierte Bausteinkombinationen vor. Hierzu werden evidenzbasierte Elemente zur Adressierung bestimmter Problemfelder (z.B. Kundeneinbindung, Demokratisierung) oder Themenfelder (z.B. Planung, Umsetzung, Projektabschluss) in einem Katalog von Bausteinen strukturiert und verknüpft. Dieser bietet dem Nutzer eine transparente, leichtgewichtige Darstellung von und einen schnellen Zugriff auf existierende Bausteine. Auf Basis des angestrebten Verbesserungsziels sowie des Projekt- und Unternehmenskontextes werden aus dem Bausteinkatalog geeignete Elemente ausgewählt und orchestriert. Zudem wird ein spezifischer Weg zur Umsetzung der adäquaten Verbesserungsmaßnahmen aus den Bausteinen abgeleitet und den Unternehmen als Vorschlag unterbreitet.

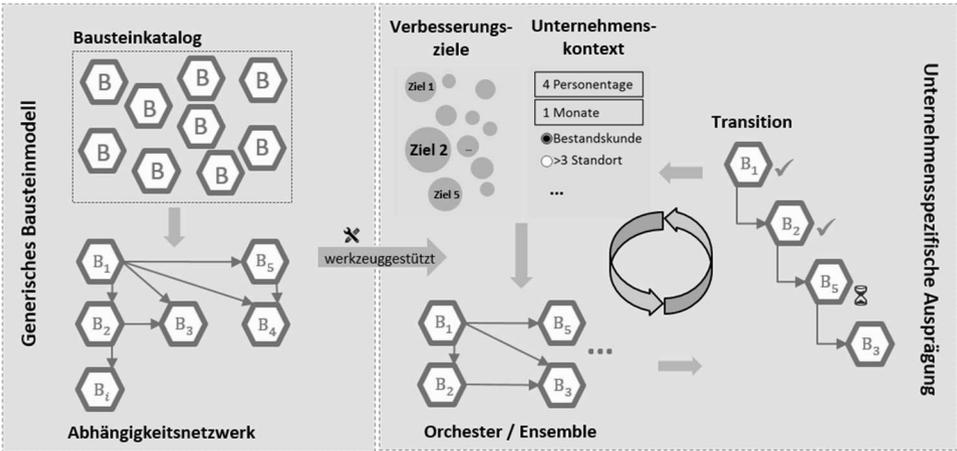


Abb. 1: Methode zur schrittweisen Prozessverbesserung

### 3.1 Der Bausteinkatalog: Wie sieht ein Baustein aus?

Zur einheitlichen Beschreibung und Weiterverarbeitung (z.B. Verknüpfung) der Bausteine wurde eine spezifische Ontologie entworfen. Auf Basis dieser Ontologie wird jeder Baustein mit Hilfe der folgenden Attribute beschrieben: (1) Name, (2) Synonyme, (3) Beschreibung/Funktionsweise, (4) Referenz, (5) Vorbedingungen, (6) Nachbedingungen, (7) Vorteile, (8) Nachteile, (9) Fallstricke, (10) Variationsparameter, (11) „Undo“-Schritte, (12) Risikofaktoren, (13) verschiedene Aufwände, z.B. Einführungsaufwand, (14) Fallbeispiele / Evaluation. Neben der Bausteinbeschreibung wurde zwischen dem einfachen „Durchführen“ / Anwenden von Bausteinen und dem komplexeren / langfristigeren „Etablieren“ unterschieden. Diese Differenzierung ist notwendig, da z.B. ein Jour-Fix direkt durchgeführt werden kann, eine Testgetriebene-Entwicklung (TDD) jedoch erst etabliert werden muss.

Zusätzlich beinhaltet die Ontologie eine Matrix (Tab. 1), bestehend aus den fünf PMI-Prozessgruppen [PMI13] und fünf abgeleiteten „Artefaktgruppen“ aus den verschiedenen Phasen der ISO12207 [ISO08]. Der Nutzen eines Bausteins ist keinesfalls auf eine bestimmte Zelle der Matrix beschränkt, sondern kann sich über mehrere Zellen erstrecken. Beispiel hierfür ist das „Daily StandUp“: Es ist in der Prozessgruppe „Planung“ zu verorten, deckt dort aber alle Artefaktgruppen ab.

Die Ontologie bildet über einen Link eine Verknüpfung des jeweiligen Bausteins auf ein oder mehrere Verbesserungsziele ab (analog zu [DZ15]). Diese wurden in vier Oberkategorien gruppiert: Produktqualität, Time-to-Market, Kundeneinbeziehung und Organisationsdemokratisierung. Neben dem Input der Vorbedingung besteht die Möglichkeit, anzugeben, in welchem Kontext der Baustein nicht anwendbar ist. Hierfür wurden Kontextcharakteristika aus der Literatur [KK13] mit Industriepartnern diskutiert. Basierend auf dieser Diskussion und dem Einfluss der Industriepartner auf die Auswahl der Bausteine kristallisierten sich folgende fünf Charakteristika als wesentlich heraus: Projektdauer, Teamgröße im Projekt, Team-Verteilung, Endnutzerverfügbarkeit und Kundenverfügbarkeit.

### 3.2 Ensemble/Orchestrierung: Wie verbinden sich einzelne Prozessbausteine?

Wenn der Fokus des Katalogs auf individuellen Bausteinen und deren jeweiligen Zielen liegt, verstärkt die Kombination einzelner Bausteine diese und weitere Ziele und bringt einen signifikanten Mehrwert. Dies deckt sich von der Stoßrichtung her zwar mit dem häufig als atomar angesehenen Scrum, wonach im erstellten Framework nichts weggelassen werden kann, ist in der Praxis aber deutlich modularer.

Sinnvoll zusammengesetzte Mengen an Prozessbausteinen werden in unserer Ontologie und damit in der ProKoB-Methodik als **Ensembles** bezeichnet. Sie werden hauptsächlich über ein Matching von Nach- zu Vorbedingungen (bzw. Aus- zu Eingaben), über die oben beschriebene Matrix und die intendierten Gesamtverbesserungsziele gebildet. Die Matrix sollte im Unternehmen möglichst vollständig und effektiv besetzt sein. Aus diesen Verbindungen entsteht ein Abhängigkeitsnetzwerk.

Mit diesem Netzwerk lassen sich auch gängige Methoden, wie zum Beispiel Scrum und XP aus der agilen Welt, als Ensembles modellieren. Scrum als Ensemble (Tab. 1) basiert auf [ScS13] und lässt sich in die sechs Bausteine<sup>4</sup> und deren Durchführung/Etablierung „Backlog“, „Definition of Done“ (DoD), „Planning Meeting“, „Review“, „Retrospective“ und „Daily StandUp“ dekomponieren. Die vorgeschriebenen Rollen werden in diesem Zusammenhang nicht genannt, da sie aufgrund ihrer fehlenden Aktivitätsbeschreibung keine Bausteine darstellen.

	Initiierung	Planung	Umsetzung	Controlling	Abschluss
Anforderung	Product Backlog		Sprint Backlog		
Entwurf		Daily StandUp		Daily StandUp	Retrospective
Implementierung	Planning Meeting			Review	Review
Test				DoD	
Deployment					

Tab. 1: Matrix mit beispielhafter Füllung durch Scrum

Neben den bisher allgemein definierten Ensembles verwenden wir in unserer Ontologie den der Musik entlehnten Begriff eines **Orchesters**. Das Orchester ist ein vollständiges Ensemble, bei dem alle 25 Zellen der oben beschriebenen Matrix mindestens von einem Baustein belegt werden. Vor dem Hintergrund dieser eingeführten Begriffswelt stellt Scrum offenkundig kein Orchester dar, da es keinerlei Bausteine für die Prozessgruppe „Umsetzung“ liefert (Tab. 1). Zudem ist aus unserer Sicht nicht jedes Orchester auch gleich ein vollständiger Entwicklungsprozess, da eine Zelle mit lediglich einem Baustein nicht vollumfänglich adressiert sein muss.

<sup>4</sup> Der Ursprung der Bausteine liegt nicht immer in der agilen Entwicklung, sondern entstand teilweise deutlich früher. Zum Beispiel wurden Reviews schon in den 70er Jahren etabliert [Fr06], auch wenn sie im agilen Umfeld ein wenig anders verwendet werden. Planning-Meetings sind ebenfalls bereits bekannt, z.B. aus der Delphi-Methode.

### 3.3 Transition: Welcher Baustein ist der „Erste“ bei einer Evolution?

Eine Transition gibt an, in welcher schrittweisen Reihenfolge die einzelnen Bausteine in das Unternehmen bzw. den aktuellen Prozess eingeführt werden sollen und wie die Schnittstellen zu bereits existierende Methoden und Prozessen gelebt werden können. Durch diesen evolutionären Ansatz wird eine kontinuierliche Weiterentwicklung des Prozesses unterstützt.

Ähnlich wie die Ensembles basiert die Transition auf den Vor- und Nachbedingungen und der Matrix aus PMI-Prozessgruppen und „Artefaktgruppen“ der ISO12207. Speziell bei der Matrix ist zu beachten, dass der gängige Weg von links oben nach rechts unten (Zeile für Zeile) verläuft. Dies ist jedoch nicht mit dem „Wasserfall“-Modell gleichzusetzen. Grund ist, dass einzelne Zeilen mehrmals iterativ durchlaufen oder einzelne Bausteine über mehrere Zellen vertikal (oder sogar eine ganze Spalte) eingesetzt werden und somit das „Durchlaufen“ der Matrix verändern können.

Die oben beschriebene Klassifizierung in „durchzuführende“ und „zu etablierende“ Bausteine kommt unter anderem nun zum Tragen, da Bausteine, die sich schnell durchführen oder erstellen lassen, keine großen Aufwände zum Heben der damit verbundenen Vorteile benötigen. Sie sind merklich leichter in einem Prozess zu testen, einzubinden oder langfristig umzusetzen. Aus diesem Grund ist eine der Strategien zum Erstellen der passenden Transition, dass die „durchführenden“ Bausteine vor den „etablierenden“ Bausteinen eingeführt werden, um schnelle Ergebnisse abzuleiten. Die „etablierenden“ Bausteine sind eher längerfristig relevant. Neben dieser gibt es weitere Strategien, wie z.B. zielgerichtet zu erkennen, welche der Bausteine am meisten relevante Verbesserungsziele adressieren. Zusätzlich könnte auch das „Ausbalancieren“ und schnelle Füllen der Matrix sowie die Anzahl der existierenden Evidenzen der einzelnen einzuführenden Bausteine eine Rolle spielen.

## 4 Anwendungsszenarien

Diese Idee der Prozessverbesserung adressiert zwei unterschiedliche Nutzergruppen, die sich in verschiedenen Anwendungsszenarien widerspiegeln: Zum einen wird das **Management** (z.B. Inhaber von KMUs oder Entscheider) durch die Möglichkeit des Einstiegs über die ziel- und problem-orientierte Fragestellung adressiert. Zum anderen werden die **operativen Mitarbeiter** (z.B. Entwickler oder Tester) durch die Möglichkeit des explorativen „Durchstöberns“ der verschiedenen Inhalte, z.B. Bausteine oder Ensembles, angesprochen.

### 4.1 Management: Zielorientierte Fragestellung

Die zielorientierte Anwendung empfiehlt auf Basis intendierter Verbesserungsziele und des gegebenen spezifischen Projektkontextes wirksame Prozessbausteine und Prozessbausteinensembles. Zudem bietet sie Möglichkeiten des Benchmarkings gegenüber anderen, ähnlichen Verbesserungsprojekten. Sie unterstützt eine feingranulare Begleitung bei der Umsetzung und Bewertung der selektierten Prozessbausteine und -ensembles.

Dieses Szenario lässt sich mit Hilfe folgender (teilweise optionaler) Schritte beschreiben:

- (1) Auswahl von relevanten Verbesserungszielen inklusive einer Priorisierung auf Basis der selbst erkannten Probleme / Potentiale des Unternehmens.
- (2) Eingabe des jeweiligen Unternehmens- bzw. Projektkontextes mit Hilfe der in Kapitel 3.1 vorgestellten Kontextcharakteristika.
- (3) Auswahl eines Prozessbausteins oder mehrerer Prozessbausteine, welche durch die Selektierung in Schritt 1 und / oder Schritt 2 vorgeschlagen und ggf. nach verschiedenen Attributen, z.B. Einführungsaufwand oder Risiko, sortiert wurden.
- (4) Auswahl eines Ensembles basierend auf der Selektion der Prozessbausteine (Schritt 3) und verschiedener Ensemble-Vorschläge, die (mindestens) alle selektierten Bausteine beinhalten.
- (5) Betrachtung der Detailinformationen des ausgewählten Ensembles, z.B. über die Abdeckung und Lage in der in Kapitel 3 beschriebenen Matrix sowie über die adressierten Verbesserungsziele.
- (6) Bereitstellung einer Transition des ausgewählten Ensembles.

#### **4.2 Mitarbeiter: Exploratives Durchstöbern**

Beim explorativen „Durchstöbern“ kann der Nutzer sich effektiv durch das Abhängigkeitsnetzwerk der Prozessbausteine, Ensembles, Verbesserungsziele, Prozessgruppen etc. navigieren. Dadurch können interaktiv interessante und ggf. zueinander passende Prozessbausteine identifiziert werden. Des Weiteren sind das Benchmarking und die Unterstützung bei der Umsetzung von selektierten Bausteinen oder Ensembles möglich.

Im Vergleich zum in Kapitel 4.1 beschriebenen Szenario unterscheidet sich dieses hauptsächlich durch den Einstieg:

- (1) Eingabe des jeweiligen Unternehmens- bzw. Projektkontextes mit Hilfe der in Kapitel 3.1 vorgestellten Kontextcharakteristika.
- (2) „Durchstöbern“ der einzelnen Elemente (Prozessbausteine, Ensembles, Verbesserungsziele, Prozessgruppen oder Prozessartefakte).
- (3) Einstieg in oberes Szenario bei Schritt 3 oder 4, abhängig vom ausgewählten Element in Schritt 2.

## **5 Zusammenfassung und Future Work**

In diesem Papier wurde eine schrittweise Prozessverbesserung durch einen Bausteinkatalog mit Elementen aus der agilen Softwareentwicklung vorgestellt. Es wurde eine Übersicht über die zugrundeliegende Ontologie inklusive Orchestrierung und Transition sowie mögliche Anwendungsszenarien gegeben. Das Ergebnis, dass Scrum kein vollständiges Entwicklungsframework ist, ist ein spannender Nebeneffekt. Die Methodik der Prozessverbesserung wird aktuell im Forschungsprojekt ProKoB im Rahmen der KMU innovativ-

Initiative des Bundesministeriums für Bildung und Forschung (BMBF) verfeinert und weiterentwickelt. Gegenwärtig werden möglichst viele Bausteine gesammelt und mit Hilfe der Ontologie vollständig systematisch beschrieben, um eine ausreichende Grundlage für vorzuschlagende Ensembles und Transitionen aufzubauen. In der Anwendung ist besonders die Verknüpfung der Bausteine (und Ensembles) mit den Verbesserungszielen relevant. Die Evidenz der Verknüpfung wird in ProKoB durch explizite Evaluationsphasen gezeigt werden, um den Einfluss der Prozessbausteine auf die Verbesserungsziele mit empirisch validierten Daten zu unterfüttern. Zusätzlich ist eine web-basierte Toolunterstützung für die Methodik vorgesehen. Des Weiteren beschäftigt sich das Konsortium mit dem Aufbau einer ProKoB-Community, mit welcher der Bausteinatalog fortgeschrieben und die Empirie ausgeweitet wird. Unter anderem für diesen Zweck wurde eine Webseite, [www.prokob.info](http://www.prokob.info), erstellt. Auf dieser können Interessierte und Nutzer Feedback in Form zusätzlicher Bausteinen geben. Hierzu müssen sie lediglich das bereitgestellte Formular zu ihrem Wunschbaustein / ihren Wunschbausteinen ausfüllen und absenden.

**Danksagung:** Diese Arbeiten wurden im Rahmen des Forschungsprojektes „ProKoB“ durch das Bundesministerium für Bildung und Forschung (BMBF 01IS15038) gefördert.

## Literaturverzeichnis

- [Ca15] Capgemini, Hewlett-Packard, Sogeti: World Quality Report 2014-2015. Sixth Edition. 2015.
- [Di14] Diebold, P.; Lampasona, C.; Kemmann, O.; Kemmann, C.: Challenges in a Creative and Dynamic Environment: A Process Review at a Medium-sized Web Company. In Tagungsband des DASMA Software Metriken Kongresses (MetriKon2014). Stuttgart 2014. Shaker, S.159 – 178, 2014.
- [Di15] Diebold, P.; Ostberg, J.P.; Wagner, S.; Zendler, U.: What Do Practitioners Vary in Using Scrum?. In Proceedings of 16th International Conference on Agile Software Development (XP2015). Helsinki 2015. Springer, S. 40-51, 2015.
- [DZ15] Diebold, P.; Zehler, T.: Agile Practice Impact Model - idea, concept, and application scenario. In: Proceedings of the 2015 International Conference on Software and System Process (ICSSP2015). Tallinn 2015. ACM, S. 92-96, 2015.
- [DZ16] Diebold, P.; Zehler, T.: The Right Degree of Agility in Rich Processes. In (Kuhrmann, M. et al. Hrsg.): Managing Software Process Evolution. Springer, S. 15-38, 2016.
- [Fa14] Fagerholm, F.; Ikonen, M.; Kettunen, P.: How do software developers experience team performance in lean and agile environments? In Proceedings of 18<sup>th</sup> International Conference on Evaluation and Assessment in Software Engineering (EASE2014). ACM, Article 7, 2014.
- [HA05] Hulkko, H.; Abrahamsson, P.: A multiple case study on the impact of pair programming on product quality. In Proceedings of International Conference of Software Engineering (ICSE2005). ACM, New York, NY, USA, S. 495-504, 2005.
- [Ha14] Eberle, H.: „Prozessbausteine“. Dissertation. Universität Stuttgart. 2014. [http://elib.uni-stuttgart.de/bitstream/11682/3262/1/Prozessbausteine\\_druckversion.pdf](http://elib.uni-stuttgart.de/bitstream/11682/3262/1/Prozessbausteine_druckversion.pdf).
- [ISO08] International Organization for Standardization: ISO/IEC 12207:2008 Systems and software engineering - Software life cycle processes. 2008.

- [KH14] Kampfmann, R., Heymann, J.: SAP's Road to Agile Software Development, In Future Business Software, Current Trends in Business Software Development. Springer, 2014.
- [KM14] Korkala, M.; Maurer, F.: Waste identification as the means for improving communication in globally distributed agile software development. J. Syst. Softw. 95 (September 2014). Elsevier, S. 122-140, 2015.
- [KK13] Kalus, G.; Kuhrmann, M.: Criteria for Process Tailoring: A Systematic Review. In: Proceedings of the 2013 International Conference on Software and System Process (ICSSP2013). ACM, 2013.
- [Fr06] Frühauf, K.; Ludewig, J.; Sandmayr, H.: Software-Prüfung: Eine Anleitung zum Test und zur Inspektion. Vdf Hochschulverlag. 2006.
- [Pa13] Parment, A.: Die Generation Y - Mitarbeiter der Zukunft motivieren, integrieren, führen. Gabler, Wiesbaden, 2013.
- [PMI13] Project Management Institute: A Guide to the Project Management Body of Knowledge. 2013.
- [Re07] Rech, J.: 2007. Discovering trends in software engineering with google trend. SIGSOFT Softw. Eng. Notes 32, 2 (March 2007), S. 1-2, 2007.
- [ScS13] Schwaber, K.; Sutherland, J.: The Scrum Guide. 2013.
- [SY07] Sison, R.; Yang, T.: Use of Agile Methods and Practices in the Philippines. In Proceedings of Asia-Pacific Software Engineering Conference (APSEC2007). IEEE Computer Society, Washington, DC, USA, S. 462-469, 2007.
- [Ve15] VersionOne: 10th Annual State of Agile Development Survey. 2015.
- [VV13] Van Waardenburg, G.; Van Vliet, H.: When agile meets the enterprise. Information Software Technology 55, 12 (December 2013), 2154-2171, 2013.
- [We15] Weidner, I.: Klassische Projektleiter sind in agilen Teams überflüssig. Computerwoche 07-2015. IDG, 2015.