

Bierdeckelskizzen - Scrum ist leicht aber nicht einfach

Alexander Krieg¹

Abstract: Vor ungefähr zwanzig Jahren wurde Scrum entwickelt. Ein Grund dafür war das Fehlen eines Rahmenwerks für die Softwareentwicklung, das die nötige Flexibilität und Schnelligkeit besitzt. Scrum ist das bekannteste Framework aus der agilen Softwareentwicklung. Der Prozess kann auf einem Bierdeckel skizziert werden. Dennoch fällt es Unternehmen und Teams schwer Scrum einzuführen. Was ist das Leichte an Scrum und warum ist es dennoch nicht einfach? Diese Fragen werden im vorliegenden Aufsatz behandelt und mögliche Lösungsansätze dargestellt. Es wird aufgezeigt, dass die Schwierigkeit nicht am Scrum Prozess selbst liegt sondern vielmehr an der Veränderung der Arbeitskultur sowie der vorherrschenden Kommunikations-, Führungs- und Denkmuster, die mit der Einführung einhergehen.

Keywords: Scrum Team, Scrum Master, moderne Führung, agiles Coaching, hybride Vorgehen, Kulturwandel, agiles Projektmanagement, Führung als Dienstleistung, agiles Schätzen, agiles Risikomanagement, agile Transition, Continuous Integration, IT-Kanban.

1 Einleitung

Scrum basiert auf Ansätzen des 1986 veröffentlichten Artikels (vergl. [TN86]) von Hirotaka Takeuchi und Ikujiro Nonaka. Sie beschreiben dort, dass sich die Regeln der Produktentwicklung verändert haben. Es geht nicht mehr nur um hohe Qualität und geringe Kosten, es kommen noch Geschwindigkeit und Flexibilität hinzu.

Ungefähr zehn Jahren nach der Veröffentlichung des Artikels „The New New Product Development Game“ entwickelten Jeff Sutherland und Ken Schwaber „Scrum“. Es gibt in Scrum nur drei definierte Rollen (Product Owner, Scrum Master und Entwickler Team), vier unterschiedliche Meetings (Daily Standup, Sprintplanungsmeeting, Sprintreview und Retrospektive) und ein sich ständig wiederholender (iterativer) Prozess (vergl. [SS13]). Aus prozessualer Sicht klingt das sehr einfach. Dennoch haben Teams und Unternehmen häufig Schwierigkeiten bei der Einführung bzw. einer späteren Ausweitung von Agilität (vergl. [Le07]) in mehrere Teams bzw. in die gesamte Organisation. Jede Scrum Rolle ist mit einem entsprechenden Empowerment ausgestattet. Das bedeutet, dass die Rollen mit einer klar zugeordneten Verantwortung und Ermächtigung ausgestattet sind, um gewisse Themen durchzusetzen bzw. einzufordern, ohne dabei eine disziplinarische Rolle einzunehmen. Die Entwicklung neuer Vorgehensmodelle, Rahmenwerke und eine Flexibilisierung der vorhandenen Strategien werden immer notwendiger, um die zunehmenden Veränderungen von Projektzielen, Rahmenbedingungen und stetig wachsender Dynamik in den Griff zu bekommen. Die klassischen Projektma-

¹ Acando GmbH, Millerntorplatz 1, 20359 Hamburg, alexander.krieg@acando.de

nagementvorgehen sind nicht flexibel und leichtgewichtig genug, um adäquat auf die hohe Veränderungsfrequenz zur Projektlaufzeit zu reagieren. Viele Projekte in der Softwareentwicklung stellen ein Team immer wieder vor neue, bisher noch nicht da gewesene Herausforderungen. Diese erfordern eine große Erfahrung der Teams und ein hohes Maß an Flexibilität und Anpassungsvermögen an die Vorgehensweisen. Softwareentwicklung ist eine relativ junge Disziplin, grundlegende Standards müssen noch weiter zu einer größeren Reife entwickelt werden. Brücken, Häuser und Straßen werden seit mehreren Jahrhunderten gebaut, entsprechend groß sind die Erfahrung und die vorhandenen Standards.

Die Darstellungen in diesem Beitrag beruhen auf Beratungs- und Projekterfahrungen des Autors. Das Spektrum der Projekte reicht von IT-Projekten, mit denen agile Methoden im Unternehmen eingeführt wurden bis hin zu IT-Großprojekten, die nach unternehmenseigenen Vorgehensmodellen durchgeführt wurden. Die benannten Erfahrungswerte spiegeln sich jedoch auch in der angegebenen Literatur wieder.

Gewisse Eckpunkte, die ein Projekt definieren bleiben gleich, egal für welches Vorgehen man sich entscheidet. Man benötigt einen Projektauftrag, der in einer bestimmten Zeitspanne und einem zuerst grob definierten Scope, einer vereinbarten Qualität und mit einer bestimmten Menge an unterschiedlichsten Ressourcen und Methoden fertig gestellt werden soll. Scrum wirbt damit, leichtgewichtig zu sein. Dies ist ein zentrales Merkmal agiler Prozesse und Methoden. Das bedeutet, es wird zum Wohl des Produkts und der Kundenzufriedenheit auf ein Übergewicht in der Organisation, Verwaltung und Dokumentation verzichtet. Das mag aus klassischer Sicht eigenartig erscheinen, besonders wenn in Erwägung gezogen wird, dass weitere agile Maxime eine kontinuierliche Verbesserung (Continuous Improvement) und ein sehr hoher Qualitätsanspruch sind und als zentrale Kulturmerkmale verstanden werden (vergl. [SS13]). Es gibt Bereiche in denen es schwierig wird, diese Leichtgewichtigkeit durchgängig zu realisieren, denkt man z.B. an die Dokumentationsanforderungen in der Pharmaindustrie. Entwickelt sich ein Unternehmen in Richtung Agilität, was die Einführung agiler Methoden, Werte und Vorgehen bedeutet, ist Scrum ein Schritt auf dem Weg der agilen Transition.

Dieser Beitrag behandelt vier Themenbereiche, die beim Arbeiten mit Scrum bzw. einer Einführung von Scrum zu beachten sind. Es sind Bereiche, in denen sich moderne Rahmenwerke und Projektvorgehen von bisher bekannten klassischen Vorgehensmodellen wie z.B. das Wasserfallmodell oder dem Rational Unified Prozess (RUP) (vergl. [Ve00]) unterscheiden. Diese vier Themenbereiche sind:

1. Vorgehen und Prozessmanagement
2. Strukturen, Rollen, Kommunikation und Empowerment
3. Werte, Kultur und Führung
4. Projektmanagement

In den beiden folgenden Kapiteln wird aus zwei unterschiedlichen Blickwinkeln auf diese Themenbereiche geschaut. In Kapitel 2) ‚Scrum ist leicht‘, betrachtet der vorliegende Aufsatz die vier Themenbereiche daraufhin, welche Punkte davon leicht umzusetzen sind. Hingegen werden in Kapitel 3) ‚Scrum ist nicht einfach‘ die vier Themenbereiche dahingehend beleuchtet, welche der Punkte nicht einfach umgesetzt werden können. In Kapitel 4) ‚Lösungsansätze zur agilen Transition‘ sind Ansätze aufgezeigt, die eine Einführung von Scrum in den Organisationsalltag vereinfachen können. Im letzten Kapitel wird ein abschließendes Resümee gebildet.

2 Scrum ist leicht

Scrum ist ein Rahmenwerk (engl. Framework) für die Entwicklung von Software und kein Projektmanagementvorgehen.

Das wird gerne missverstanden. In Scrum werden **Vorgehen und Prozessmanagement** oder auch sehr vereinzelt Methoden aus dem Projektmanagement beschrieben (vergl. [PI07]). Dennoch gibt es bei Scrum weder Personalbeschaffung noch Risiko- oder Stakeholdermanagement. Hingegen sind Moderations- und Kommunikationsfähigkeiten wesentliche Kompetenzen für die Durchführung von Meetings (vergl. Tab. 1) bzw. Retrospektiven. Das wohl bekannteste Bild aus Scrum ist der Scrum Prozess (vergl. Tab. 1) selbst, der auf einen Bierdeckel passt. Dieses Bild skizziert nur den grundsätzlichen Ablauf des Scrumprozesses. Der Prozess ist nicht kompliziert und kann von jedem Team nach einer kurzen Eingewöhnungsphase umgesetzt werden. Er besteht aus einer einzigen Phase, die Sprint genannt wird. Weitere gibt es nicht. Ein Sprint hat immer die gleiche Länge, denselben Ablauf und als Sprintergebnis wird immer ein lauffähiges Softwareinkrement ausgeliefert. Der Sprint wird bis zum Projektende kontinuierlich in gleicher Länge wiederholt. Hinzu kommen vier Meetings und eine Dokumentationsform (vergl. Tab. 1) für die Anforderungen. Der Sprint und auch die Meetings haben sehr klar strukturierte Inhalte und Abläufe (vergl. [Wo12]). Als Dokumentationsform für das Anforderungsmanagement gibt es die User Story. In der User Story wird nur dokumentiert, was die Entwicklung und der Testbereich für den aktuellen bzw. den nächsten Sprint benötigen. Die Basisabläufe, Strukturen und Vorgehen des Sprints, der Meetings und der User-Stories sowie die damit verbundenen Philosophie können in der Standardliteratur (vergl. [SS13]) nachgelesen und im Rahmen einer Weiterbildung zum Scrum Master und Product Owner vertieft werden.

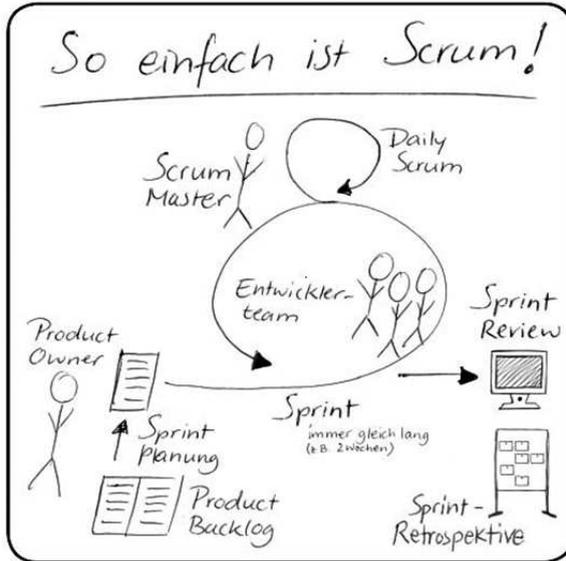


Abb. 1: Der Scrumprozess [Wo12]

Strukturen, Rollen, Kommunikation und Empowerment innerhalb des Scrum Teams sind klar definiert (vergl. [SS13]). Grundsätzlich sind alle Personen gleichberechtigt. Es gibt eine horizontale Struktur (vergl. Tab. 1), die die Aufgaben und die Verantwortung der Rollen (vergl. Tab. 1) definiert, aber keine disziplinarische Hierarchieebene oder Kommunikationsstruktur widerspiegelt. Die nicht vorhandene vertikale Hierarchie und die auf diverse Feedbackschleifen begründete Kommunikation erzeugt ein sehr offenes und von gegenseitigem Respekt begründetes Arbeits- und Kommunikationsklima. Dies lässt wenig Raum für Fehlinterpretationen und ist gleichzeitig die Basis für einen maximalen Wissenstransfer (vergl. Tab. 1). Das Scrum Team wird nach kurzer Zeit flexibel auf Veränderungen, die aus dem Anforderungsmanagement kommen, reagieren können. Es mag Teams oder Einzelpersonen geben, die mit derart viel Freiraum und der damit verbundenen Verantwortung und Selbstorganisation (vergl. Tab. 1) nicht sofort zurechtkommen. In der Regel entsteht jedoch nach kurzer Zeit eine intensive Gruppendynamik.

Die in Scrum gelebten **Werte, Kultur und Führung** unterstützen die Rollen, deren Aufgaben und Verantwortung. Die Werte sind im „Agile Manifesto“ (vergl. [Be01]) und „The Scrum Guide“ (vergl. [SS13]) festgehalten und werden von jedem agilen Team gelebt und auch geschützt. Ohne sie würde die agile Art zu arbeiten und zu kommunizieren nicht funktionieren. In einem Team, dessen Maxime Selbstorganisation, Lösungsorientierung, Vertrauen und Kooperation sind, gibt es nicht die große Notwendigkeit etwas „managen“ zu müssen. Das Team und jede einzelne Rolle hat eine hohe Verantwortung, höher als in klassischen Vorgehen. Der Führungsansatz (vergl. Tab. 1) des Product Owner und des Scrum Master liegen darin, die Arbeit des Scrum Teams zu fördern und zu unterstützen, damit es die hohen Erwartungen in jedem Sprint auch erreichen kann. Ent-

stehen Arbeitshindernisse, liegt es im Aufgabenbereich des Scrum Masters, diese schnellstmöglich für das Entwickler-Team zu beseitigen.

Einzelne Tätigkeiten, die sonst im klassischen **Projektmanagement** angesiedelt sind, gibt es ebenfalls. Arbeitspakete müssen priorisiert, geschätzt (vergl. [Pi07]) und auf die einzelnen Sprints eingeplant werden. Das geschieht im Sprintplanungsmeeting unter der Leitung des Product Owners, der die Kundenwünsche vertritt und den betriebswirtschaftlichen Erfolg des Projekts verantwortet. Nach jedem Sprint wird ein Sprintreview durchgeführt, neu priorisiert und entsprechend der aktuellen Velocity des Entwickler-Teams eingeplant. Auch die klare Gegenüberstellung zwischen geleisteter Arbeit und dem aktuellen Ergebnis wird durch den Product Owner erarbeitet und an die entsprechenden Stakeholder kommuniziert. Am Ende eines jeden Sprints steht ein lauffähiges Produktinkrement. Das ermöglicht eine sehr realistische Sicht auf den tatsächlichen Fertigstellungsgrad (vergl. Tab. 1). Die Aussage in klassischen Projekten, „das Projekt hat einen Fertigstellungsgrad in Prozent“, basiert auf der Fertigstellung einzelner Phasen, wie z.B. der Anforderungsphase. Diese Betrachtungsweise ist ungenauer und mit höherem Risiko zu bewerten als die Aussage über ein entwickeltes, getestetes und implementiertes Softwareinkrement. Die Tätigkeit des Product Owners im Projektmanagement konzentriert sich auf eine maximale Effizienz und Flexibilität auf Ebene der Arbeitspakete, nicht auf Verwaltungstätigkeiten und das pflegen von Plänen und Anforderungsdokumenten, die im aktuellen Projektstand noch keine Relevanz haben.

3 Scrum ist nicht einfach

Teams die es bisher gewohnt waren in einem klassischen Projektumfeld zu arbeiten und dadurch meist auch in Unternehmen mit vertikaler Organisationsstruktur, werden schon zu Beginn die Offenheit und die Effizienz schätzen, die sie im Scrum Alltag erleben. Allerdings wird ihnen das agile **Vorgehen und Prozessmanagement** etwas ungewohnt erscheinen. Schwierig wird es, wenn es das erste Scrum Team im Unternehmen ist und kein erfahrener agiler Coach zur Verfügung steht, der für das Team plant und auch die entsprechende Methodenkompetenz (vergl. Tab. 1) besitzt. Ein Agiler Coach ist keine explizite Scrum-Rolle. Erfahrungsgemäß empfiehlt es sich aber, die Einführung von Scrum durch einen erfahrenen Coach zu unterstützen. Dieser kann aus dem Unternehmen stammen oder extern hinzugezogen werden. Im Mittelpunkt steht seine langjährige praktische Erfahrung, die ggf. durch entsprechende Zertifikate nachweisbar sein sollte. Grundsätzlich benötigt ein Scrum Team bei den ersten Sprints ein Coaching, bis das Vorgehen, die Prozesse und die Kommunikation im Alltag vollständig gelebt werden. Der Einführungsprozess ist immer unterschiedlich und hängt von den gegebenen Projektrahmenparametern (vergl. Tab. 1) ab. Können die Teammitglieder vollständig im Scrum Team arbeiten oder werden sie über eine Matrixorganisation auch in anderen Projekten benötigt? Wie groß ist die Offenheit und Bereitschaft der Stakeholder gegenüber agilen Vorgehen? In welchem Maß ermöglicht die Organisation dem neuen Team Gestaltungsraum zur Eingewöhnung und Entfaltung? An die sehr konsequent moderier-

ten Meetings, wie z.B. dem Daily Scrum müssen sich viele erst gewöhnen. Man hat nur wenig Zeit, die Informationen der eigenen Arbeit den anderen strukturiert mitzuteilen und gleichzeitig alle Informationen der Teammitglieder für sich zu filtern und auch gezielt Fragen zu stellen. Zu Beginn wird der Scrum Master in den Meetings noch sehr viel moderieren müssen. Eine zentrale Aufgabe für den Scrum Master ist es, in jedem Sprint zuerst den vertikalen Durchstich (technische Realisierbarkeit) durch das vorliegende Produktinkrement zu schaffen. Erst danach werden Stories umgesetzt, die nicht zur zentralen technischen Achse des Produktinkrements gehören. Durch dieses Vorgehen werden Planungssicherheit und Beherrschbarkeit sowie ein zentraler Anteil am Risikomanagement unterstützt. Wird dieses Vorgehen in jedem Sprint angewandt, verspricht das die Verringerung des Gesamtrisikos und führt zu sehr frühem Erkennen von Fehlern in der fachlichen, technischen und architektonischen Planung und Konzeption. Dieses Vorgehen unterstützt insbesondere die Planungssicherheit von einander abhängigen Sprintzielen, wenn mehrere agile Teams in einem Projekt oder in hybriden Programmen (vergl. [Hi12]) koordiniert werden. Module müssen rechtzeitig fertig gestellt und integriert werden, um davon abhängige Softwareteile rechtzeitig zu bedienen.

Oft haben Organisationen die Herausforderung, für ein agiles Pilotprojekt die richtigen **Strukturen, Rollen, Kommunikation und Empowerment** zu definieren bzw. zu finden. Nicht einfach ist die Rollenbesetzung (vergl. Tab. 1) des Product Owners und des Scrum Masters. Viel zu häufig werden gerade diese beiden Positionen mit unerfahrenen Personen besetzt. Dadurch werden die ohnehin großen Herausforderungen bei der Einführung schnell zu unüberwindbaren Hürden. Oft muss das Pilotprojekt einen schnellen Erfolg vorweisen, um Akzeptanz zu finden. Scrum und andere agile Konzepte sind prädestiniert, schwierige Situationen durch hohe Effizienz und flexibles reagieren zu meistern. Das ist aber nur mit einem erfahrenen Scrum Team möglich, das auch das notwendige Empowerment (vergl. Tab. 1) besitzt. Ist das nicht gegeben, ist ein Scheitern, das dann gerne dem Vorgehen zugeschrieben wird, vorprogrammiert. Empowerment und Kommunikation sind entscheidende Punkte. In einigen Branchen werden IT und Fachbereich strikt getrennt. Auf Projektebene muss diese Trennung aber aufgehoben werden, um funktionsübergreifende Scrum Teams (vergl. Tab. 1) aufzubauen. Ein Scrum Master oder Product Owner, der nicht mit dem nötigen Empowerment ausgestattet ist oder nur wenig Erfahrung mit der sehr transparenten und offenen Art der Kommunikation agiler Teams hat, wird in der Umsetzung der erforderlichen Aufgaben nur wenig Erfolg haben. Dies gefährdet das gesamte Projekt. Oft tun sich klassische Organisationen schwer damit, einer neuen und für sie unklaren Rolle das nötige Empowerment zu geben und eine transparente Kommunikation zu fördern. Schwierig wird es auch, wenn ein bisheriger fachlicher Projektleiter als Scrum Master agiert und ein agiles Entwickler-Team anleitet oder als Product Owner ohne jegliche agile Erfahrung und Methodenkompetenz ein Scrum Team führen soll.

Gerne werden **Werte, Kultur und Führung** des agilen Vorgehens als nicht so wichtig erachtet. In zweitägigen Weiterbildungen zum Scrum Master oder Product Owner werden sie vermittelt, aber die Praxis zeigt, dass sich unerfahrene Product Owner und Scrum Master fast ausschließlich auf Methoden und Prozesse berufen, wie sie in der Fachlitera-

tur beschrieben werden. Damit handeln sie entgegen des ersten Kernsatzes des „Agile Manifesto - Menschen und Kommunikation sind wichtiger als Prozesse und Strukturen“ (vergl. [Be01]). Moderne Führung (vergl. Tab. 1) basiert auf Fördern, Vertrauen, Wertschätzung und auch los lassen. Moderne Führung sollte als Leistung im Dienst des Teams verstanden werden. Die durch agile Arbeiten entstehenden Möglichkeiten sind neben der Flexibilität und der Effizienz auch das Kreieren völlig neuer, kreativer Lösungen. Das benötigt Raum und Kommunikation - weniger Prozess und Struktur. In „Management Y“ (vergl. [Br14]) wird diese Art der Führung mit Douglas Mc. Gregors ‚Theorie X und Theory Y‘ gestützt. Das auf den Kopf gestellte Dreieck symbolisiert die neue Art des Führens. Die Mitarbeiter dienen nicht ausschließlich dem Management, sondern auch das Management dient den Mitarbeitern.

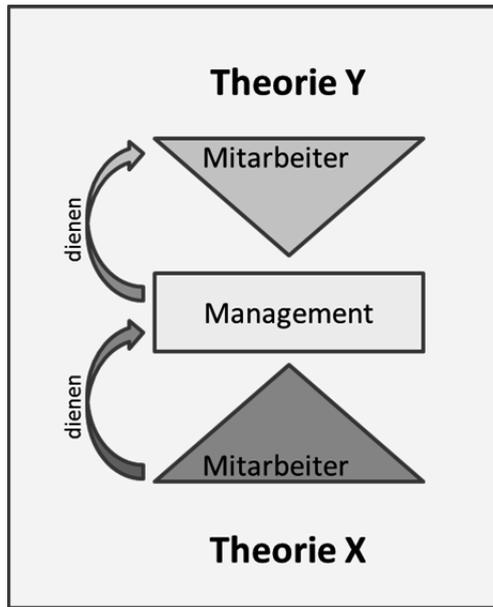


Abb. 2: Theory X und Theory Y [Br14]

In der gängigen Scrum Literatur gibt es wenig Aussagen zum **Projektmanagement** wie z.B. die im PMBOK (vergl. [Pm13]) beschriebenen Wissensgebiete Personalmanagement, Beschaffungsmanagement, Risikomanagement sowie Berichtswesen gegenüber Gremien und Lenkungsausschüssen. Zum Schätzen werden Methoden zur Sprintschätzung angeboten wie z.B. das Planning Poker. Eine Projektschätzung (vergl. Tab. 1) über das gesamte Projekt ist nicht vorgesehen. Geht es um ganze Programme oder die Einbettung in große Unternehmen und Konzerne, sind diese Themen unerlässlich, um ein oder mehrere Scrum Teams in die Organisation, in Multiprojekte oder Portfolien zu integrieren. Zu Beginn eines Projekts muss es eine agile Gesamtschätzung (vergl. [OW07]) aller Arbeitspakete geben. Der Erfolg hängt entscheidend von der Größe des Projekts und der

Erfahrung des Product Owners ab. Ein agiler Coach oder ein agiler Projektmanager ist an dieser Stelle einzubeziehen. Die Schätzung wird vom gesamten Scrum Team (bei z.B. Scrum of Scrums werden Vertreter aus jedem Scrum Team ausgewählt) unter der Anleitung des Product Owners bzw. des agilen Projektmanagers (vergl. [HK14]) durchgeführt. Es liegt nahe, dass diese Themen im Unternehmen von vorhandenen klassischen Projektleitern übernommen werden. Problematisch dabei ist, dass sich Scrum Teams und ihre Arbeitsweise nicht linear in eine klassische Projektplanung integrieren und steuern lassen. Beim Scrum of Scrums werden mehrere Scrum Teams gebildet, jedes Scrum Team ist für ein Produkt oder Teil eines Produkts verantwortlich und jedes Scrum Team hat auch einen Product Owner. Der Scrum Master ist in der Regel für mehrere Entwickler Teams parallel zuständig. Oft wird darüber noch ein Chief Product Owner installiert, der als zentrale Stelle alle Product Owner koordiniert. Diese Position könnte auch von einem agilen Projektleiter begleitet werden.

Oft sind unternehmensweite Vorgehensmodelle (vergl. Tab. 1) nicht auf das iterative Vorgehen agiler Teams angepasst. Sie beinhalten in der Regel Meilensteine oder Quality Gates, die zum Teil die gesamte Fachdokumentation sehr früh im Projekt erfordern. Klassische Projektleiter (vergl. Tab. 1) sind es auf Prozessebene noch nicht gewohnt in sehr kurzen Zyklen auf Anforderung, Entwicklung, Test und Integration zu steuern und zu planen. Oft fällt es schwer, die Idee des „managen“ los zu lassen und die Kontrolle für Abläufe und Kommunikation in die Teams abzugeben, zu fördern oder Hindernisse zu beseitigen (vergl. [BS08]). Das kann auch an Abteilungsgrenzen, einer strikten Trennung von IT und Fachbereich oder an Organisationshierarchien liegen. Wenn es eine Organisation gewohnt ist, große Projekte mit einem IT-Projektleiter und einem Fach-Projektleiter zu besetzen, müssten sie auf Teamebene dennoch versuchen interdisziplinäre Teams, bestehend aus Fach- und IT-Kräften, einzurichten. Ein Vorgehen nach Scrum macht sonst wenig Sinn. Projektsteuerkreise (vergl. Tab. 1) erwarten Kennzahlen, Projektstatus und Restaufwände. Diese basieren auf klassisch ablaufenden Phasen (Anforderungs-, Entwicklungs-, Test- und Integrationsphase). Ein Agiles Team hat wie in Kapitel 2) beschrieben eine veränderte Sicht auf diese Kennzahlen.

Folgende Tabelle gibt eine Übersicht über die in Kapitel zwei und drei behandelten Argumente innerhalb der vier Themenbereiche:

Themenbereich	Scrum ist leicht	Scrum ist nicht einfach
Vorgehen und Prozessmanagement	Scrum Prozess, Meetings, Dokumentationsform	Methodenkompetenz, Projektrahmenparameter
Strukturen, Rollen, Kommunikation und Empowerment	Rollen, horizontale Struktur, Wissenstransfer	Rollenbesetzung, Empowerment, funktionsübergreifende Scrum Teams
Werte, Kultur und Führung	Führungsansatz, Selbstorganisation	Moderne Führung
Projektmanagement	Fertigstellungsgrad	Projektschätzung, unternehmensweite Vorgehensmodelle, klassische Projektleiter, Projektsteuerkreis

Tab. 1: Wann ist Scrum leicht und wann nicht?

4 Lösungsansätze zur agilen Transition

Die Einführung von Scrum und agiler Methoden in ein Unternehmen sollte schrittweise durchgeführt werden. Dabei haben die Teams und Abteilungen Zeit, sich an die Methoden, Werte und Prozesse zu gewöhnen. Wird ein erstes agiles Pilotprojekt durchgeführt bzw. ein erstes Pilotteam von vier bis sieben Personen gebildet, ist im Unternehmen oft kein erfahrener Scrum Master oder Product Owner vorhanden. Die Rollen des Product Owners und des Scrum Masters sind die Schlüsselrollen in Scrum und müssen richtig besetzt werden. Interne fachliche Projektleiter können die Rolle des Product Owner übernehmen. Es kann dabei aber zu Diskussionen über Verantwortungsbereiche und Empowerment für Budget und Ressourcen kommen, was individuell zu klären ist. Etwas einfacher ist die Besetzung des Scrum Master. Es eignen sich erfahrene Entwickler und Softwarearchitekten - technisches Verständnis für die Prozesse der Softwareentwicklung sind sehr hilfreich. Zur Einführung bzw. Bildung eines agilen Teams ist es von Vorteil, dem Team einen agilen Coach zur Seite zu stellen. Ein agiler Coach kann bereits bei der Rollendefinition seine Erfahrungen einbringen und ein auf die vorhandenen agilen Kenntnisse zugeschnittenes Coaching aufbauen. Oft wird ein erstes agiles Team gebildet, aber die restlichen Stakeholder und Abteilungen sind sehr klassisch orientiert. Hier kann ein erfahrener Coach vermitteln und auch helfen, die richtigen Kommunikationsstrukturen aufzubauen. Die Unterstützung eines Coachs ist besonders zum Projektstart wichtig, wenn es um die Planung und Schätzung der Arbeitspakete und Ressourcen geht und das Team noch keine Sprinterfahrung hat. Die Positionen des Scrum Masters und des Product Owners benötigen zu Beginn ein stärkeres Coaching, das nach den ersten Sprints abnehmen wird. Die so gecoachten Rollen können ihr Wissen an andere Teams

weitergeben. So kann ein Wissenstransfer der agilen Teams beginnen. Je größer die Projekte sind die agil umgesetzt werden sollen, desto mehr Erfahrung und auch Coaching ist am Anfang notwendig. Zusätzlich müssen Rollen definiert bzw. umgestaltet werden, wie zum Beispiel ein Chief Product Owner oder ein agiler Projektleiter (vergl. [OW07]), der das klassische und das agile Vorgehen beherrscht. Er versteht es, die agilen Teams zu führen, die Gesamtprojektplanung und Koordination zu übernehmen und die erwarteten Zahlen Richtung klassischer Steuerkreis zu kommunizieren. Sollen Projekte mit mehr als zwei bis drei Scrum Teams durchgeführt werden, wird ein agiles Projektmanagement oder ein hybrides Programm zur Steuerung und Planung notwendig. Existiert ein unternehmensweites Vorgehensmodell, stellt sich die Frage, ob es agile Teams unterstützt oder evtl. eine Anpassung notwendig ist. Soll ein klassisches Projekt oder eine hierarchisch strukturierte Organisation mit agilen Methoden unterstützt werden, kann die Verwendung von IT-Kanban (vergl. [An12]) hilfreicher sein als Scrum. Ist es nicht möglich, funktionsübergreifende Teams - bestehend aus IT und Fachbereichen - aufzubauen, kann ebenfalls die Einführung agiler Visualisierungs- und Prozessverbesserungsmethoden aus dem IT-Kanban ein Lösungsansatz sein. Es hat den Vorteil, dass der Fokus auf dem Verbessern der bereits vorhandenen Prozesse und des Prozessflusses liegt und nicht auf der Veränderung von Strukturen, Rollen, Kommunikation und Vorgehen. IT-Kanban steuert und optimiert die sich im Fluss befindenden Arbeitspakete (Work in Progress ,WIP'). Eine weitere Möglichkeit ist es, in der Softwareentwicklung und den Projekten ein iteratives Vorgehen einzuführen. Der Einsatz von Kanban und die Entwicklung iterativer Vorgehen kann parallel ablaufen. Hat ein Team oder eine Organisation gelernt in Iterationen zu planen und zu koordinieren, ist der erste Schritt zu mehr Agilität bereits vollzogen und eine Entwicklung in Richtung Scrum kann die konsequente Folge sein.

5 Resümee

Der Aufsatz soll zeigen, dass die Herausforderung mit Scrum nicht in einem einfach zu skizzierenden Prozess liegt, sondern in der Veränderung der Arbeits-, Führungs- und Kommunikationskultur. Agile Vorgehen sind leichtgewichtig, dynamisch und flexibel. Oft beginnt die agile Transition in den Entwicklungsabteilungen und Projektebenen. Die Einführung und Integration in eine bestehende, durch hierarchische Strukturen definierte Organisation ist nicht einfach. Sie erfordert die Bereitschaft zu Beginn die richtigen Fragen zu stellen, ein klares Ziel auszugeben und die dafür notwendigen Veränderungen konsequent durchzuführen. In jedem Fall sollte die Einführung von Scrum mit erfahrenen Scrum Mastern, Product Ownern bzw. einem agilen Coach besprochen werden, um Misserfolge weitestgehendst zu vermeiden. Ein Blick in die Gegenwart der modernen Softwareentwicklung und IT-Infrastrukturen zeigt, dass in Unternehmen Begriffe wie Continuous Delivery, Continuous Integration und DevOps längst Einzug hielten. Um dies adäquat bedienen zu können, ist es von Vorteil wenn mindestens ein iteratives Vorgehen und Ansätze von agilen Teams eingeführt sind.

Literaturverzeichnis

- [An12] Andersen, D.J.: Kanban: Evolutionäres Change Management für IT-Organisationen. dpunkt Verlag, 2012
- [Be01] Beck et al.: Manifesto for Agile Software Development. Siehe <http://agilemanifesto.org/>, 2001.
- [Br14] Brandes, U. et al.: Management Y. Campus, 2014.
- [BS08] Buhse, W.; Stamer, S.: Enterprise 2.0: Die Kunst, loszulassen. Rhombos, 2008
- [Hi12] Hilmer, S.: Hybride Vorgehensmodelle für ein unternehmensweit einheitliches, flexibles Projektmanagement. In (Linssen, O.; Kuhmann, M. Hrsg.): Qualitätsmanagement und Vorgehensmodelle, 19. Workshop der Fachgruppe Vorgehensmodelle (WI-VM) der Gesellschaft für Informatik e.V. (GI). Shaker Verlag, Aachen, S. 129-138, 2012.
- [HK14] Hilmer, S.; Krieg, A.: Standardisierung vs. Kultur: Klassisches und agiles Projektmanagement im Vergleich. In: (Engstler M.; Hanser, E.; Mikusz, M.; Georg Herzwurm Hrsg.): Vorgehensmodelle 2014. GI-Edition, Lecture Notes in Informatics, Gesellschaft für Informatik, Bonn, S. 65-76, 2014.
- [Le07] Leffingwell, D.: Scaling Software Agility: Best Practices for Large Enterprises. Addison-Wesley, 2007.
- [OW07] Oestereich, B.; Weiss, C.: APM - Agiles Projektmanagement: Erfolgreiches Timeboxing für IT-Projekte. dpunkt.verlag, 2007.
- [Pi07] Pichler, R.: Scrum - Agiles Projektmanagement erfolgreich einsetzen. dpunkt Verlag, 2007.
- [Pm13] Project Management Institute: A Guide to the Project Management Body of Knowledge: PMBOK(R) Guide. Project Management Institute, 2013.
- [SS13] Sutherland, J.; Schwaber, K.: The Scrum Guide. Siehe: <http://www.scrumguides.org/docs/scrumguide/v1/Scrum-Guide-US.pdf>
- [TN86] Tekeuchi, H.; Nonaka I.: The new new product development game. Harvard Business Review, 1986.
- [Ve00] Versteegen, G.: Projektmanagement: mit dem Rational Unified Process. Xpert.press, 2000.
- [Wo12] Wolf, H.; von Solingen, R.; Rustenburg, E.: Die Kraft von Scrum. Addison-Wesley, 2012.