



Jochen Krebs

[E-Mail: mail@jochenkrebs.com] ist aktives Mitglied der „Agile Alliance“ und des Netzwerks für agiles Projektmanagement, für das er die Region New York City leitet. Außerdem ist er Direktor in der IT-Organisation eines Investment-Hauses an der Wall Street und Koautor des Buchs [Kre07-c].

SCRUMS IM BACKLOG

Der Begriff „Backlog“ ist für Produktbesitzer sowie für Scrum-Master und ihre Teams nichts Neues. Aber fragen Sie einmal einen Projektportfolio-Manager – vor allem, wenn er in einem Projektbüro arbeitet, das agile Projekte betreibt. Lassen Sie uns agile Konzepte und insbesondere Scrum einmal auf das Projektportfolio-Management anwenden. Mit anderen Worten, wir stecken Scrum selbst in den Backlog.

Unternehmensweite Agilität

In dem Artikel „The agile pyramid“ (vgl. [Kre07-a]) vertrete ich die These, dass agile Softwareentwicklung und agiles Projektmanagement durch das Portfolio-Management zu einer Unternehmensstrategie verknüpft werden. Wie in der Architektur so müssen auch hier alle Ebenen zueinander passen und aufeinander abgestimmt sein, um eine solide Struktur zu haben (siehe Abb. 1). Für unsere Art von Projekten bedeutet Struktur die Effektivität in der Zusammenarbeit und Realisierung.

Während es beim agilen Projektmanagement darum geht, einzelne Projekte zu planen und zu steuern, kümmern sich Portfolio-Manager um eine ganze Reihe von Projekten. Ein traditionelles Portfolio-Management zeigt gewisse Schwächen bei dem Versuch, die Vorteile der Agilität in die Unternehmensstrategie zu übersetzen und zu integrieren. Ist das nicht paradox? Firmen müssen immer agiler handeln, um

sich durchzusetzen, sich anzupassen und in ihrer Branche zu überleben, was neue Formen der Softwareentwicklung erforderlich macht. Die agile Softwareentwicklung hält ihr Versprechen, inkrementell zu liefern. Die Spitze der Pyramide ist also agil, die Basis ist agil – was ist dann das Problem beim agilen Fluss dazwischen?

Da Scrum bestimmte Aspekte des Projektmanagements und des Software-Engineerings agil macht, lege ich den Akzent auf den Projekt-Auswahlprozess in Organisationen, der oft Teil des Portfolio-Managements ist. Genau diese Organisationseinheit – entweder Linienmanagement oder Projektbüro (*Project Management Office, PMO*) – ist es, die Prozessrichtlinien und Standards für die produktiven Projekte aufstellt.

Wie lassen sich nun Scrum-Prinzipien auf das Portfolio-Management anwenden?

Zunächst einmal sollte die Agilität eines Scrum-Projekts (die unterste Ebene der

Pyramide) nicht einem bürokratischen Portfolio-Management-Prozess zum Opfer fallen. Solche überkommenen Arbeitsabläufe und Vorschriften wurden oftmals mit der Zeit standardisiert, ohne ihre Effektivität in Frage zu stellen. Zum zweiten muss das agile Portfolio-Management Agilität nicht nur tolerieren, sondern sie fördern. Das heißt, die nicht agilen Aspekte des Portfolio-Managements müssen angepackt und beseitigt werden. Und das erfordert einen strikten externen Führungsstab.

Portfolio-Scrum-Meeting

Man könnte beispielsweise dem Portfolio-Manager die Aufgaben des Scrum-Master übertragen. Anstatt der täglichen Scrum-Sitzungsfragen („Was haben Sie getan?“, „Was werden Sie tun?“, „Was steht dem im Wege?“) wenden sich agile Portfolio-Manager in einem wöchentlichen oder monatlichen Portfolio-Scrum mit folgenden Fragen an die Produktbesitzer (*Product Owner*):

- „Welche Fortschritte macht das Projekt?“
- „Was ist für den nächsten Sprint geplant?“
- „Welche Hindernisse gibt es für das Projekt oder den Sprint?“

Anstatt Kennzahlen zu produzieren, was im Projekt häufig einen manuellen Aufwand verursacht und als Belastung empfunden wird, verfügen die Produktbesitzer bereits über die *Burndown*- und Leistungswert-Informationen aus dem Backlog – und das sozusagen in Echtzeit. Mit weniger gefilterten und verzerrten Daten aus erster Hand gewinnt der Portfolio-Manager so wichtige Erkenntnisse über den Zustand

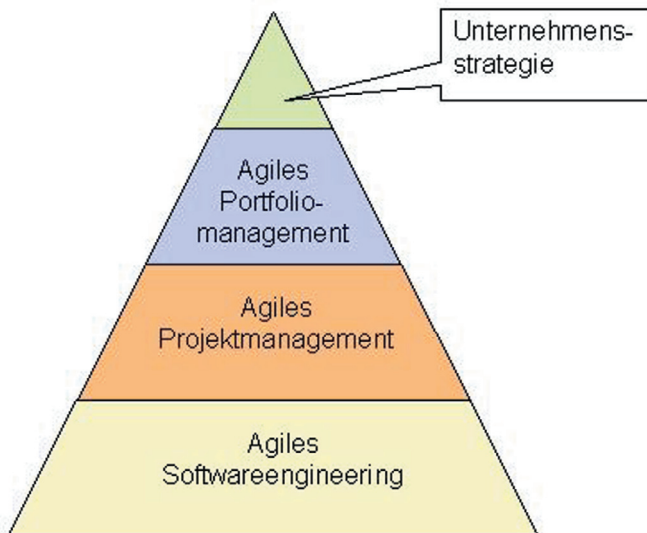


Abb. 1: Agile Pyramide (aus: [Kre07-a]).

eines jeden Projekts. Wie beim *Sprint Review Meeting* können Projekte den Portfolio-Managern sogar konkrete, funktionsfähige Fortschritte demonstrieren. Was noch wichtiger ist: Portfolio-Manager und Produktbesitzer können ihre Planung mit der Unternehmensstrategie in Einklang bringen. Hand in Hand kann so die Richtung der *Scrum*-Projekte bestimmt werden. Bei einem solchen Vorgehen könnten Führungsstab und *Scrum-Master* im *Portfolio-Scrum-Meeting* die Rolle der *Chickens* übernehmen, während die Produktbesitzer und Portfolio-Manager die *Pigs* sind.

Burndown-Diagramme

Portfolio-Manager verlangen häufig nach einer Visualisierung der Entwicklung ihres Portfolios, etwa in Form eines Dashboards oder eines Blasenendiagramms. Diese Diagramme enthalten – je nach Risiko, ROI, aktuellem Status der einzelnen Projekte usw. – Blasen unterschiedlicher Größe und Farbe. Ein Aspekt, der in *Scrum-Burndown*-Grafiken deutlich wird, fehlt jedoch solchen Blasenendiagrammen, nämlich der aktuelle Fortschritt und der Trend des Projekts.

In *Scrum* stellen die *Burndown*-Kurven den noch verbleibenden Aufwand auf der Zeitachse grafisch dar. Ob mit oder ohne Trendkurve – *Scrum-Master* und Portfolio-Manager bekommen ein klares Bild des Projektfortschritts. Ein Portfolio-Manager kann alle *Burndown*-Grafiken der einzelnen Projekte zu einem *Portfolio-Burndown* zusammenfügen. Mit ein wenig Kreativität können die Informationen aus den Blasenendiagrammen in den *Portfolio-Burndown* aufgenommen werden, zum Beispiel über die Dicke, Farbe oder Form der Linie. Auf diese Weise entsteht dem *Scrum-Team* keine zusätzliche Arbeit für das *Portfolio-Burndown*-Diagramm, das als Input für den Portfolio-Management-Prozess dient. Blasenendiagramme und Portfolio-Dashboards mögen zwar für den Führungsstab und das Projektbüro für die Chancen-Risiko-Analyse und das Ressourcenmanagement weiterhin von Interesse sein, aber die Verantwortung dafür sollte nun beim Projektbüro liegen und die *Scrum-Teams* nicht während eines *Sprints* von der Arbeit abhalten.

Backlogs

Portfolio-Manager steuern Portfolios, *Scrum-Master* kümmern sich um Projekte.

Beide überwachen wichtige Dinge, die sich grundsätzlich nicht sehr unterscheiden. Ein *Scrum-Backlog* enthält Einträge für beispielhafte Merkmale oder Produktanforderungen, etwa in Form von Anwenderberichten. Von einer organisatorischen Warte aus kann man das gesamte Projekt als ein Merkmal der Organisation oder eine Anforderung an die Organisation betrachten. Um es noch einmal klar zu machen: Es muss einen Grund dafür geben, dass sich dieses Projekt im Portfolio befindet und überhaupt erst angestoßen wurde. Ein *Produkt-Backlog* enthält Beschreibungen, Schätzungen und den noch verbleibenden Aufwand. Diese grundsätzlichen Informationen benötigen auch die Portfolio-Manager. „Der *Produkt-Backlog* entwickelt sich parallel zur Entwicklung des Produkts und der Umgebung, in der es verwendet wird“ (vgl. [Sch04]). Übertragen auf die Bedürfnisse eines Portfolio-Managers heißt das, dass sich das Portfolio parallel zur Entwicklung der Unternehmensstrategien und der Umgebung, in der es verwendet wird, entwickelt. Das ist die Geburtsstunde des agilen *Portfolio-Backlog*.

Sprint-Backlogs können Portfolio-Managern enormen Nutzen bringen, den sie möglicherweise bislang nicht ausgeschöpft haben. Jedes *Scrum-Team* bezieht die Arbeit für den kommenden *Sprint* aus dem *Produkt-Backlog*. Daher stellen die *Sprint-Backlogs* aller Projekte ein Planungsinstrument für den Portfolio-Manager dar und – was noch besser ist – der *Sprint-Backlog* ist bereits allgemein einsehbar. So kann der Portfolio-Manager leicht Einfluss nehmen und agile Projekte an eine geänderte Strategie anpassen, indem er den *Produkt-Backlog* des Projekts beeinflusst.

Die im *Portfolio-Scrum* gesammelten Informationen, der *Portfolio-Backlog* und die Planungsarbeit im *Sprint-Backlog* setzen die agile Energie im Portfolio-Management frei, nach der wir gesucht haben. Wenn die Informationen greifbar sind und der agile Prozess installiert ist, können wir uns noch einmal den Projekt-Auswahlprozess anschauen. Genau dieser Auswahlprozess und die Schaffung einer guten Balance für die Projekte ist mit die größte Herausforderung für einen Portfolio-Manager (vgl. [Kle01]). Wie lässt sich nun dieser Punkt mit agilen Projekten angehen?

Projekt-Auswahlprozess

Bei agilen Projekten im Portfolio sind die Entscheidungskriterien zur Genehmigung oder Ablehnung eines Projekts nicht mehr so eindeutig schwarz oder weiß (vgl. [Kre07-b]). Agile Portfolio-Manager setzen möglicherweise bereits Systeme ein, die beispielsweise nur zu 80% fertiggestellt sind und die noch zu Ende entwickelt werden oder auch nicht. Sie können Projekte einfrieren oder zu einem frühen Zeitpunkt abbrechen, um ein Ausbluten zu verhindern und die Ressourcen in vielversprechendere Projekte zu stecken. Dennoch wären diese Projekte kein Totalverlust, denn Teile der Funktionalität können noch in Betrieb genommen werden, andere werden nicht mehr realisiert. Natürlich bietet nicht jedes abgebrochene Projekt brauchbare Funktionalität in nutzbarer Form, aber agile Softwareentwicklung unterstützt dies.

Projekte einzufrieren, d. h. die Arbeiten für eine gewisse Zeit einzustellen und wieder aufzunehmen, ist für agile Projekte, die iterativ-inkrementell abgewickelt werden, kein Problem. Projekte können, auch wenn sie nur zum Teil fertiggestellt sind, früher zum ROI beitragen. Investigative Projekte (zum Beispiel Machbarkeitsstudien), die eine Durchführbarkeit prüfen, können aus dem Portfolio herausgenommen werden, wenn sich eine These als wahr oder falsch erwiesen hat. Der *Portfolio-Backlog* ist ein Fundus neuer Projektideen, die priorisiert, geschätzt und angestoßen werden können, wenn die Zeit dafür reif ist. All dies ist aus den Aktien-Portfolios der Finanzwirtschaft wohl bekannt: kurzfristige und langfristige Investitionen, risikoreichere Optionscheine (*Put/Call*), eine Liste zu beobachtender Aktien und der Verkauf lukrativer oder auch schlechter Anlagen. Portfolio-Manager schauen voraus, sie sind Visionäre und müssen eine Strategie mit den Projekten in Einklang bringen. *Scrum-Backlogs* und ein dynamischer Projekt-Auswahlprozess werden sie bei ihrer Arbeit unterstützen.

Aber nicht alle agilen Projekte sind ebenso innovativ und kreativ wie bahnbrechend. Manche Projekte und Programme braucht man, um politische oder juristische Vorgaben zu erfüllen. Obwohl die agilen Vorteile auch solcher Projekte ins Portfolio einfließen, ist eine harte Ja/Nein-Entscheidung hier unter Umständen gar nicht erwünscht. Außerdem können ganze Projektprogramme (ein Bündel verschiede-



Burndown Chart

Grafik, die den Projektfortschritt eines Produkts, *Sprints* oder *Releases* in einer Kurve visualisiert. Die Kurve gibt für jeden Punkt auf dem horizontalen Zeitstrahl an, wie viel Arbeit nach jeweils aktueller Schätzung zu jedem Zeitpunkt noch übrig ist, um das Ziel zu erreichen. Mittels einer auf den vergangenen Schätzwerten basierenden Trendlinie lässt sich das voraussichtliche zeitliche Ende des Projekts vorhersagen und eine Abweichung vom Zeitplan prognostizieren.

Chicken

Am Projekt interessierte, jedoch nicht direkt an der Umsetzung und am Projektrisiko beteiligte Person.

Pig

Bezeichnung für eine Person, die direkt am Projekt beteiligt ist, indem sie eine formale *Scrum-Rolle* (Teammitglied, Produktbesitzer, *Scrum-Master*) ausfüllt oder auf sonstige Weise zu den Projektbeteiligten gerechnet werden kann. Ein *Pig* trägt im Gegensatz zu einem *Chicken* ein Risiko im Projekt und ist „committed, not just involved“, also mehr als ein interessierter Außenstehender.

(Produkt-)Backlog

Eine priorisierte Liste von Anforderungen mit zeitlichen Schätzwerten für deren Fertigstellung. Die Einheit der Schätzwerte ist meist der Personentag. Je höher die Priorität einer Anforderung, desto genauer sind tendenziell die Schätzwerte, da der *Produkt-Backlog* im Wesentlichen nach absteigender Priorität abgearbeitet wird. Der *Produkt-Backlog* ändert sich im Lauf der Zeit: In einem *Sprint* realisierte oder verworfene Anforderungen werden entfernt, Schätzwerte aufgrund von Teilbearbeitung oder neuer Erkenntnisse aktualisiert, neue Anforderungen aufgenommen, Prioritäten verändert usw.

Produktbesitzer (Product Owner)

Die für die Pflege des *Produkt-Backlogs* verantwortliche Person. Der Produktbesitzer vertritt die fachliche Auftraggeberseite und somit sämtliche Projektbeteiligte. Er priorisiert die *Produkt-Backlog-Items* in einer Weise, die dazu geeignet ist, den Geschäftswert des Produkts zu maximieren. Beispielsweise überlegt er sich, welche Menge von Anforderungen als erste implementiert werden sollen, um ein frühes erstes *Release* des Produkts, verbunden mit einem frühen *Return On Investment (ROI)*, zu realisieren.

Scrum

Scrum ist kein Akronym, sondern die englische Bezeichnung für das angeordnete Gedränge im Rugby-Sport. Der Bezug zu *Scrum*

als Prozess für Projektmanagement und -entwicklung besteht darin, dass das Team im Mittelpunkt steht, sich selbst organisiert und jeweils vor dem nächsten *Scrum* den geplanten Spielzug bespricht.

Scrum-Master

Die für den *Scrum-Prozess* verantwortliche Person. Der *Scrum-Master* sorgt für die korrekte Implementierung und den maximalen Nutzen des Prozesses.

- Der *Scrum-Master* sollte kein Vorgesetzter der Teammitglieder sein.
- Der *Scrum-Master* sollte keine Doppelfunktion als Teammitglied oder Produktbesitzer haben. Interessenkonflikte wären die unvermeidliche Folge.
- Der *Scrum-Master* ist kein Projektleiter im herkömmlichen Sinn, sondern ein Vermittler und Unterstützer, dessen Hauptaufgabe darin besteht, Hindernisse für das Team aus dem Weg zu räumen.

Sprint

Eine gewisse Zeitspanne, innerhalb derer das Team alles tut, um sein Versprechen einzulösen, die in der *Sprint-Planungssitzung* ausgewählte Menge von Anforderungen als lauffähige Funktionalität umzusetzen und somit das *Sprint-Ziel* zu erreichen.

Während eines *Sprints* gibt es keine neuen oder geänderten Anforderungen von außen – der Produktbesitzer und alle anderen Projektbeteiligten sowie insbesondere die *Chicken* lassen das Team in Ruhe arbeiten.

Sprint Backlog

Eine Liste von Aufgaben, die den Arbeitsumfang des Teams für den *Sprint* festlegt. Die Liste präzisiert sich während des *Sprints* und wird täglich von allen Teammitgliedern gepflegt, sodass sie immer den aktuellen Bearbeitungsstand reflektiert. Der *Sprint-Backlog* ermöglicht es dem *Scrum-Master*, jederzeit zu erkennen, wo das Team steht und gegebenenfalls steuernd einzugreifen, damit das *Sprint-Ziel* nicht in Gefahr gerät.

Sprint Review Meeting

Sitzung am Ende eines jeden *Sprints*, bei der das Team – nicht irgendein übergeordneter Manager – dem Produktbesitzer und anderen interessierten Personen live am funktionierenden System zeigt, was es innerhalb des *Sprints* erreicht hat. Es darf nur wirklich fertige Funktionalität vorgeführt werden, keine Dummies oder Grafiken.

Kasten 1: *Scrum-Glossar* (frei nach www.scrum-master.de).

ner zusammenhängender Projekte) auch in eigenen Portfolios gesteuert werden.

Fazit

Agile Softwareentwicklung alleine bringt nicht automatisch alle Vorzüge, die mit Agilität erreicht werden können. Das gilt vor allem für große Organisationen, bei denen Projekt- und Portfolio-Management aufeinander abgestimmt werden müssen. In diesem Artikel ging es um die Anwendung

von *Scrum-Konzepten* auf eine Organisation durch die Veränderung des Portfolio-Management-Prozesses. Der Vorschlag zeigt, dass *Scrum-Prinzipien* sich nicht nur bis zur organisatorischen Ebene hochziehen, sondern – was noch wichtiger ist – in allen Ebenen sichtbar und spürbar sind. ■

Aus dem Englischen von Barbara Gruhn.

Literatur

- [Kle01] E. Kleinschmidt, R. Cooper, S. Edgett, *Portfolio Management for new products*, Perseus Publishing 2001
- [Kre07-a] J. Krebs, *The Agile Pyramid*, in: *Agile Journal*, Mai 2007
- [Kre07-b] J. Krebs, *Managing an Agile Project Portfolio*, in: *Agile Development Magazine*, April 2007
- [Kre07-c] J. Krebs, *Rational Unified Process – Reference and Certification Guide*, Addison-Wesley IBM Press 2007
- [Sch04] K. Schwaber, *Agile Project Management with Scrum*, Microsoft Press 2004