

# SCRUM

Prof. Dr. Eckhart Hanser, Duale Hochschule Baden-Württemberg (DHBW), D-79539 Lörrach

## Abstract

Dieser Artikel beschäftigt sich mit der agilen Projektmanagement-Methode Scrum. Nach einer Beschreibung der Projektrollen, des Prozessablaufs und der Artefakte (Dokumente), werden die Erfahrungen mit Scrum im studentischen Labor der Dualen Hochschule Baden-Württemberg beschrieben. Als Resultat wird die Erweiterung von Scrum um eine sozietäre Komponente diskutiert [1].

## 1 Einleitung

Im Gegensatz zu anderen agilen Prozessmodellen ist Scrum eine agile *Projektmanagement*-Methode. Der Ablauf des Projekts steht im Fokus, nicht die Struktur oder Zusammenarbeit des Teams. Scrum folgt der Idee des *Lean Managements* bei Toyota [2], auch wenn die Autoren immer wieder betonen, dass Scrum dort nicht seinen Ursprung hat. Scrum legt nur drei Projektrollen fest:

- Product Owner
- Team
- ScrumMaster

Andere Rollen, insbesondere solche, die sich aus der Entwicklungsarbeit im Team ergeben, werden nicht festgelegt. Außerdem regelt Scrum den Projektablauf als Abfolge von Sprints – Entwicklungszyklen von wenigen Wochen. Scrum folgt dem [Agilen Manifest](#).

## 2 Scrum – die Projektrollen

### Product Owner

Der Produktverantwortliche übernimmt die Sichtweise des Kunden. Er arbeitet eng mit dem Team zusammen und steuert die Entwicklung durch die Releaseplanung. Er allein bestimmt *wohin die Reise geht* [3]. Er hat folgende Aufgaben:

- **Anforderungsmanagement:** Die Anforderungen des Kunden werden erfasst und als Liste – das *Product Backlog* – zusammengeschrieben. Dieses bildet die Grundlage für Risikoanalysen und (grobe) Aufwandschätzungen.
- **Release Management:** Der Product Owner bestimmt durch die gewünschte Reihenfolge der Releases (mit ihrem jeweils definierten Funktionsumfang) maßgeblich den Erfolg des Produkts. Er ist allerdings gut beraten, den Release Plan mit dem Team abzusprechen. Die Aufwandschätzungen aus dem Team sind die Basis für seine Planung.
- **Kommunikation:** Der Product Owner muss ständig mit allen Beteiligten kommunizieren. Dies gilt insbesondere für Team und Kunde. Der Product Owner muss dem Team ausreichend (idealerweise jeden Tag) zur Verfügung stehen. Da in agilen Techniken i.d.R. keine ausführliche Dokumentation zur Verfügung steht, ist der Product Owner die einzige relevante Quelle des Teams, um die Anforderungen an das Produkt zu erfahren (abgesehen vielleicht vom Endkunden).

## Team

Das Team entwickelt („liefert“) das Software-Produkt. Scrum geht nicht auf die Rollenverteilung im Team ein, sondern sagt über seine Zusammensetzung lediglich:

- **Selbstorganisiert und klein:** Das Team organisiert sich während der Entwicklungsphasen – genannt *Sprints* – selbst. Nur die Teammitglieder entscheiden, welche Aufgaben zu lösen sind, um das Ziel des Sprints zu erreichen. Das Scrum-Team braucht keinen klassischen Projektleiter, weil es klein ist. Die Größe wird mit sieben  $\pm$  zwei Mitglieder angegeben [4]. Bedingung für alle Teammitglieder ist die ständige Präsenz im Team. Das Team muss sich als Einheit verstehen, die in Vollzeit miteinander an der Erfüllung der Sprintziele arbeitet. Große Teile der Kommunikation und damit des Wissentransfers laufen „osmotisch“, d.h. durch gleichzeitige Anwesenheit im selben Raum. Dies kann nicht gelingen, wenn einige der Teammitglieder nur in Teilzeit anwesend sind.
- **Multidisziplinär und autonom:** Die Mitglieder des Scrum-Teams müssen alle Fähigkeiten abdecken, die zum Erreichen der Projektziele erforderlich sind. Software-Architekten müssen genauso im Team sein, wie z.B. Programmierer, Qualitätssicherer und Tester. Die Teammitglieder dürfen aber nicht auf ihren traditionellen Rollen beharren. Grundsätzlich muss jedes Teammitglied alle Aufgaben im Projekt übernehmen (können). Das Scrum-Team muss *unabhängig* sein. Das definierte Sprint-Ziel muss und soll ohne externe Hilfe (oder Beeinflussung) erreicht werden. Ist dies nicht gewährleistet, muss die Struktur des Teams dahingehend verbessert werden.

Weiteres sagt Scrum über die Zusammensetzung des Teams nicht aus. Insbesondere wird nichts über die „sozietäre“ (organisationspsychologische) Zusammensetzung ausgesagt. Eine diesbezügliche Erweiterung soll am Ende dieses Artikels diskutiert werden.

## ScrumMaster

Der ScrumMaster ist der Prozessverantwortliche. Er coacht und berät das Team, wenn es um den Prozess geht. Er überprüft die Korrektheit des Prozessablaufs. Der ScrumMaster gehört immer nur zu einem Team gleichzeitig. Seine Aufgaben sind:

- **Scrum einführen und Product Owner und Team unterstützen:** Der ScrumMaster etabliert den Prozess im Team. Er identifiziert Hindernisse, die meist in alten und verkrusteten Denkweisen begründet sind. Er erklärt dem Team den Ablauf des Prozesses und coacht es. Er unterstützt aber auch den Product Owner. Es muss vermieden werden, dass dieser in eine Art Projektleiterrolle verfällt und das Team einschränkt. Insbesondere darf der Product Owner das Team während eines Sprints nicht mehr stören! Das muss der ScrumMaster auf jeden Fall verhindern.
- **Hindernisse beseitigen:** Scrum bezeichnet Hindernisse, d.h. Abweichungen vom Sollprozess als *Impediments*. Diese „Effizienz-Blockaden“ müssen gelöst werden. Oft ergeben sie sich aus falsch verstandenen Scrum-Rollen.
- **Dem Team dienen:** Der ScrumMaster führt das Team aufgrund seines Wissensvorsprungs bzgl. des Prozesses. Seine Aufgabe ist es, Hindernisse aus dem Weg zu räumen, nicht dem Team Aufgaben zuzuweisen. Er sollte einen kollegialen Führungsstil pflegen.

### 3 Scrum – der Prozess und seine Dokumente (Artefakte)

Scrum folgt einem einfachen Prozessablauf (Abbildung 1):

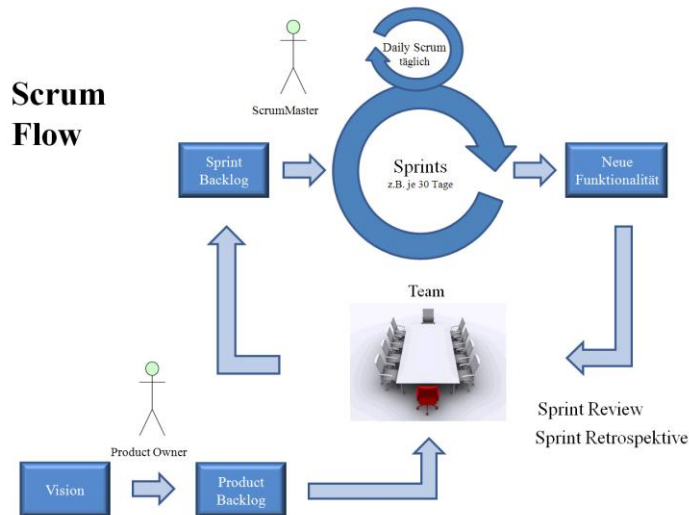


Abb. 1.: Scrum Flow

Nachdem der Product Owner aus der *Vision* des Produkts das *Product Backlog* erzeugt hat (das erste *Scrum-Artefakt*, Abbildung 2), plant er zusammen mit dem Team den ersten Sprint in einer Planungssitzung.

Priorität	Thema	Beschreibung	Akzeptanzkriterien	Geschätzter Aufwand [in Personentagen]
3	Eingabemaske Abonnenten	Die Eingabemaske muss um das Feld "Kampagne" erweitert werden.	Es dürfen nur gültige Kampagnen eingegeben werden.	1
1	Versandsystem	Versand muss unterbrechbar gemacht werden	Abgebrochener Versand muss an dieser Stelle wieder aufgenommen werden.	5
2	Anlegen neue Kampagne	Neue Maske "Kampagne" mit Zuordnung zu Newslettern (Listbox zur Auswahl)	Kampagne muss nach Anlegen sichtbar sein	3

Abb. 2.: Typisches Product Backlog für ein Newsletter-Versandsystem

Diese *Sprint-Planungssitzung* eröffnet den Sprint. Der Product Owner wählt die Anforderungen aus dem Product Backlog, die er als erstes implementiert haben möchte. Das Team schätzt den zeitlichen Aufwand und prüft, ob diese Anforderungen in einer vorher festgelegten Sprint-Länge realisiert werden kann (*Timeboxing*). Es ist wichtig, zu beachten, dass die einmal festgelegte Sprintlänge auch für *jeden* der nachfolgenden Sprints gilt und unbedingt eingehalten werden muss. (Üblich sind Sprintlängen von einer Woche bis zu 30 Tagen.)

Hat sich das Team und der Product Owner auf den zu implementierenden Funktionsumfang geeinigt, wird das *Sprint Backlog* erstellt (ein weiteres Scrum-Artefakt). Oft geschieht das in Form von User Stories

(Abbildung 3). Dabei wird aus jeder Anforderung eine eigene Story. Diese werden bei Bedarf um weitere *Task Cards* erweitert.



Abb. 3.: Sprint Backlog als User Stories an einer Stellwand

Ist das Sprint Backlog erstellt, beginnt das Team mit dem eigentlichen Sprint. Hier entwickelt das Team ohne Störung von außen das neue Software-Inkrement (die neue Release). Innerhalb des Sprints gibt es ein Ritual, welches täglich stattfindet – die *Daily Scrum*-Sitzung. Unter Anwesenheit des ScrumMasters (obligatorisch) und ggf. des Product Owners macht das Team eine kurze, viertelstündige, Standortbestimmung [4]. Jedes Teammitglied beantwortet drei Fragen:

- Was habe ich seit dem letzten Daily Scrum für das Projekt getan?
- Was will ich bis zum nächsten Daily Scrum für das Projekt tun?
- Was hindert mich daran, so effektiv wie möglich zu arbeiten?

Hinderungsgründe (sog. *Impediments*) werden vom ScrumMaster notiert und möglichst beseitigt. Wichtig ist auch, dass der ScrumMaster verhindert, dass sich das Team in Diskussionen verstrickt. Es geht nur um das Aufzählen von Hindernissen, nicht um das Lösen des Problems! (Oft ist in solchen Fällen einiges psychologisches Geschick des ScrumMasters nötig.) Die Lösung eines aufgetauchten Problems sollte erst nach der Daily Scrum in Angriff genommen werden.

Am Ende des Sprints steht das *Sprint Review*. In dieser maximal vierstündigen Sitzung [4] wird das entstandene Software-Inkrement vom Product Owner überprüft und abgenommen. Es werden nur vollständige und fehlerfreie Arbeitsergebnisse akzeptiert. Der Projektfortschritt wird in Form eines weiteren Artefakts, der *Release Burndown Chart*, dokumentiert (Abbildung 4). Bei größeren Projekten empfiehlt es sich zudem, dass der Product Owner einen *Releaseplan* erstellt – ein Scrum-Artefakt, in dem die gewünschten Releases und die dazu notwendige Abfolge der Sprints dokumentiert sind.

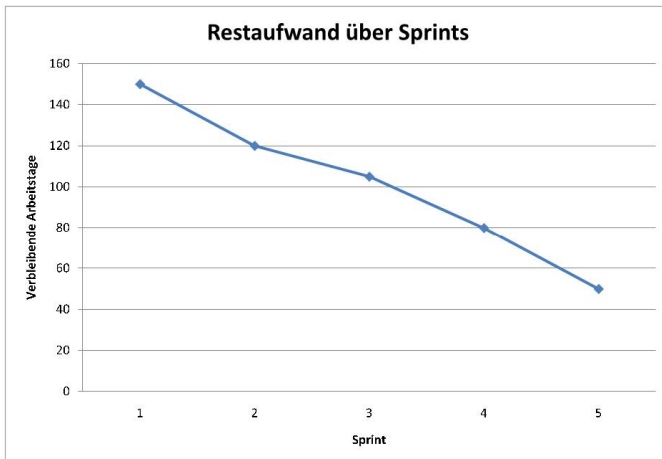


Abb. 4.: Release Burndown Chart

An das Sprint Review schließt sich direkt die *Sprint Retrospektive* an. Auch diese Teamsitzung ist „time-boxed“. In maximal drei Stunden [4] hat das Team Gelegenheit, über den Ablauf des Sprints nachzudenken, insbesondere über die Zusammenarbeit im Team und über notwendige Verbesserungsmaßnahmen. Diese fließen in den nächsten Sprint ein. Teilnehmer dieser Sitzung sind das Team und der ScrumMaster. Der Product Owner *kann* teilnehmen. Der ScrumMaster moderiert die Sitzung.

#### 4 Scrum im studentischen Labor

Die Erfahrungen mit Scrum in studentischen Projekten zeigen, dass die korrekte Besetzung der Rollen eine zentrale Voraussetzung für den Erfolg eines Projekts ist. Da die Rolle des ScrumMasters typischerweise mit dem Dozenten (d.h. dem Autor dieses Artikels, der ja auch das Prozesscoaching in der Vorlesung übernimmt) besetzt ist, ist es besonders die Rolle des *Product Owners*, die mit großer Sorgfalt besetzt werden muss. Hierzu wurde vom Autor das [Meta Agile Process Model \(MAP\)](#) entwickelt, welches in einem anderen Artikel des Methodenwürfels behandelt wird. MAP kann z.B. die *Eignung* eines Studierenden zum Product Owner mit erstaunlicher Präzision vorhersagen! Das Zusammenspiel von MAP mit Scrum soll im nächsten Kapitel näher beleuchtet werden.

Das studentische Labor zeigt deutlich, dass in Scrum-Projekten die Gefahr der Überbeanspruchung des studentischen Product Owners besteht. Er ist das „Nadelöhr“, das hauptsächlich den Kontakt zum Kunden hält, was ihn stark beansprucht. Tatsächlich mussten die anderen Studierenden des Teams in einem Projekt mit externem Kunden in einer zeitkritischen Phase auf die Entscheidung des Product Owners bzgl. der Anforderungen des nächsten Sprints *warten*. Das darf natürlich nicht passieren, weil es den Zeitplan des Projekts direkt beeinflusst.

Im studentischen Labor zeigte sich auch, dass der Prozess um die Möglichkeit erweitert werden musste, weitere User Stories *während* eines Sprints in das Product Backlog zu integrieren. Dazu wurde die neue Projektrolle des *Story Koordinators* geschaffen, welcher diese Scrum-Erweiterung durchführt. (Andernfalls hätten einige Teammitglieder bis zum nächsten Sprint keine Arbeit mehr im Projekt gehabt – sicherlich ein unhaltbarer Zustand!)

Daily Scrums wurden vom studentischen Team als sehr nützlich erlebt. Insbesondere die Länge anderer Scrum-Sitzungen wurde aber in Frage gestellt.

Zusammenfassend lässt sich sagen, dass Scrum im studentischen Labor – übrigens wie auch alle anderen agilen Prozessmodelle – verändert und angepasst wurde. Noch kein Prozessmodell hatte diesbezüglich in seiner „Urform“ Bestand! (Mehr Hintergrund zu diesen Beobachtungen findet man in [1].)

## 5 Erweiterung von Scrum um eine „sozietäre“ Komponente

Das *Meta Agile Process Model (MAP)* ist der Ansatz, mit dessen Hilfe Scrum um eine sozietäre Komponente erweitert werden kann. Die in MAP beschriebene, angestrebte Zusammensetzung des Projektteams, inkl. Product Owner, ist der Schlüssel zum Erfolg – so zeigen es die Ergebnisse des studentischen Labors!

Wie in [1] ausführlich diskutiert wird, ist MAP der übergeordnete Meta-Prozess, dessen Fokus darauf gerichtet ist, die richtigen „Typen“ (oder auch *Verhaltensweisen*) ins Team zu bringen, während dieses so zusammengestellte Team Scrum als konkreten agilen Prozess „fahren“ kann. Da die in Scrum- und in MAP definierten Prozessrollen weitgehend kompatibel sind (siehe [1], Kap. 7.6: MAP hat lediglich noch zusätzlich den *Integration Engineer*) stehen sie diesem Vorgehen nicht im Weg. Der Scrum-Flow wird im Ablauf lediglich um zwei (rot markierte) Prozessschritte erweitert (Abbildung 5)

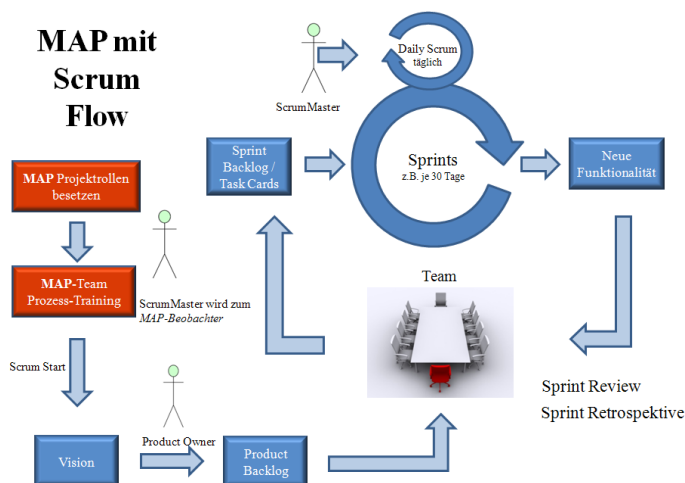


Abb. 5.: Scrum-Flow mit MAP-Erweiterungen

Der Fokus der Scrum-Erweiterungen liegt somit auf der *Vorhersage* der Eignung eines Teammitglieds für eine erfolgreiche Mitarbeit im betrachteten Projektteam und ggf. auch für eine spezielle Projektrolle. Hierfür wurde vom Autor ein Verfahren zur Eigen- und Fremdevaluation der Teammitglieder bezogen auf wichtige Projekteigenschaften entwickelt (Abbildungen 6 und 7).

Haupteigenschaften		bereit zur Kommunikation	erfahren	belastbar	allg. menschl. Eigenschaften
<b>MAP</b>	Kreuz setzen	X	X		
visionär					
kreativ	x				
kostenbewusst	x				
kritisch/hinterfragend	X				
strukturiert	X				
analytisch	X				
planend	X				
umsetzungsorientiert					

projektarbeitsbezogene menschl. Eigenschaften

3 Nennungen  
2 Nennungen

Abb. 6.: Formular für Selbst-/Fremdeinschätzung (gewichtet)

Haupteigenschaften		bereit zur Kommunikation	erfahren	belastbar	allg. menschl. Eigenschaften
<b>MAP</b>	Kreuz setzen	X	X		
visionär			Ideengeber		
kreativ	x				
kostenbewusst	x		Qualitätsprüfer		
kritisch/hinterfragend	X			Techn. Problemlöser/Troubleshooter	
strukturiert	X	Kommunikator			
analytisch	X				
planend	X		Experte/Fachspezialist		
umsetzungsorientiert				Teamworker	

projektarbeitsbezogene menschl. Eigenschaften

Abb. 7.: Auswertung: Landkarte der Verhaltensweisen des späteren Product Owners im studentischen Labor („Typisierung“ des Teammitglieds)

Ausgehend von der Auswertung dieser Evaluationen können ein oder auch mehrere Projekttypen (Verhaltensweisen) bestimmt werden, die das Teammitglied mit in das Projekt bringt (Näheres dazu in [1], Kap. 7. Ein Product Owner, der für diese Rolle aufgrund der Auswertung der Evaluation eine hohe „Affinität“ hatte, war im studentischen Projekt 2009 der Garant für den Erfolg des Produkts. Dies wurde von den anderen Teammitgliedern bestätigt.

## 6 Literatur

- [1] Hanser E (2010) Agile Prozesse: Von XP über Scrum bis MAP. Heidelberg: Springer
- [2] Gloger B (2009) Scrum – Produkte zuverlässig und schnell entwickeln. München: Hanser
- [3] Pichler R (2008) Scrum – Agiles Projektmanagement erfolgreich einsetzen. Heidelberg: dpunkt Verlag
- [4] Schwaber K (2004) Agile Project Management with Scrum. Washington, DC: Microsoft Press

## 7 Autor

Prof. Dr. Eckhart Hanser ist Studiengangsleiter „Biosystem-Informatik“ (Life Science) und als hauptamtlicher Dozent für Software-Engineering und -Qualitätssicherung im Studienbereich Technik der Dualen Hochschule Baden-Württemberg (DHBW) Lörrach tätig. Er liest außerdem in den Studiengängen Elektrotechnik und

---

Informationstechnik. Die Idee des studentischen Labors der DHBW Lörrach entstand aus einer 20jährigen Projekterfahrung im Projektmanagement und Qualitätsmanagement. Der Autor sammelte seine praktischen Erfahrungen sowohl im streng validierten Pharma-Umfeld als auch im Bereich der Software-Entwicklung für Internet-Marketing.