

Erweiterung agiler Prozesse durch sozietaere Vorgehensbausteine

Prof. Dr. Eckhart Hanser

Duale Hochschule Baden-Wuerttemberg (DHBW) Loerrach

hanser@dhbw-loerrach.de

Abstract: Im Folgenden wird das neuartige agile Meta-Modell MAP (Meta Agile Process Model) vorgestellt, welches u.a. eine Evaluationsmethode zur Bestimmung der menschlichen Verhaltensweisen (Typen) im agilen Team beinhaltet. MAP ermöglicht es, Projektrollen optimal zu besetzen, was maßgeblich dazu beitraegt, die Projekt- und Produktqualitaet zu hoeehen. Da MAP zwei Rekursionsebenen des neurokybernetischen Viable System Models (VSM) implementiert, kann gezeigt werden, dass MAP-Teams im Sinne einer systemischen Beschreibung lebensfaehige Systeme darstellen, die ueber entsprechende organisatorische Intelligenz zur Weiterentwicklung des Teams und zur Bewaeltigung kritischer Projekte verfuegen. Am Beispiel der agilen Projektmanagementmethode Scrum wird gezeigt, wie die Qualitaet in einem agilen "Standard-Projekt" durch das Einfuegen sozietaerer Vorgehensbausteine hoeoht werden kann. Dies wird durch die systematische Analyse eines geeigneten agilen Projektteams illustriert.

1 Verhaltensweisen (Typen) im Team – das Meta Agile Process Model (MAP)

Die Organisationspsychologie weiß seit längerem, dass die richtige „Mischung“ von Denk- und Verhaltensmustern von Teammitgliedern ueber den Erfolg eines Projekts mitbestimmt. Arbeiten von Baldegger und Gotsman [BG01] oder Belbin [Bel04] beschaeftigen sich mit diesem Thema. Allerdings sind die fuer den Erfolg eines Teams notwendigen Verhaltensweisen der Teammitglieder abhaengig vom Tuetigkeitsfeld des Teams. Ein Managementteam braucht eine andere Mischung der Verhaltensweisen als ein Software-Entwicklungsteam. Diese Luecke schliesst der Autor mit seinem kuerzlich veroeffentlichten *Meta Agile Process Model* (MAP) [Ha10][Ha08a], welches eine „Landkarte“ der Verhaltensweisen im Software-Entwicklungsteam entwickelt [Abb. 1] und in der Diskussion mit dem Organisationspsychologen Baldegger [Ba04] entstanden ist. Diese „Landkarte“ basiert auf der mehrjaehrigen Beobachtung erfolgreicher studentischer Teams an der Hochschule des Autors, die in realen Software-Entwicklungsprojekten ihren Erfolg unter Beweis stellen konnten [Ha08b].

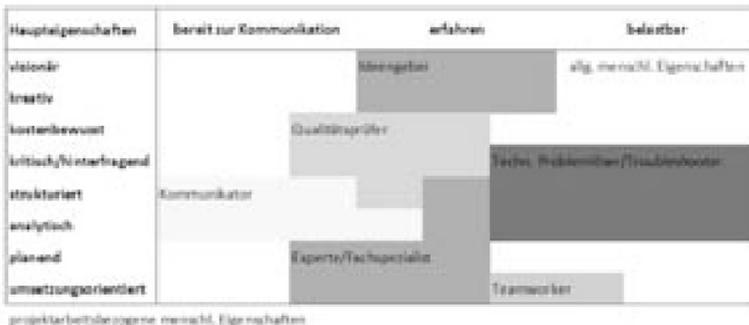


Abbildung 1: „Landkarte“ (MAP) der verschiedenen menschlichen „Typen“ (Verhaltensweisen) im Entwicklungsprojekt

Abbildung 1 ist wie folgt zu „lesen“: In der oberen Zeile von links nach rechts stehen allgemeine Eigenschaften der Teammitglieder, die der Autor in seiner langjährigen Projektstätigkeit als für den Projekterfolg wichtig identifiziert hat. In der Spalte vorne stehen hingegen projektbezogene menschliche Eigenschaften, die weitgehend den von Baldegger identifizierten wichtigen Eigenschaften von Teammitgliedern entsprechen ([Ba04] S. 22). Speziell für Entwicklungsteams müssen sie aber um die Eigenschaften *strukturiert* und *umsetzungsorientiert* erweitert werden, während die von Baldegger identifizierte Eigenschaft erfahren sinnvollerweise den allgemeinen menschlichen Eigenschaften zugeordnet wird.

Aus den verschiedenen menschlichen Eigenschaften der Teammitglieder leitet MAP „Typen“ von Teammitgliedern ab, die für die spätere Besetzung der Projektrollen bedeutend sind:

- Kommunikator
- Ideengeber
- Qualitätsprüfer
- Technischer Problemlöser (Troubleshooter)
- Experte (Fachspezialist)
- Teamworker (Entwickler).

Diese Typen ergeben sich aus der Kombination allgemeiner menschlicher Eigenschaften und projektbezogener Eigenschaften der Teammitglieder. In der „Landkarte“ in Abbildung 1 überlappen sich die resultierenden Typen zum Teil.

Es ist wichtig, darauf hinzuweisen, dass jedes Teammitglied mehrere Typen repräsentieren kann. Auch kann (und wird) sich der „Typ“ eines Teammitglieds im Laufe der Jahre verändern. Die Diskussion mit Organisationspsychologen [Wa08] ergibt folglich, dass es besser ist, von *Verhaltensweisen* anstatt von „Typen“ zu reden. Damit sind aber Kombinationen grundlegender menschlicher Eigenschaften gemeint und nicht „antrainierte“ Verhaltensmuster. Der Einfachheit halber wird im weiteren Text *Typ* wie auch *Verhaltensweise* synonym benutzt.

Wie in [Ha10] (siehe Kap. 7.2) gezeigt wird, entstehen die MAP-Typen bzw. MAP-Verhaltensweisen aus der Analyse eines besonders erfolgreichen Teams des Autors im Jahr 2005. Alle MAP-Typen sind in diesem Team vorhanden! Es ist aber auch möglich, die MAP-Typen aus den 9 Team-Rollen von Belbin [Bel07, Tab. 3.1, S. 22] herzuleiten (siehe auch *Henley-Studie*, [Bel04]). Hier muss allerdings berücksichtigt werden, dass Belbin Managementteams und keine Entwicklungsteams analysiert (Tabelle 1).

Team-Rollen nach Belbin	MAP-Typen (-Verhaltensweisen)
Co-ordinator & Resource investigator	⇒ Kommunikator
Plant (“ <i>Kreativist</i> ”)	⇒ Ideengeber
Completer & Monitor evaluator	⇒ Qualitätsprüfer
Shaper (“ <i>Gestalter</i> ”)	⇒ Technischer Problemlöser (Troubleshooter)
Specialist	⇒ Experte (Fachspezialist)
Teamworker & Implementer	⇒ Teamworker (Entwickler)

Tabelle 1: Abbildung der Team-Rollen nach Belbin auf die MAP-Typen (-Verhaltensweisen)

- Kunde
- Kommunikationsmanager
- Integrationsingenieur
- Team
- MAP-Beobachter (MAP-Observer)

Während der agile Kunde eine externe Projektkontrolle ist und dem Team hinreichend zur Verfügung stehen muss um z.B. den Funktionsumfang eines Software-Produkts abzuklären und Rückfragen zu beantworten, hat das Team die herausragenden Rollen *Kommunikationsmanager* und *Integrationsingenieur* zu besetzen: Der oder die Kommunikationsmanager übernehmen dabei typischerweise Projektleitungs- oder Qualitätssicherungsfunktionen, während der Integrationsingenieur die Software-Integration durchführt. Der Prozess der Software-Integration wurde im studentischen Labor als besonders wichtig und kritisch identifiziert. Nur Teams, die die Projektkontrolle Integrationsingenieur besetzten, konnten dabei zufriedenstellende Resultate erzielen. Natürlich entstehen Konflikte zwischen dem Integrationsingenieur und dem Team, aber diese dienen dem Projekt, weil der Kunde seine Entscheidungen zu jedem Zeitpunkt auf der Basis des aktuellen Stands des Produkts fällen kann.

Der *MAP-Beobachter* schließlich coacht das Team und begleitet den Prozess. Dabei ist der MAP Cycle in der Regel nur der initiale Prozess, der vom Team im Verlauf des Projekts mehr und mehr *abgewandelt* oder gegebenenfalls auch ausgetauscht wird. Das ist legitim, wenn dadurch Prozess und Produkt verbessert werden.

3 Das MAP-Team als System – Integration in das Viable System Model

Die bisherige Praxis der MAP-Teams mit ihrem sich im Laufe des Projekts anpassenden agilen Prozess zeigt, dass zum Verständnis des neuen Ansatzes nicht der (agile) Prozess im Mittelpunkt stehen sollte, sondern das Team selbst mit all seinen internen und externen Wechselwirkungen. Das MAP-Team ist also ein soziales System, welches über Steuerungs- und Regelungsmechanismen für einen erfolgreichen Projektverlauf verfügt. Diese Mechanismen stehen auf mehreren Ebenen zur Verfügung, sowohl auf der Ebene des Teams als auch auf der Ebene des einzelnen Teammitglieds mit seinen verschiedenen MAP-Verhaltensweisen (also den identifizierten Typen aus Abbildung 1).

Mit dem Viable System Model (VSM) von Stafford Beer [Be79][Be81] steht ein geeignetes neurokybernetisches Modell zur Verfügung, um das MAP-Team „systemisch“ zu beschreiben: Während sich die MAP-Teams eines Unternehmens auf Ebene 1 wie andere agile Projektteams in das Unternehmen eingliedern (siehe hierzu [WH12]), kann im Folgenden das MAP-Team im Rahmen des VSM auch auf der Team-Ebene (Ebene 2) und der Ebene 3 der individuellen Verhaltensweisen (MAP-Typen) mit seinen Wechselwirkungen, sowie Steuerungs- und Regelungsmechanismen dargestellt werden.

Das Viable System Model bildet die Basis des „St. Galler Managementkonzepts“ [WH12] (siehe hierzu auch [Sc00]). Es beschreibt ein Unternehmen mit seinen Strukturen und seinen Projektteams als lebensfähiges System und orientiert sich dabei stark am Aufbau des zentralen Nervensystems: Das *Operative Subsystem* (im biologischen Organismus wären das z.B. Muskeln oder Organe) interagiert mit seiner *Umwelt*, überwacht von einem *Metasystem*, welches in der Biologie dem Nervensystem und dem Gehirn entspricht [Wal12]. Das Metasystem wird in die *Systeme 2 bis 5* unterteilt, während das operative Subsystem als *System 1* bezeichnet wird. Während System 2 die Systeme 1 koordiniert, ist das System 3 das operative Management, welches durch die Audit- und Monitoringfunktionen des Systems 3* erweitert

wird. System 4 interagiert mit der Umwelt und beherbergt das strategische Management, während System 5, das normative Management, für die Wertesystem des Teams und der Teammitglieder zuständig ist.

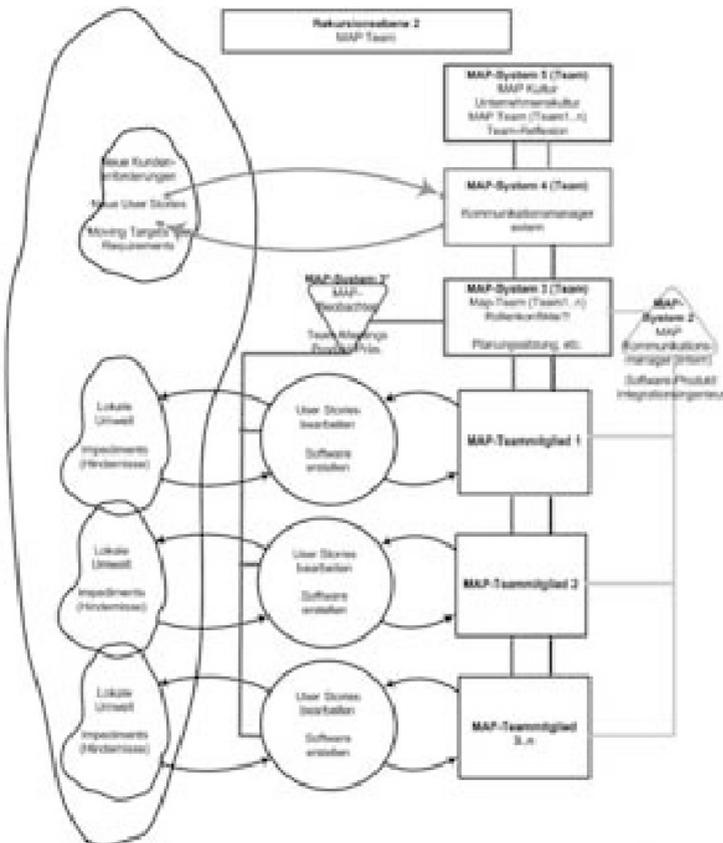


Abbildung 3: VSM-Rekursionsebene 2 – MAP Team

Abbildung 3 zeigt die Rekursionsebene 2 des Viable System Models [HW12]: Die MAP-Teammitglieder sind die operativen Systeme 1, welche durch das System 2, bestehend aus internen Kommunikationsmanagern, koordiniert werden. Bei der Software-Produktion übernimmt die Koordinationsfunktion des Systems 2 der Integrationsingenieur. System 3, das operative Management, ist das Team selbst, welches sich im Rahmen seiner Planungssitzungen etc. selbst steuert. Unterstützt wird es dabei durch den MAP-Beobachter (System 3*), der das Team coacht. Der oder die externen Kommunikationsmanager halten den Kontakt mit dem Kunden, um neue oder sich ändernde Spezifikationen (User Stories) zu identifizieren. Das normative System 5 entspricht der „MAP-Kultur“ des Teams, welche dieses in den Team-Reflexionssitzungen pflegt und weiterentwickelt.

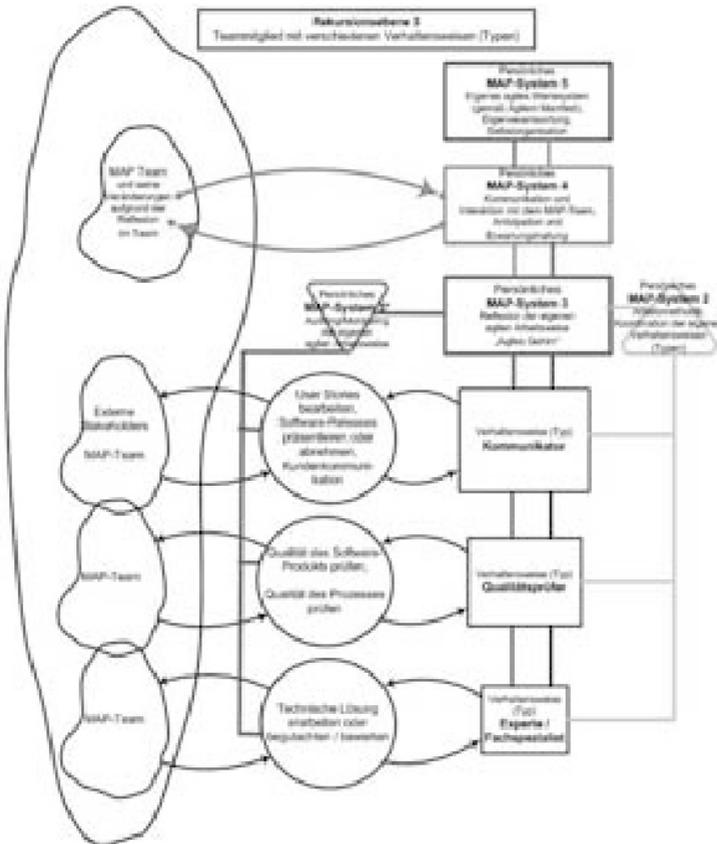


Abbildung 4: VSM-Rekursionsebene 3 – MAP-Teammitglied mit verschiedenen Verhaltensweisen (Typen)

In Abbildung 4 ist die Rekursionsebene 3 des Viable System Modells dargestellt, also das einzelne *MAP-Teammitglied* mit seinen verschiedenen Verhaltensweisen bzw. ausgeprägten MAP-Typen [HW12]. Exemplarisch ist hier ein Teammitglied dargestellt, das Potenzial zum *Product Owner* hat (siehe auch Kapitel 4). Es verfügt folglich über 3 persönliche MAP-Systeme 1, die MAP-Typen *Kommunikator*, *Qualitätsprüfer* und *Experte* mit abnehmender Wichtigkeit, was durch die abnehmende Größe der System 1-Boxen dargestellt ist. Jedes System 1 hat seine spezifischen Aufgaben, die in der Summe den Aufgaben eines *Product Owner* entsprechen. Das Teammitglied koordiniert seine Verhaltensweisen (System 2) und steuert sich durch Reflexion der eigenen agilen Arbeitsweise (System 3 und 3*). Das Teammitglied kommuniziert und interagiert mit den übrigen Teammitgliedern, wobei Erwartungshaltungen zu Konflikten führen können. Das Teammitglied hat ein starkes „agiles“ Wertesystem mit einem hohen Anteil an Eigenverantwortung und Selbstorganisation, wie dies bei Mitgliedern von agilen Teams generell zu beobachten ist.

Die problemlose Integration von MAP in das Viable System Modell auf 2 Rekursionsebenen (ohne Anpassung von MAP!) zeigt, dass MAP gut in das VSM „passt“. MAP-Teams bilden

also im Sinne des VSM lebensfähige Systeme, die über die „organisatorische Intelligenz“ [Sc00] verfügen, die notwendig ist, um das Überleben auf dem Markt zeitlich und anforderungstechnisch kritischer Projekte zu sichern.

4 MAP und andere agile Prozesse – Erweiterung von Scrum um sozietäre Vorgehensbausteine

Vergleicht man den MAP-Referenzprozess [Abb. 2] mit anderen agilen Prozessen, wie z.B. mit der agilen Projektmanagementmethode *Scrum* [Abb. 5] fällt auf, dass sich die Prozesse vom Ablauf her ähneln. Dies hängt sicherlich damit zusammen, dass die Prozesse nicht nur agil, sondern grundsätzlich iterativ-inkrementell sind. Aber auch die Team-Rollen sind vergleichbar [Ha10, Kap. 7.6.1]. Folglich sollte auch MAP als „Oberbau“ zum agilen Prozess Scrum „passen“. Scrum muss lediglich um „sozietäre Vorgehensbausteine“ gemäß MAP erweitert werden.

Scrum ist eine agile *Projektmanagement*methode. Fragen der Teamstruktur stehen nicht im Vordergrund, sondern der (iterative) Ablauf des Projekts. Scrum wurde in den frühen 1990er Jahren von Ken Schwaber und Jeff Sutherland entwickelt, die Grundgedanken sind in [Sc04] niedergelegt. Scrum kennt nur wenige Projektrollen:

- Product Owner
- Team
- ScrumMaster

Der *Product Owner* ist für das Produkt verantwortlich. Er legt den funktionalen Umfang des Produkts fest und übernimmt die Sichtweise des Endkunden. Über die Festlegung der in der nächsten Iteration zu realisierenden Funktionalität des Produkts kann er den Prozess steuern.

Das *Team* entwickelt das Produkt, also die Software. Seine Zusammensetzung ist entscheidend für das Gelingen des Projekts. Dabei geht es nicht nur um die fachliche Qualifikation, sondern auch um die Teampsychologie, also das „Zusammenpassen“ der Teammitglieder. Scrum geht nicht weiter auf die Rollenverteilung im Team ein. Es ist jedoch klar, dass alle Rollen oder besser gesagt *Fähigkeiten*, die für eine erfolgreiche Projektentwicklung notwendig sind, vorhanden sein müssen. Das Team organisiert sich selbst und ist klein (sieben ± zwei Mitglieder laut Schwaber [Sc04]).

Der *ScrumMaster* ist der Prozessverantwortliche. Er coacht das Team, wenn es um die Einhaltung des Prozesses geht. Insbesondere bei mit Scrum unerfahrenen Teams hilft er, den Prozess zu etablieren. Er hilft dem Team, den Prozess immer wieder zu überdenken und gegebenenfalls anzupassen.

Der Ablauf des Scrum-Prozesses (*Scrum Flow*) ist einfach gehalten und wird in Abbildung 5 dargestellt.

Am Anfang steht die *Vision* des Product Owners mit einer kurzen Beschreibung des Produkts. Daraus entwickelt der Product Owner das *Product Backlog*, einen Katalog der gewünschten Anforderungen an das Produkt. Dieser ist die Grundlage der Planungssitzung, die zusammen mit dem Team stattfindet und jede Iteration, den sog. *Sprint* einleitet. Nachdem sich Product Owner und Team geeinigt haben, wird der festgelegte Funktionsumfang des Produkts in einem Sprint (meist in einer festgelegten Zeitspanne von einer Woche) implementiert. In dieser Zeit darf das Team nicht gestört werden. Nach dem Sprint präsentiert das Team im *Sprint Review* dem Product Owner das Ergebnis, welches von ihm abgenommen werden muss. In der *Sprint Retrospektive* reflektiert das Team unter Moderation des ScrumMasters den Prozess und überlegt sich Verbesserungen.

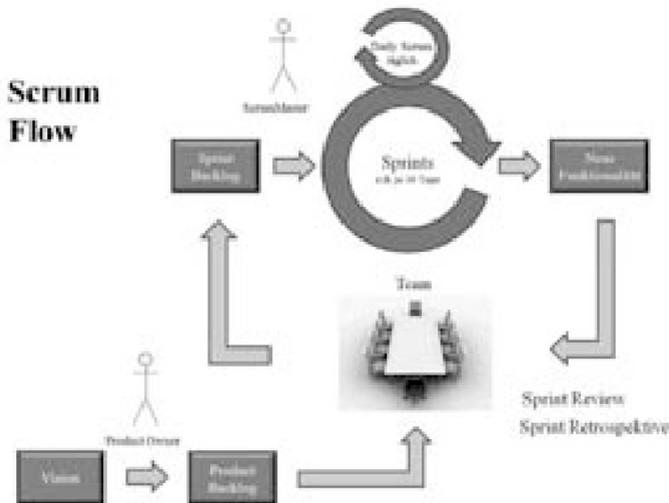


Abbildung 5: Scrum Flow

Zu erwähnen ist noch ein täglich wiederkehrendes Ritual des Teams: Im *Daily Scrum* koordiniert sich das Team und spricht auftauchende Probleme an. Der Daily Scrum dauert max. 15 Minuten und findet immer zur gleichen Tageszeit statt. Der ScrumMaster moderiert die Teamsitzung.

Da Scrum keine Aussage über die Zusammensetzung des Teams macht, kann an dieser Stelle die *MAP-Evaluierung* ansetzen. Der Scrum Flow wird um zwei entsprechende sozietäre MAP-Vorgehensbausteine erweitert [Abb. 6].

Im ersten Schritt durchlaufen die Teammitglieder die oben beschriebene MAP-Evaluierung. Die Teilnahme ist stets freiwillig! Danach werden die Projektrollen besetzt, wobei die MAP-Evaluationsergebnisse eine Grundlage bilden. Natürlich gehen hier auch fachliche Aspekte ein: So muss z.B. ein Product Owner das zu entwickelnde Produkt kennen. Eine Eignung nach sozietären Aspekten kann nur eines von mehreren Entscheidungskriterien sein.

Im nächsten Schritt findet ein *Prozess-Training* statt. Das Team muss insbesondere bei agilen Projekten gute Prozesskenntnisse haben und einen guten Überblick über agile Praktiken haben. Im Gegensatz zum reinen Scrum eröffnen die vorgelagerten sozietären Vorgehensbausteine nämlich noch eine weitere Möglichkeit: Das geschulte Team mit der gemäß MAP optimalen Besetzung der Projektrollen verfügt, wie die Praxis zeigt, auch über die Möglichkeit, bei Bedarf das gewählte Prozessmodell zu „verlassen“ und durch ein geeigneteres zu ersetzen. Deswegen spricht der Autor auch bei MAP von einem agilen *Meta-Modell*, weil das „unterliegende“ Prozessmodell geändert werden kann (mehr dazu in [Ha10, Kap. 7]). Natürlich verliert dadurch der ScrumMaster seine Aufgabe, die Einhaltung des Prozessmodells Scrum zu überwachen. Er wird zum Prozessmodell-Beobachter, zum *MAP-Beobachter*, der dem Team als Coach zur Verfügung steht und die Änderungen des Prozesses dokumentiert.

Um zu zeigen, wie weit die Prozessänderung gehen kann, sei auf das Team in der Session 2006 des studentischen Labors des Autors verwiesen. Hier führte die Prozessänderung im Laufe des Projekts zu einem „klassischen“ iterativ-inkrementellen Prozess mit einem starken Team-Leader. Dies ist zugegebenermaßen ein Extrembeispiel!

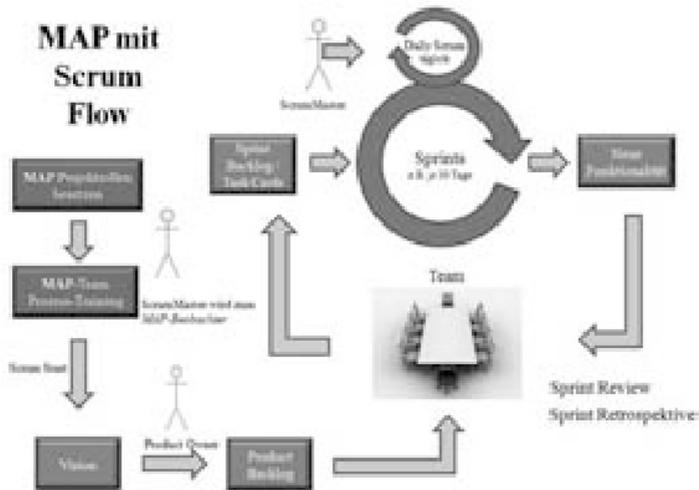


Abbildung 6: Scrum Flow um sozietäre Vorgehensbausteine erweitert

5 Sozietäre Vorgehensbausteine in der Praxis

Seit 2004 konnte an der DHBW Lörrach in insgesamt 13 studentischen Projekten (die neueren davon im Unternehmensumfeld) gezeigt werden, dass sozietäre Erweiterungen von Entwicklungsprojekten gemäß MAP die Qualität von Projekt *und* Produkt erhöhen. Die Projekte liefen meist über ein Semester, oft aber auch kompakt über 5 bis 8 lange Arbeitstage mit 2 bis 3 „Short Sprints“ von je zwei Tagen. Insbesondere in 4 studentischen Projekten, die 2011 zeitgleich im Wettbewerb untereinander stattfanden, setzte sich unter den konkurrierenden Projektteams das Teams durch mit der besten Teamrollenbesetzung gemäß MAP. Zwischenzeitlich konnten auch weitere 7 Teams einer schweizerischen Hochschule nach MAP analysiert werden. Auch diese Ergebnisse zeigen eine Korrelation zwischen dem Projekterfolg und einer guten Teamrollenbesetzung gemäß MAP. Schließlich werden diese Ergebnisse auch durch die Analyse zweier Unternehmensteams bestätigt, deren Projektlängen ein halbes Jahr und länger waren (2008 und 2011, siehe auch [Wo08]). Die Unternehmensprojekte zeigen keine Abweichungen zu den studentischen Projekten! Dies ist auch nicht zu erwarten, das sich die üblicherweise in agilen Unternehmensprojekten arbeitenden „Young Professionals“ von den Verhaltensweisen und der Ausbildung her meist nicht sehr von den Studierenden in den studentischen Projekten unterscheiden, insbesondere da diese während der Projekte schon in den hö-

heren Semestern sind und aufgrund ihres dualen Studiums bereits Praxiserfahrung in Unternehmensprojekten sammeln konnten.

Alle so gewonnenen Erfahrungen sind Indizien für das gute Funktionieren des neuen sozietären Ansatzes. Besonders wertvoll sind dabei die Ergebnisse der folgenden studentischen Projekte, die unter realen Bedingungen im Unternehmensumfeld abliefen: Im Rahmen des studentischen Labors an der DHBW Lörrach erstellten Studierende der Life Science Informatik [LSI08] 2010 und 2011 Software-Lösungen für das gemeinnützige Berufsförderungswerk (BFW) Bad Wildbad [BFW12] als Projektkunden. Das BFW Bad Wildbad ist ein Unternehmen der beruflichen Rehabilitation mit angeschlossener Klinik, insbesondere für querschnittsgelähmte Reha-Teilnehmer. Es gehört zur Josefs-Gesellschaft Köln, einem katholischen Träger von bundesweit 17 Einrichtungen für Menschen mit Behinderungen.

Neben einer Raumplanungs-Software für Reha-Gruppen mit behinderten Teilnehmern (2010), die besondere Ansprüche an Unterrichtsräume haben, wurde 2011 für das BFW eine webbasierte Reha-Teilnehmerverwaltungs-Software geschrieben. Beide Softwares sind zwischenzeitlich produktiv im Einsatz. In beiden Fällen konnten die sehr speziellen Bedürfnisse des BFW durch maßgeschneiderte Lösungen maximal berücksichtigt werden.

Der große Erfolg der BFW-Projekte beruht auf der konsequenten Umsetzung der von MAP geforderten „Projektzutaten“. Im Vorfeld unterzogen sich die Teammitglieder einer Selbst- und zwei Fremdevaluationen. (In den beiden Projekten waren das 11 und 7 Teammitglieder.) Daraus wurde die projektspezifische „Landkarte“ (MAP) der Verhaltensweisen im Team ermittelt und die Projektrollen besetzt.

Wie man in Abbildung 7 sieht, werden bei der Zuordnung der Verhaltensweisen oder *Typen* zu den Projektmitgliedern zwei getrennte Berechnungen durchgeführt. Deshalb sind jedem Teammitglied (dargestellt durch die Nummern 1 bis 7) zwei Balken zugeordnet. Grundsätzlich ist die Höhe des Balkens ein Maß für die Affinität zum jeweiligen Typ. Der kleinere Balken (blau) berücksichtigt dabei nur Ergebnisse aus der Evaluation, wenn sie mindestens in zwei von drei Evaluationen auftauchen. Alles andere wird nicht berücksichtigt. Die gute Zuordnung des Teammitglieds Nr. 2 zum Typ Kommunikator heißt also, dass die entsprechenden projektbezogenen und allgemeinen menschlichen Eigenschaften in zwei bis drei Evaluationen angekreuzt wurden. Einfache Nennungen werden nicht berücksichtigt. Aus der Überdeckung der „Kreuzungsflächen“ der projektbezogenen und der allgemeinen menschlichen Eigenschaften mit den Typen in der „Landkarte“ der Verhaltensweisen in Abbildung 1 berechnet sich die Affinität zum jeweiligen Typ. Anders verhält es sich mit dem höheren Balken (rot). Hier werden alle Nennungen der Eigenschaften berücksichtigt. So bildet sich also pro Teammitglied ein Balkenpaar, welches im Vergleich mit den anderen Teammitgliedern die Zuordnung zum entsprechenden Projekttyp verdeutlicht: Beispielsweise sind die Teammitglieder Nr. 2 und 5 gut als Kommunikatoren geeignet!

Abbildung 7 zeigt aber auch Teammitglieder, bei denen die Ausbildung eines Typs nicht sehr ausgeprägt ist. So ist für Teammitglied Nr. 3 bei der Zuordnung zu vier Typen der von der Ausprägung her „schärfere“ linke Balken (blau) gleich 0. Der zugehörige rote Balken (rechts) zeigt viermal eine Höhe zwischen 20% und 30%. Dieses Teammitglied ist also bezüglich der vier Verhaltensweisen Kommunikator, Ideengeber, Experte und Qualitätsprüfer recht ausgeglichen und „unauffällig“. Es zeigt sich aber eine Affinität zum Teammitglied (Teamworker) und – abgeschwächt – zum Troubleshooter.

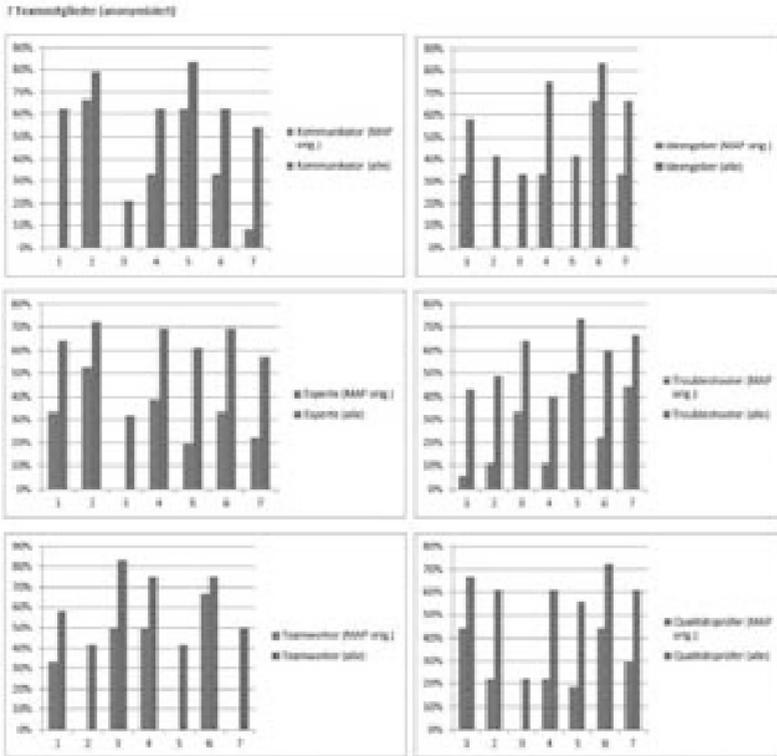


Abbildung 7: Verhaltensweisen im Team – Auswertung Projekt 2011

Interessant ist die Ableitung der Scrum-Projektrollen aus den beschriebenen Verhaltensweisen. Sie wird in Abbildung 8 dargestellt.

Dabei wird beispielsweise beim Product Owner angenommen, dass sich diese Scrum-Projektrolle aus den Typen *Kommunikator* (50%), *Qualitätsprüfer* (30%) und *Experte* (20%) zusammensetzt. Dazu kommt natürlich insbesondere eine gute Kenntnis des zu entwickelnden Produkts. Deshalb fiel die Wahl im Projekt 2011 auf das Teammitglied Nr. 6. Dies erwies sich als Glücksgriff, insbesondere da Teammitglied Nr. 6 über Krankenhauserfahrung verfügte und sich damit leicht in die Probleme einer Reha-Klinik versetzen konnte.

Natürlich ist die Kombination aus den o.g. MAP-Typen – insbesondere bei den %-Anteilen – vom Autor gewählt. Es ist deshalb wichtig, die Stabilität des Modells gegenüber Änderungen in den %-Anteilen zu testen. Dies wurde kürzlich in einer wissenschaftlichen Arbeit durchgeführt [Dy12]. Die Arbeit zeigt, dass Änderungen an den %-Anteilen keine signifikanten Änderungen bei der Zuordnung eines Teammitglieds zu einer Projektrolle ergeben. Das Verhältnis der Balken *zwischen* den Teammitgliedern wird wenig verändert. Die Entscheidung für ein Teammitglied ist also stabil.

Zwischenzeitlich hat der Autor eine Software geschrieben, die die Zuordnung von Teammitgliedern zu Projekttrollen automatisiert. Auch die Unterstützung des Meta Agile Process Models (MAP) durch Anpassung eines kommerziellen Projektmanagement-Tools ist zwischenzeitlich im Rahmen einer Studienarbeit implementiert worden [Bo12].

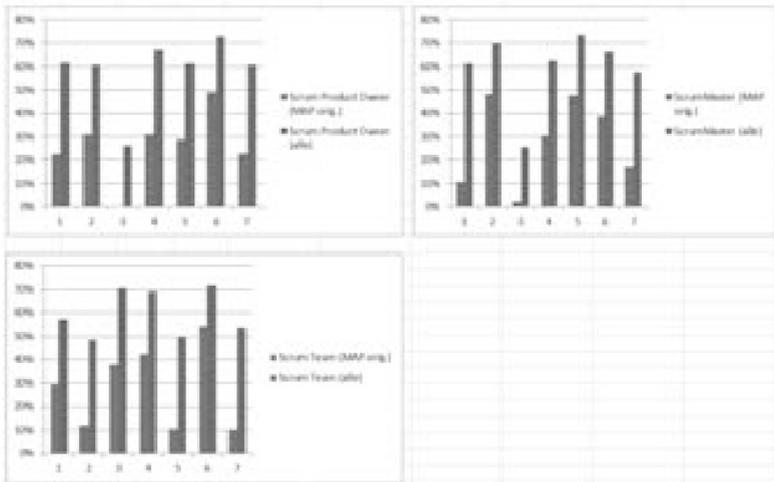


Abbildung 8: Scrum-Projekttrollen – Auswertung Projekt 2011

Insgesamt lässt sich sagen, dass die Zuordnung der Scrum-Projekttrollen mittels MAP-Evaluation sehr erfolgreich ist: Nach MAP-Evaluation und intensiver Schulung in agilen Techniken (in diesem Fall in Scrum) konnte das in den Abbildungen 7 und 8 betrachtete Team den ersten Sprint in Angriff nehmen. In *nur* drei kurzen Sprints konnte das Produkt in einer Version 1.0 entwickelt werden. Hier zeigte sich ein großer Unterschied zwischen reinem Scrum und der Erweiterung durch den sozietaären Ansatz MAP: Während das reine Scrum als agile Projektmanagementmethode rigide ist bezüglich der Abfolge der Projekt-„Rituale“, ist das durch MAP erweiterte Scrum lediglich als agiler „Startprozess“ zu verstehen, der im Rahmen der MAP-Regeln verändert und angepasst werden darf und soll. Die (zugegebenermaßen provokante) Frage sei erlaubt, inwiefern es dann überhaupt noch Entwicklungsprozesse braucht: Reicht für den Projekterfolg nicht ein in Prozesspraktiken gut geschultes Team, dessen Schlüsselpositionen durch eine MAP-Evaluierung besetzt werden? Die bisherigen Ergebnisse des Autors lassen diesen Schluss zu.

Literatur

- [Ba04] Baldegger R (2004) Erfolgreich im Team. Baldegger-Verlag
- [Be79] Beer S (1979) The Heart of Enterprise. Chichester: Wiley
- [Be81] Beer S (1981) Brain of the Firm. 2nd ed. Chichester: Wiley
- [Bel04] Belbin R M (1981, 2004) Management Teams. Elsevier

- [Bel07] Belbin R M (1993, 2007) Team Roles at Work. Elsevier
- [BFW12] Berufsförderungswerk Bad Wildbad (Reha-Klinik).
<http://www.bfw-badwildbad.de>. Letzter Zugriff März 2012
- [BG01] Baldegger R, Gotsmann L (2001) Ganzheitliches Projektmanagement. Baldegger Verlag
- [Bo12] Börmert E (2012) Projektmanagement-Werkzeuge: MAP und in-Step. Studienarbeit DHBW Lörrach
- [Dy12] Dyck M (2012) MAP: Auswertung des „Meta Agile Process Models“. Studienarbeit DHBW Lörrach
- [Ha08a] Hanser E (2008) <http://www.gpm-infocenter.de/PMMethoden/MAP>. Letzter Zugriff März 2012
- [Ha08b] Hanser E (2008) Agile Software-Entwicklung: Das „Projektmanagement-Labor“ der BA Lörrach, Beitrag in „Innovationen durch Projektmanagement – oder ?!“, interPM 2008. dpunkt.verlag
- [Ha10] Hanser E (2010) Agile Prozesse: Von XP über Scrum bis MAP. Heidelberg: Springer-Verlag
- [HW12] Hanser E, Wehinger J (2012) Integration MAP in VSM. Private Notizen
Abb. 3 und 4 wurden gemeinsam mit J. Wehinger (Volkswagen Konzernforschung, Zukunftsforschung und Trendtransfer) entwickelt. Weitere gemeinsame Veröffentlichungen sind geplant.
- [LSI08] Life Science Informatik, Studienrichtung an der DHBW Lörrach, seit 2008.
<http://www.dhbw-loerrach.de/index.php?id=life-science-informatik>
Letzter Zugriff Juli 2012
- [Sc00] Schwaninger M (2000) Das Modell Lebensfähiger Systeme - Ein Strukturmodell für organisationale Intelligenz, Lebensfähigkeit und Entwicklung. Universität St. Gallen.
<http://www.tranquilla.ch/serveattachment/984e4202df9fe30e5459fa815a35c6dc/D35.pdf>
Letzter Zugriff Juli 2012
- [Sc04] Schwaber K (2004) Agile Project Management with Scrum. Microsoft Press
- [Wa08] z.B. *Diskussion mit Dr. Klaus Wagenhals, Frankfurt, an der interPM 2008*
- [Wal12] The VSM Guide.
http://www.esrad.org.uk/resources/vsmg_3/screen.php?page=preface
Letzter Zugriff Juli 2012
- [WH12] Wehinger J, Herrmann C (2012) Das Viable System Model als Bezugsrahmen für Agilität am Beispiel Scrum. Beitrag in „IT-Projektmanagement 2012+ im Spagat zwischen Industrialisierung und Agilität?“, interPM 2012. Dpunkt.verlag
- [Wo08] Wolf O (2008) Entwicklung eines Schulungsverwaltungssystems unter Anwendung der Methoden des Extreme Programmings. Diplomarbeit, Berufs-akademie Lörrach (jetzige DHBW Lörrach)