

Hybrid ist Pflicht – mit Ultimate/Reliable Scrum und Critical Chain zu einer hochskalierbaren agilen Projektorganisation

Wolfram Müller

Abstract: Agile Methoden sind produkt- und teamorientierte Ansätze und stehen im Widerspruch zu Projekten mit hartem Terminen und Abhängigkeiten. Um die Vorteile aus beiden Welten zu nutzen muss man auch an beiden Welten Veränderungen vornehmen.

Das klassische Projektmanagement leidet an zu viel Work-in-Progress und starrer Terminplanung auf Arbeitspaketebene. Hier bietet Critical Chain Projektmanagement (CCPM) ein bewährtes Methodenset um Fluss und Agilität zu ermöglichen. Auf der anderen Seite können Agile Methoden keine Lieferumfänge zusichern und sind zu langsam. Auch hier existieren mit Ultimate/Reliable Scrum Agile Methoden der 3. Generation, die wiederum Fluss und Zuverlässigkeit ermöglichen. Damit ist der Weg frei – je nach Projektsituation, Projektphase, Teilprojekt oder Arbeitspaket genau die Vorgehensweise zu wählen, die mit geringstem Aufwand die Erfordernisse erfüllt.

In der Praxis wird das Portfoliomanagement, mit dem Konzept der „Virtual Drum“ aus der „Theory of Constraints“, massiv vereinfacht. Hier werden an einen „künstlichen/virtuellen“ Engpass die Projekte so gestaffelt, dass keine Ressource auf Dauer überlastet ist. Im Projektmanagement werden die Terminpläne durch Puffermanagement ersetzt. Zur Steuerung wird nur noch Fortschritt auf der kritischen Kette zu Pufferverbrauch herangezogen. Innerhalb von Arbeitspaketen oder Teilprojekten können nun auch agile Methoden zum Einsatz kommen. Die Teams erreichen mit Hilfe von Ultimate Scrum Boards den Zustand des „one piece flow“ und optimalen Durchsatz. Wie im Projekt wird die Zuverlässigkeit auch hier mit einem Puffer und Puffermanagement sichergestellt – namentlich Reliable Scrum. Es entsteht ein vollskalierbares agiles Projektmanagement Rahmenwerk.

Dabei handelt es sich nicht um Theorie sondern um bewährte Praxis mit enormen Potentialen.

Keywords: Ultimate Scrum, Reliable Scrum, Critical Chain Projektmanagement, CCPM, Theory of Constraints, TOC, Tame the Flow, Skalierbarkeit, agiles Projektmanagement

1 Einleitung

Wenn man das Ziel verfolgt eine hochskalierbare agile Projektumgebung zu entwickeln, dann gilt es auf der einen Seite die Projektwelt zu verbessern und auf der anderen Seite die agilen Methoden. Erst wenn beides auf der nächsten Stufe angelangt ist lässt sich beides integrieren und skalieren.

Im folgenden Beitrag wird daher zuerst aufgezeigt worin sich Projekte und agile Methoden unterscheiden. Erst wenn man diesen Unterschied kennt wird es möglich zu beurteilen wann und wo welche der Ansätze am sinnvollsten genutzt werden können.

Als nächstes wird gezeigt, wie man das Projektmanagement so entwickeln kann, dass es näher an die agile Welt reicht. Hierzu wird das Portfoliomanagement deutlich verschlankt so dass es ausgehend von einem virtuellen Engpass („Virtual Drum“) immer sichergestellt ist, dass der „Work-In-Progress (WIP)“ oder Bestand an Arbeit so eingestellt ist, dass kein Team überlastet ist. Hinzu kommt eine Projektampel, die basierend auf Fortschritt und Pufferverbrauch ein eindeutiges operatives Prioritätssignal generiert. Das Ergebnis ist eine in sich leichtgewichtige und agile Projektorganisation – namentlich Critical Chain Projektmanagement.

Nun kommen die agilen Methoden an die Reihe – auch hier gibt es Nachholbedarf. Als erstes müssen die agilen Methoden befähigt werden zuverlässig bestimmte Lieferergebnisse zu erzielen – dies geschieht durch ein Puffermanagement sehr ähnlich dem der Projekte. Dies wird als „Reliable Scrum“ bezeichnet. Darüber hinaus sind die aktuellen agilen Methoden in der Praxis zu langsam. Hierzu wird der Arbeitsfluss, wie in Lean Management, auf ein „One-Piece-Flow“ umgestellt und durch Konzepte aus der TOC ergänzt. Das Ergebnis ist optimaler Arbeitsfluss, minimaler Bestand und höchster Durchsatz. Die wird als „Ultimate Scrum“ bezeichnet.

Im letzten Teil wird das nun verbesserte Projektmanagement mit den optimierten agilen Methoden über die Arbeitspakete miteinander verknüpft. Es entsteht eine voll skalierbare agile Projektmanagement Organisation.

Dieser Beitrag kann nur an der Oberfläche kratzen und versuchen einen Überblick zu geben. Vertiefende Literatur und Quellen werden daher ausführlich genannt.

2 Unterschied zwischen Projekten und Produkten (Agile)

Diese Abgrenzung hat weitreichende Auswirkung für das Projektmanagement und hilft vor allem die Unterschiede zwischen den agilen und den projektorientierten Ansätzen zu verstehen.

2.1 agile Methoden – Produktion?

Von Produktion spricht man bei einem Anteil der Bearbeitungszeit („Touch Time“) zur Durchlaufzeit („Lead Time“) von kleiner als 10% [EG08]. Wenn man sich z.B. die Produktion von einer Schraube, einem Auto oder einem elektronischen Gerät betrachtet sieht das immer wie folgt aus: Das Produkt wird in vielen kleinen Teilen produziert. Viele dieser kleinen Teile bilden zusammen ein Produktionslos („Batch“). Aus Sicht des einzelnen Teils besteht der größte Teil der Durchlaufzeit aus "warten". Warten in einer

Palette, dann transportierten, entnehmen, kurz bearbeiten, wieder in eine Palette, transportieren u.s.w. - zwischen den einzelnen Bearbeitungsschritten entstehen relativ lange Wartezeiten.

Das was sich jetzt vielleicht negativ anhört hat aber große Vorteile. Durch die verhältnismäßig kurzen Bearbeitungszeiten kann man die einzelnen Zeiten recht genau schätzen. Wenn eine Bearbeitungszeit einmal überschritten wird, kann sie sich mit der Unterschreitung des nächsten ausgleichen und fällt nicht ins Gewicht. Wenn mehrere Teile parallel zu bearbeiten sind erhält man durch die langen Wartezeiten die Möglichkeit die Reihenfolge der Teile vor jedem Arbeitsschritt zu vertauschen und kann so jeden beliebigen Termin einhalten. Der Bestand („Work-In-Progress“ WIP) lässt sich leicht begrenzen, da er sehr überschaubar ist. Produktionssteuerungen sind extrem einfach, stabil und benötigen wenig Planungs-overhead.

Hierfür muss man aber einen Preis bezahlen: man muss alle Vorhaben in kleinste, relativ unabhängige Teilaufgaben zerlegen und man braucht irgendwo Warteschlangen (Puffer oder „Backlogs“) die die mittlere Durchlaufzeit erhöhen.

Wenn man sich nun die agilen Methoden betrachtet wird man feststellen, dass diese im Kern Produktionssteuerungen sind - kleine Teilaufgaben, unabhängige Teilaufgaben, Puffer auf den Taskboards und Backlog.

2.2 Projektmanagement

Von Projekten oder Projektmanagement spricht man wenn, der Anteil der Bearbeitungszeit zur Durchlaufzeit größer als 20% beträgt. Hierbei wird vor allem der kritische Pfad oder, wenn man die Ressourcenverfügbarkeit mit betrachtet, die kritische Kette (Critical Chain) [EG97] betrachtet. Das Ziel im Projektmanagement ist typischerweise das vereinbarte Ergebnis in der kürzest möglichen Zeit zu erbringen. Hierzu sind die echten Abhängigkeiten zwischen den Arbeitspaketen zu betrachten. Die Arbeitspakete sind hierbei optimal mit Ressourcen zu versorgen, so dass die Projektlauftzeit minimal wird.

Im Gegensatz zur Produktion strebt man nun aber einen Anteil der Bearbeitungszeit an der Durchlaufzeit von 100% an (typisch sind effektiv bis 80%). Hieraus ergibt sich, dass man folgende Eigenschaften von Projekten explizit managen muss: die Abhängigkeiten, die Streuungen in den Schätzungen und die operative Ressourcenzuordnung im Multiprojektmanagement. Projektmanagement ist deutlich komplexer im Vergleich zur Produktion und benötigt mehr Management-overhead.

Bei Projekten muss man daher: ausreichend konzeptionieren und planen, so dass die Arbeitspakete und ihre Abhängigkeiten klar werden, muss die Aufwände schätzen um die Ressourcenzuordnung sicher zu stellen und die Streuungen am Projektende oder im Projektportfolio managen.

2.3 Projekte und Produktion (agile Methoden) im Vergleich

Die folgende Tabelle gibt eine Übersicht über die unterschiedlichen Systemklassen – Projekt und Produktion.

Kriterium	Projektwelt	Agile Welt (Produktion)
Touch/Lead-Time	nahe 100%	nahe oder kleiner 10%
Kopplung der Aufgaben	Starke Abhängigkeiten	Relativ unabhängig
Größe der Aufgaben	Stark unterschiedlich – großer Unterschied zwischen kleinstem und größtem	Relativ ähnlich und gleichmäßig klein,
Streuung/Risiko der Schätzungen	hohe Unsicherheiten	kleine Unsicherheiten
Menge der Aufgaben	Nur wenige kritische	Große Menge
Ziel	Lieferung eines definierten Umfangs zu einer definierten Zeit Zeitlich begrenztes Vorhaben	Kontinuierliche Lieferung von optimalen Kundennutzen Evolutionäres Entwickeln eines Produktes
Steuerungen	Klassisches Projektmanagement, Critical Chain Portfoliomanagement und Einzelprojektmanagement	div. Methodensets: u.a. eXtreme Programming, Scrum, Kanban, Reliable/Ultimate Scrum, Safe, Less, u.v.a.
Steuerungsaufwand	Hoch	Gering

Bei genauerer Betrachtung wird offensichtlich, dass in der Praxis beide „Welten“ immer gemischt auftreten.

In jedem Projekt sind per Definition die Charakteristika der Projektwelt vorhanden. Je nach Teilprojekte, Phasen oder Arbeitspaket können aber die Charakteristika der Produktionswelt (agile Methoden) überwiegen und somit von der einfacheren Steuerung profitieren können.

Umgekehrt weisen große agile „Vorhaben“ Charakteristika von Projekten auf. Vor allem in der Schlussphase werden Abhängigkeiten zu angrenzenden Fachgebieten stärker. In

großen Vorhaben sind Abhängigkeiten zwischen mehreren Teams zu managen und Ressourcenkonflikte mit dem Engpass aufzulösen. Damit steigt der Bedarf nach Methoden aus der Projektwelt.

3 Projektmanagement agilisieren – Critical Chain

Eine Projektmanagementorganisation besteht immer aus zwei Ebenen – dem Portfoliomanagement in dem die strategischen Entscheidungen (Prioritäten) gefällt werden und dem Projektmanagement darunter in dem die operative Ausführung im Vordergrund steht.

Um das bisherige Projektmanagement agiler zu gestalten werden folgende zwei Veränderungen vorgenommen.

Vorrangig wird Portfoliomanagement, mit dem Konzept der „Virtual Drum“ [UT14, HA10] aus der „Theory of Constraints“, massiv vereinfacht. Hier werden an einen „künstlichen/virtuellen“ Engpass die Projekte so gestaffelt, dass keine Ressource auf Dauer überlastet ist. Diese Situation ist bzgl. des Arbeitsflusses optimal.

Im Folgenden werden im Projektmanagement die Steuerung über Terminpläne durch Puffermanagement ersetzt [EG97, UT14]. Der Puffer wird hierzu den Arbeitspaketen entnommen und am Projektende zusammengefasst. Zur Steuerung wird nur noch Fortschritt auf der kritischen Kette zu Pufferverbrauch herangezogen.

Ausgangspunkte der von Dr. Eliyahu Goldratt entwickelten Methodik sind folgende Annahmen:

- Multitasking führt zu Mehraufwand, kostet Zeit und ist deshalb zu verhindern.
- In jeder Entwicklung gibt es einen Engpass und nur dort kann wirklich optimiert werden. Je reibungsloser es hier läuft, desto schneller der gesamte Prozess.
- Da Mitarbeiter zuverlässig sein wollen, bauen sie bei ihren Aufwandschätzungen oft bewusst oder unbewusst Puffer ein. Doch solche Reserven in einzelnen Aufgaben nützen dem Gesamtprojekt nichts. Werden sie nicht gebraucht, wird der Verantwortliche das nicht kundtun, um beim nächsten Projekt nicht unter Druck zu geraten. Dauert eine Aufgabe hingegen länger, wird der Verzug weitergegeben.

Ziel von Critical Chain ist es, schädliches Multitasking zu verhindern und die Reserven nutzbar zu machen. Die Aufgaben werden so organisiert, dass das Team/die Fähigkeit, deren Verfügbarkeit den größten Engpass bildet, konzentriert eine Aufgabe nach dem anderen abarbeiten kann.

Die Zeitschätzungen der Mitarbeiter werden um die geschätzten Puffer reduziert. Die gewonnene Zeit wird teilweise als Projektpuffer aggregiert und darf nach und nach verbraucht werden. In der Praxis entsteht ein regelrechter Wettbewerb darin, Puffer zu identifizieren und die Projektlaufzeit zu verkürzen.

Ist die Zahl der Projekte so hoch, dass schädliches Multitasking entstehen kann, wird der Bestand „Work in Progress (WIP)“ so weit reduziert, dass der Engpass gerade noch ausgelastet ist. Es entsteht eine Situation, in der alle Projekte genügend Ressourcen haben – was in klassischen Projektumgebungen meist nur für das Projekt mit der höchsten Priorität gilt. Erst wenn die kritischste Aufgabe fertig ist, wechselt der Mitarbeiter/das Team zur nächsten. Das bedeutet in der Regel, dass die Zahl der gleichzeitig aktiven Projekte sinkt, aber die Projektlaufzeiten deutlich kürzer werden.

Die Ressourcen werden nun operativ nach Fortschritt zu Pufferverbrauch der Projekte gesteuert. Das Projekt mit dem schlechtesten Verhältnis erhält die kritischen Ressourcen und die höchste Aufmerksamkeit.

Bei Projekten erweist sich oft die Integrationsphase (Integration, End-2-End-Tests, Bugfixing) als kritischer Engpass: D.h., das Unternehmen muss klar definieren, wie viele Integrationsprozesse es gleichzeitig verkraften kann, ohne dass die Projekte sich durch schädliches Multitasking gegenseitig behindern. An diesem Engpass werden die Projekte entsprechend getaktet. Die Integration hat höchste Aufmerksamkeit und Unterstützung des Managements inklusive.

Ergebnis: Die Projektlaufzeiten sinken, die Zahl der Projekte, die pro Monat oder Jahr fertig gestellt werden, steigt ebenso wie die Zuverlässigkeit – und das bei gleichen Ressourcen wie zuvor. Es kommt mehr Ruhe in die Projekte und der Stress für alle Beteiligten sinkt.

Durch die Umstellung der Steuerung auf Critical Chain reduziert sich die Durchlaufzeit initial um ca. 25% bis hin zu langfristig 50-70%. Im Gegenzug steigt der Durchsatz um entsprechende ca. 30% und darüber hinaus. Dies alles geschieht bei gleichem Ressourceneinsatz und höchster Zuverlässigkeit von ca. 90%.

Eine Übersicht über diese Erfolge findet sich auf folgenden Webseiten:

- www.goldrattconsulting.com/Industries
- speed4projects.net/critical-chain/Erfolgsgeschichten

Die Art der Steuerung über den Projektpuffer und die enorme Steigerung der Geschwindigkeit sind ein deutlicher Schritt in Richtung Agilisierung.

4 Agile Methoden zuverlässig machen – Reliable/Ultimate Scrum

Agile Methoden haben in den letzten Jahren ihre Leistungsfähigkeit verdeutlicht. Die starke Fokussierung auf den Kunden und die Nutzung der Teamdynamik sind zwei Faktoren. Die einfache Steuerung und wenigstens temporäre Begrenzung des Work-In-Progress der zumeist größere Erfolgsfaktor.

Folgende Themen sind in der agilen Welt aber bisher nicht adressiert:

1. Zuverlässigkeit der Lieferung – keine agile Methode kann einen bestimmten Umfang zu einer bestimmten Zeit sicherstellen.
2. Alle agilen Methoden weisen hohe Bestände an Aufgaben entweder im Backlog oder als Work-in-Progress auf – daher sind alle agilen Methoden zwar scheinbar schneller – die Potentiale werden aber noch nicht ausgeschöpft.
3. Die Skalierbarkeit für große Vorhaben ist bisher nur unzureichend und unter großem Aufwand möglich.

Daher gilt es die guten Eigenschaften der agilen Methoden beizubehalten und die oben genannten Schwachpunkte zu beseitigen.

4.1 Zuverlässigkeit herstellen – „Reliable Scrum“ [ST13, WM12]

Ausgangspunkte sind das bekannte Scrum oder Kanban – diese werden ergänzt um zwei Elemente aus Critical Chain:

- Begrenzung des Work-in-Progress durch sachrichtiges ausbalancieren von Umfang, Ressourcen und Termin.
- Darstellung des Projektstatus als Fieberkurve mit Pufferverbrauch und Fortschritt (identisch zu Critical Chain).

Ziel von Reliable Scrum ist es dem Team eine realistische Erfolgswahrscheinlichkeit zu geben und dem Product Owner (Auftraggeber und Projektmanager) ein Werkzeug zur Steuerung des Backlogs (Work-In-Progress) zu geben, so dass zum zugesicherten Termin und die zugesicherten Funktion sicher geliefert werden.

Das Backlog wird vervollständigt und geschätzt. Die Abarbeitungsgeschwindigkeit wird ermittelt. Zusammen ergibt sich die Erfolgswahrscheinlichkeit. Das Backlog wird nun so eingestellt und verhandelt, dass die Erfolgswahrscheinlichkeit ausreichend hoch wird.

Durch diese realistische Erfolgswahrscheinlichkeit entsteht ein Puffer am Projektende. Mit diesem Puffer kann die aus Critical Chain bekannt Fieberkurve gezeichnet werden. Hier können die Stakeholder objektiv verfolgen ob das Release/Vorhaben auf Kurs ist und der „Product Owner“ (Auftraggeber/Projektmanager) kann sein Backlog immer so managen, dass die Erfolgswahrscheinlichkeit erhalten bleibt.

Wenn mehrere agile Projekte zusammen arbeiten müssen, ist oft schwer den Überblick über die einzelnen Teilstränge zu halten und die Abhängigkeiten zu managen. Die Fieberkurve kann man nun auch für ein Portfolio erstellen und so sicherstellen, dass der Großteil der Projekte „grün“ ist. In dem Moment werden Abhängigkeiten sicher eingehalten und Probleme früh entdeckt.

Ergebnis: der Auftrag wird deutlich schneller und besser geklärt. Das Team erhält dadurch einen klaren Leuchtturm, Fokus und Motivation. Der „Product Owner“ kann sein Backlog managen und den Status objektiv in Richtung Stakeholder kommunizieren. In Folge werden die „Releases“ (Projekte) deutlich schneller und zuverlässiger.

4.2 Ultimate Scrum [ST13, WM13]

Die Teams erreichen mit Hilfe von Ultimate Scrum Boards den Zustand des „one piece flow“ und optimalen Durchsatz, was einer Drum-Buffer-Rope (DBR) Produktionssteuerung, aus der Theory of Constraints [UT10], entspricht.

Um eine DBR als Projektsteuerung zu nutzen muss man, wie bei Scrum oder Kanban, das Projekt in kleinste Einheiten (Stories/Tasks) zerlegen. Zusätzlich muss man einen Arbeitsschritt/Ressource als Engpass („Drum“/Trommel) definieren und dann den Start von neuen Aufgaben genau an diesem Arbeitsschritt ausrichten. Hierzu wird vor/in diesem Arbeitsschritt ein Arbeitsvorrat („Buffer“/Puffer) installiert. Wenn dieser unter einen definierten Bestand sinkt werden neue Stories/Tasks gestartet – der Bestand dient als Signal („Rope“/Signalseil). Dies wird immer in Form eines Taskboards visualisiert.

Ein Ziel ist es die Sprints zu entfernen um damit die unnatürlichen Brüche am Ende der Sprints zu vermeiden und in einen kontinuierlichen Fluss zu kommen. Der Fluss ist wichtig um die Anzahl der offenen Stories und Tasks zu verringern und somit die Durchlaufzeit zu Verkürzen. Am Schluss natürlich auch der Durchsatz zu steigern.

Viele Dinge aus Scrum bleiben bestehen - nur die Steuerung wird angepasst. Eine Retrospektive alle 2-3 Wochen – ist immer noch sinnvoll. Es gibt weiterhin ein „Planning 1“ – durch „Reliable Scrum“ wird das Backlog aber in den ersten Sprints weitgehend komplett qualifiziert, so das Backlog vollständig priorisiert und geschätzt vorliegt und nur noch angepasst werden muss. Reviews werden natürlich auch gemacht - aber nicht in einem definierten Rhythmus, sondern alle 5-10 Stories, wenn das Lieferergebnis wirklich Wert aufweist. Ebenso die „Daylies“ bleiben erhalten.

Was ändert sich dann? Es gibt keine Sprints mehr! Es ist alles ein kontinuierlicher Fluss mit dem Ziel, so wenig wie möglich Stories und Tasks geöffnet haben.

Um das zu erreichen, wird der Prozess in zwei unabhängige Teile geteilt 1. das Schneiden der Stories in Tasks (auf der linken Seite des Taskpuffer) und 2. das Abarbeiten der Tasks (auf der rechten Seite).

Die Steuerung der linken Seite ist extrem einfach. Das Aufteilen von Stories in Task ist ja für eine Person nur wenige Stunden Aufwand – ungefähr so viel wie ein Task selbst. Daher wird das Schneiden von Stories ausschließlich durch die Menge der Tasks im Taskpuffer gesteuert. Wenn nur noch zwei Tasks übrig sind – wird einer der Entwickler die nächste Story aus dem Backlog nehmen und sie in Tasks aufbrechen. Es kann sein, dass die Alarmgrenze von 2 Task zu riskant ist. Sie sehen, dass - wenn ein Pufferloch auftritt – also keine Aufgabe im Taskpuffer mehr übrig ist. Wenn Pufferlöcher auftreten, muss man einfach die Alarmschwelle solange (langsam und schrittweise) erhöhen, bis keine Pufferlöcher mehr auftreten. Die Alarmschwelle sollte dabei nicht die Hälfte der Anzahl der beteiligten Entwickler überschreiten, ansonsten ist dies ein deutlicher Hinweis auf Prozessprobleme.

Das Monitoring geschieht über die, aus dem „Reliable Scrum“ bekannten Werkzeuge, also das klassische „Product Burndown Chart“ und die neue „Fieberkurve“. Diese Diagramme werden immer aktualisiert, sobald eine Story fertig gestellt wurde. Hierdurch entstehen viel mehr Messpunkte und noch feinere „Echtzeit“ Transparenz.

Und jetzt auf der rechten Seite? Normalerweise hat eine Drum-Buffer-Rope viele Prozessschritte (wie in Kanban). Hier haben wir aber einen kontinuierlichen Prozess mit zwei Schritten "Entwicklung" und "Review/Test". Das Besondere ist aber, dass es nur eine Art von Ressourcen gibt – Entwickler. Diese haben zwar Unterstützung durch Qualitätsmitarbeiter, die die Tests zu schreiben/durchführen - aber am Ende sind die Entwickler der begrenzende Faktor. Daher macht es keinen Sinn, zwischen den Prozessstufen zu unterscheiden. Beide werden durch die Verfügbarkeit der Entwickler eingeschränkt.

Die Drum-Buffer-Rope Steuerung besteht nun aus folgenden drei Teilen:

- Die Trommel - dies ist die begrenzende Ressourcen - in diesem Fall die Entwickler. Die Trommel ist wie der Herzschlag der Produktionskette. Sie gibt den Takt vor – nach ihr müssen sich alle richten.
- Dann benötigen wir einen Puffer (in der Regel vor der Trommel), aber in diesem Fall, wenn es nur einen begrenzenden Prozessschritt gibt ist der komplette Bestand (Anzahl der offenen Tasks) selbst der Puffer - spielt aber für die Steuerung keine Rolle ob ein angefangener Task im Bearbeitung ist oder in einem Puffer liegt. Keine dedizierten Puffer ist Letzt endlich sogar ideal.
- Und das Seil? Das ist die Signalleitung vom Puffer um neue Aufgaben zu starten. In diesem Fall ganz einfach - wenn eine Aufgabe abgeschlossen ist, dann darf eine neue Aufgabe gestartet werden.

Das Monitoring für die rechte Seite besteht aus einem "aggregierten Input-Output-Diagramm", oder manchmal auch "Flussdiagramm" oder „Continuous Flow Diagram (CFD)“ genannt. Das Ziel ist es, den Bestand (Differenz zwischen Ein- und Ausgangslinie) so niedrig wie möglich zu halten.

Dies kann auf sehr einfache Weise erreicht werden. Es werden keine neuen Aufgaben begonnen, bis die ersten "Pufferlöcher" entstehen. Wenn der erste Entwickler keine Task mehr hat, dann kann ein zusätzlicher Task gestartet werden und damit wird der Puffer um eins erhöht. Dies sollte aber eine Ausnahme sein und es müssen die Ursachen hierfür untersucht werden. Pufferlöcher sind voll von Informationen über Hindernisse oder verfahrenstechnische Probleme. Aber schließlich, wenn alles getan wurde und immer noch Pufferlöcher auftreten, dann muss man den Bestand erhöhen um den Durchsatz zu sichern.

Das Ergebnis ist ein kontinuierlicher Fluss ohne Prozessstörungen. Durch den minimalen WIP ist die Durchlaufzeit ebenfalls minimal und der Durchsatz erreicht das Optimum.

5 Das voll skalierbare agile Projektmanagement Rahmenwerk

Mit Critical Chain Projektmanagement (CCPM) und den verbesserten agilen Methoden – Reliable und Ultimate Scrum sind die Einzelteile vorhanden. Diese müssen nur noch zu einem Ganzen integriert werden [MH14].

Das hier vorgestellte System geht daher einen großen Schritt weiter in Richtung eines Framework für vollintegriert Projekt- und Produktentwicklung. Beide Steuerungen (Projekt und agile Methoden) sind in der Praxis tatsächlich einfach zu kombinieren und zwar an der Stelle der operativen Priorität von Arbeitspaketen/Stories. Alle hier vorgestellten Elemente sind schon in unterschiedlichsten Kontexten erfolgreich im Einsatz - das Framework ist die Zusammenführung.

Das Framework soll kein Rückschritt in die tayloristische prozessuale Welt sein - sondern der Schlüssel, mit sehr wenig zentralen Informationen (strategische und operative Priorität), der zugrundeliegenden Organisation möglichst viel Freiheit für Selbstorganisation zu geben. Hierdurch können sich die positiven Wirkungen der teamorientierten agilen Welt entfalten und die Vorteile einer großen Organisation und Projekten genutzt werden.

Beschreibung:

- Es handelt sich um ein 3-Schichten-Framework. Jede Schicht hat eigene Charakteristika und Steuerungen.
- Das Portfolio-/Demandmanagement ist die oberste Schicht. Der Eingang wird über aber über Throughput Accounting Ansätze [JC04] gesteuert. Zum terminieren kommt entweder eine Stafflung an der „Virtaul Drum“ (für Projekt) oder eine simplified-Drum-Buffer-Rope (für die Kleinaufträge) zum Zuge. Damit werden Trade-Offs schnell transparent und das Board kann die strategischen Prioritäten setzen.

- Auf der untersten Eben finden sich die Teams, die „Epics“ (Teilprojekte) und Arbeitspakete („Stories“) umsetzen. Hier kommt Reliable/Ultimate Scrum zum Zuge. Reliable Scrum um „Epics“ sicher zu einem Termin liefern zu können – damit diese in Projekten funktionieren und Ultimate Scrum um einfacher Kleinaufträge und Stories aus „Epics“ mischen zu können und um schnellere Durchlaufzeiten zu erreichen und einen klaren Fokus des Teams auf Fluss zu ermöglichen.
- Der Clou ist die Mittelschicht. Die Mittelschicht dient dazu die „operative Priorität“ der Stories in den Backlogs zu liefern – so dass jedes Team genau weiß in welcher Reihenfolge es die Stories ziehen muss um die im Portfoliomanagement genannten Termine erreichen zu können. Hier gibt es zwei bewährte Methoden a) Critical Chain Buffer Management – ergibt eine rot-gelb-grün Ampel und b) die sDBR-Ampel über die Restlaufzeit bis zum Liefertermin – auch wieder rot-gelb-grün mit allen Feinabstufungen. Das heißt man kann beides mischen. Wenn die Teams sich "einigermaßen" an die Ampel halten dann werden die Termine insgesamt gehalten.

Das Framework ist ein vollintegrierter Ansatz Projekt- & Produktionsteuerung der neuesten Generation - kombiniert mit den ganzen positiven Aspekten der agilen und Lean Methoden [MH14].

6 Praxisbeispiele

Im Rahmen dieses Beitrags ist es leider nicht möglich von allen Praxisbeispielen hier detailliert zu berichten. Daher beschränke ich mich hier auf einen kurzen Abriss, wo dieses Framework und in welcher Ausprägung im Einsatz ist.

Projekt	Situation	Lösung/Wirkung
Entwicklung einer App (Start-Up in Berlin)	Entwicklung einer innovativen App zum sicheren verteilten Abgleich von Kontaktdaten Zwei Firmen, vier Projektbeteiligte Projekt nach einem Jahr massiv verzögert und Beteiligte nur noch über Anwalt in Kontakt	(1) Reliable Scrum - Klärung des restlichen notwendigen Umfang und Abschätzung eines realistischen Termins (2) Überwachung des Termin mit Fieberkurve (3) Ultimate Scrum – optimaler Fluss, Vermeidung von Wartezeiten ➔ Release wurde pünktlich geliefert, Kommunikation deutlich verbessert

<p>Entwicklung Wire Swiss App/Website (Start-Up von Janus Friis ehem. Skype Gründer)</p>	<p>Entwicklung einer Kommunikationsplattform speziell Android-App</p> <p>14 Entwickler, zwei Standorte, 1 Jahr Entwicklung, laufende Verzögerungen</p>	<p>(1) Ultimate Scrum +50% mehr Durchsatz in vier Wochen</p> <p>(2) Reliable Scrum Termintreue und Verkürzung der Projektlaufzeit</p> <p>➔ App wurde früher fertig als die restlichen Bestandteile und in höchster Qualität</p>
<p>Ablösung von Softwareplattformen (1&1 Internet AG)</p>	<p>Drei Großprojekte zur Ablösung komplexer Altsoftwaresysteme (10 Jahre), jeweils ca. 20 Mitarbeiter, 4 Standorte, bis zu 25 Teams</p>	<p>(1) Reliable und z.T. Ultimate Scrum</p> <p>(2) z.T. ergänzt um Portfoliomanagement nach CCPM</p> <p>➔ alle drei Vorhaben pünktlich und in vollem Scope geliefert</p> <p>➔ durch Ultimate Scrum Konzept „Continuous Integration“ etabliert</p> <p>➔ höchste Agilität – neue Funktionen in 2-3 Tage implementierbar</p> <p>➔ massive Reduzierung der Entwicklungsaufwände für Folgeprojekte z.T. um Faktor weit über 100</p>
<p>Drei Mechatronische Produktentwicklungen (Internationale Unternehmen: Medizintechnik, Komponenten für Anlagenbau, Messsysteme)</p>	<p>Mechatronische Entwicklung, 250 bis 500 Entwickler und Konstrukteure, weltweite Standorte, weltweit bekannte Unternehmen, starker Marktdruck durch Globalisierung</p>	<p>(1) vorrangig Critical Chain Portfolio- und Projektmanagement</p> <p>➔ Steigerung der Taskerledigungsrate um ca. 50-70% in 4-6 Wochen</p> <p>(2) kombiniert Ultimate Scrum in Softwareentwicklung, Elektronik und Mechanik</p> <p>➔ Massive Verbesserung des Arbeitsfluss und der Kommunikation untereinander</p> <p>➔ höhere Motivation der Mitarbeiter</p>

<p>Embedded System Softwareentwicklung (Weltmarktführer im Automobilbereich)</p>	<p>Softwareentwicklung mit bis zu 4500 Entwicklern – weltweit verteilt</p> <p>Hochkomplexe Organisation im Spannungsfeld zwischen Kundenanforderungen und Plattformentwicklung.</p>	<p>(1) Ultimate Scrum in Engpassteams</p> <p>➔ +60 bis +70% Durchsatz in 2 Wochen</p> <p>➔ hohe Termintreue</p> <p>➔ Beruhigung des Arbeitsumfeldes der Mitarbeiter, Reduktion der Vorstandseskalationen um Faktoren</p> <p>(2) sDBR-Steuerung zur Synchronisation der Ablieferungen</p> <p>➔ höchste Zuverlässigkeit</p> <p>(3) nächster Schritt Critical Chain als Projektsteuerung in der Kundenorganisation</p>
--	---	---

Abgerundet wird das Framework in der Praxis durch eine konzentrierte Vorgehensweise zur Einführung basierend auf den Strategie & Taktik Bäumen [HA10] aus der „Theory of Constraints“ [UT10].

In diesen S&T-Bäumen ist die optimale Schritt-für-Schritt-Reihenfolge beschrieben, wie der Veränderungsprozess, unter Nutzung der Selbstorganisation, schnell und nachhaltig durchgeführt werden kann.

7 Fazit – Hybrid ist Pflicht!

Es gibt gar keine Alternative zu hybriden Ansätzen. Die unterschiedlichen Bedarfe von Projekten und Produkten (Agile) erfordern eine differenzierte Herangehensweise.

Durch Critical Chain (CCPM) wird die Projektorganisation schneller und agiler. Auf Seiten der agilen Methoden wird durch Reliable/Ultimate Scrum (advanced Agile) die Zuverlässigkeit und Geschwindigkeit erhöht, so dass beides in Kombination eingesetzt werden kann.

In Kombination mit einem Veränderungsprozess auf Basis der Strategie & Taktikbäume der TOC sowie unter Nutzung der Selbstorganisation kann der notwendige Veränderungsprozess deutlich beschleunigt werden. Das Risiko wird massiv reduziert und die Nachhaltigkeit steigt.

In der Verschmelzung von CCPM, advanced Agile und Veränderung basierend auf Selbstorganisation, liegt in der Praxis ein großes Potential mit Produktivitätssteigerungen von 50% bis über 100% bei gleichzeitiger Fokussierung auf den Kundennutzen.

Literaturverzeichnis

- [GC84] Eliyahu M. Goldratt und Jeff Cox “The Goal: A Process of Ongoing Improvement”, North River Press, 1984
- [Go97] Eliyahu M. Goldratt “Critical Chain”, Gower Publishing Ltd , 1997
- [Go08] E. Goldratt „Standing on the Shoulders of Giants: Production concepts versus production applications. The Hitachi Tool Engineering example”, Goldratt Consulting, 2008
- [GB10] Eliyahu M. Goldratt und Alan Barnard „Projects CO Strategy & Tactics Tree“, www.harmonytoc.com, 2010
- [Ca04] John A. Caspari “Management Dynamics: Merging Constraints Accounting to Drive Improvement”, John Wiley & Sons, September 2004
- [HMR14] Mike Hannan, Wolfram Müller und Hilbert Robinson “The CIO Guide to Break-through Portfolio Performance”, booklocker, 2014
- [TM13] Steve Tendon und Wolfram Müller “Hyper-Productive Knowledge Work Performance: The Tameflow Approach and Its Application to Scrum and Kanban”, J. Ross Publishing, 2013
- [Te07] Uwe Techt und Holger Lörz „Critical Chain: beschleunigen Sie Ihr Projektmanagement“, Haufe-Gruppe, 2007
- [Te10] Uwe Techt „Goldratt und die Theory of Constraints: Der Quantensprung im Management“, Editions La Colombe, 2010
- [Te14] Uwe Techt “Projects that Flow: mehr Projekte in kürzerer Zeit, die Geheimnisse erfolgreicher Projektunternehmen”, Ibidem-Verlag, 2014
- [WM12] Wolfram Müller “ Scrum + Critical Chain = Reliable Scrum: Das Beste aus zwei Welten kombinieren“, ProjektMagazin.de, Ausgabe 18/2012
- [WM13] Wolfram Müller „Schneller geht's nicht – Ultimate Scrum: Agiles Projektmanagement“, ProjektMagazin.de, Ausgabe 05/2013