

Critical Chain

Definition Critical Chain

Critical Chain Projektmanagement (CCPM) ist ein, auf Erkenntnissen der Systemtheorie basierender, Satz von Projektmanagementmethoden mit dem Ziel den Projektfluss zu verbessern.

Critical Chain erweitert das klassische Projektmanagement um die zwei Steuerungselemente (a) Vermeidung von schädlichem Multitasking durch effektive Begrenzung der Arbeitsmenge und Staffelung der Projekte am Engpass (b) Sicherstellung der Termintreue durch Zusammenfassung der Puffer am Projekteende sowie die operative Steuerung der Ressourcen über Fortschritt zu Pufferverbrauch.

Was heißt Critical Chain im Kontext von Projekten?

Durch diese ergänzenden Steuerungen werden Kapazitätsreserven frei, negatives Multitasking verschwindet, Risiken werden durch Projektpuffer und im Portfolio aufgefangen, der organisatorische Aufwand in der Projekt- und Portfoliosteuerung sinkt, die synchrone Verfügbarkeit der Ressourcen steigt und zu guter Letzt werden Verfrühungen nutzbar.

Durch die zentrale Visualisierung von Fortschritt zu Pufferverbrauch kommt es zu einer Fokussierung des gesamten Unternehmens auf die Verbesserung des Projektflusses – es wird ein operativ wirksamer und kontinuierlicher Verbesserungsprozess in Gang gesetzt. Critical Chain wirkt hierbei wie ein Katalysator, der alle anderen Anstrengungen (klassisches und agiles Projektmanagement, Lean Management, 6 Sigma, u.a.) verstärkt. Die Mitarbeiter erleben ihre Arbeit als planbar, konzentriert und produktiv was sich im Folgenden positiv auf Motivation, Qualität und Innovation auswirkt.

Durch die Einführung von Critical Chain steigt typischerweise die Zuverlässigkeit auf über 95 %, die Durchlaufzeiten der Projekte verkürzen sich um mehr als 25 % und in Folge steigt der Projektdurchsatz signifikant.

Hintergrund zu Critical Chain

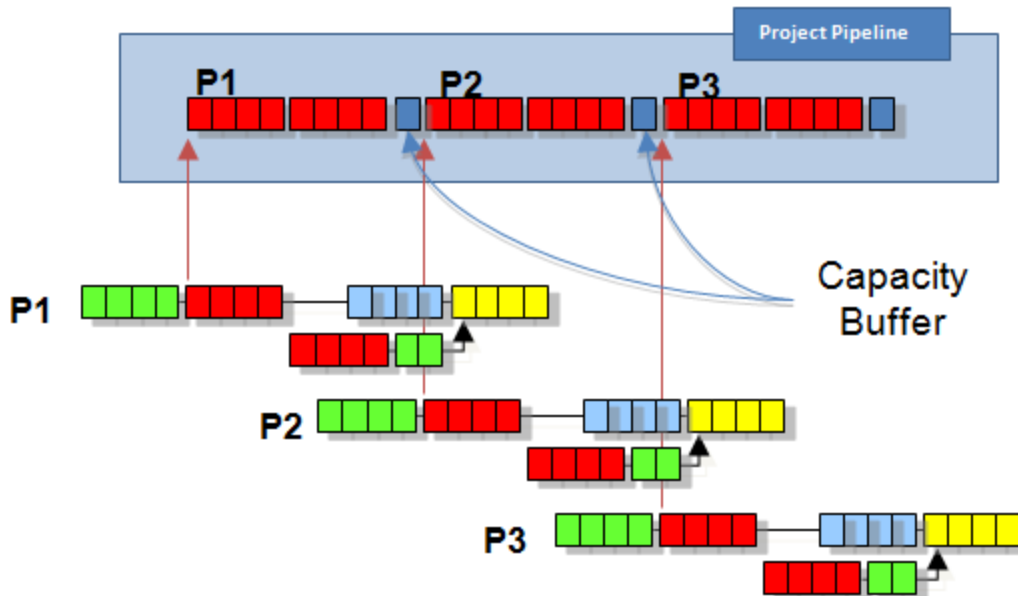
Im Folgenden werden die Element von Critical Chain und ihre Wirkung im Einzelnen beschrieben.

Begrenzung des "Work-In-Progress" (Einlastung)

Aus der Warteschlangentheorie oder Fertigungssteuerung ist das "Little's Law" bekannt. Es besagt "umgangssprachlich" nichts anderes, als dass, in einem stabilen System, die Durchlaufzeit eines Arbeitspaketes proportional zur Menge eingelasteter Arbeitspakete steigt. Die dümmste Idee ist also mehr Arbeit zu planen als man Kapazität zur Verfügung hat - oh Wunder!

Aber wie macht man das in einem komplizierten Unternehmen mit vielen hunderten Teams und vielen hunderten Projekten (also Multiprojektmanagement)?

Wieder hilft die Systemtheorie (in Form der [Theory of Constraints](#)). Diese besagt, dass je komplizierter ein System (z.B. ein Unternehmen) ist, desto stärker müsse es ein Element geben, dass das Ganze steuert. Daher ist der Artikel auch unter ganzheitlichem Projektmanagement angesiedelt. Im Multiprojektmanagement ist dies immer ein realer Engpass (es fehlen Ressourcen eines bestimmten Skills) oder ein virtueller Engpass (die Verfügbarkeit von vielen unterschiedlichen Skills). Wenn man diesen Engpass kennt, kann man alle Projekte anhand dessen planen (wie eine Pipeline) und durch einfache Abschätzung der Laufzeit vor und nach dem Engpass den Projektstart und das Projektende festlegen.



Auf dem Bild sieht man symbolisch drei Projekte mit Arbeitspaketen in unterschiedlichen Teams. Das rote Team stellt einen realen Engpass dar. Anhand dieses Teams (realer oder virtueller Engpass) werden die Projekte geplant und terminiert (Project Pipeline).

Anmerkungen aus der Praxis:

- Wenn in einem Unternehmen Ressourcenkonflikte "springen", sprich immer in unterschiedlichen Teams auftauchen, ist das ein deutliches Zeichen, dass insgesamt zu viel Projekte eingelastet sind. Wenn man die Menge der Projekte reduziert bis ein Team genau zu 100% ausgelastet ist hat man den Engpass gefunden und die Überkapazitäten der anderen Teams werden sichtbar.
- Die Projektpipeline kann auch eine größere Kapazität als "ein Projekt" aufweisen. In diesem Fall darf ein neues Projekt in die Pipeline sobald eines der anderen diese verläßt.
- Im Bild ist auch schon der "Capacity Buffer" angedeutet. Um zu vermeiden, dass Störungen im Engpass auf die restliche Planung durchschlagen werden die Arbeitspakete im Engpass abgepuffert. Der Puffer wird so eingestellt, dass die Menge an Störungen die durchschlägt erträglich wird.

Dieses Verfahren wird auch **Drum-Buffer-Rope-Steuerung** genannt, kommt in der Steuerung von Fertigungssystemen seit Jahren zum Einsatz und stellt nachhaltig sicher, dass die Einlastung (Work-In-Progress) so begrenzt wird, dass die Durchlaufzeit und der Projektfluss optimal wird.

Puffer ans Projektende

Bei zweiten Teil von Critical Chain hilft die Psychologie, Wahrscheinlichkeitsrechnung und Versicherungsmathematik (dafür ist sie wirklich mal gut!). Es dreht sich alles um Schätzungen und wie sie wirklich funktionieren.

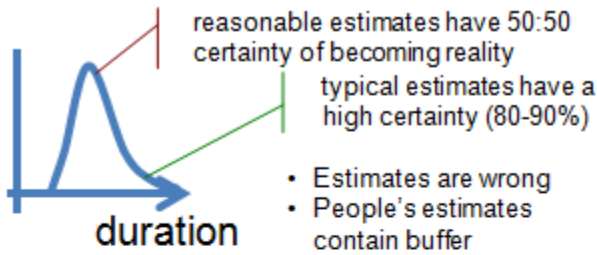
Grundlegende Eigenschaften von Schätzungen:

- Schätzungen sind immer falsch - je genauer man versucht zu schätzen umso "falscher" werden sie
- Schätzungen sind nie ein Wert - es sind immer Funktionen der relativen Wahrscheinlichkeit des Auftretens eines Wertes. In der Praxis ist es ausreichend drei Punkte zu bestimmen: a) den optimistischen Wert (0% rel. Wahrscheinlichkeit) bei dem alle glatt läuft b) den realistischen Wert (der mit der relativ größten Wahrscheinlichkeit) und c) den pessimistischen (wenn fast alles schief geht - abgesehen von Meteoriteneinschlägen und Weltuntergängen, o.ä.) - fast 0% (so 1-5%) relative Wahrscheinlichkeit
- Schätzungen werden nie unterschritten - dafür sorgen a) das **Studentensyndrom** (wenn man weiss man hat Puffer und vieles andere zu tun fängt man später an) und b) **Parkinsonsches Gesetz** (es wird immer so viel Zeit gebraucht wie zur Verfügung steht)
- Schätzungen werden oft überschritten - dafür sorgt das Studentensyndrom (s.o.) in Kombination mit c) **Murphys Gesetz** (das eigentlich kein Gesetz ist sondern eine Lebensweisheit - dass immer etwas schief geht - vor allem wenn man es nicht gebrauchen kann weil man ja dem Studentensyndrom erlegen ist)

Also ganz einfach - es gibt keine Verfrühungen nur Verspätungen. Der Trick von Critical Chain ist nun ganz einfach - typischerweise sind die Hälfte der Schätzungen Puffer/Reserve. Je größer der Druck (z.B. WIP) desto größer der Puffer (in der Praxis findet man z.T. 10fach Puffer). Wenn man alle Arbeitspakete in der Dauer (nicht im Aufwand) halbiert und alles was man gewinnt an den Schluß des Projektes packt dann hat man das Studentensyndrom, Parkinson und Murphi ausgehebelt und das Problem mit Schätzungen gelöst.

Aber jetzt kommt noch das mit der Versicherungsmathematik. Ein einfaches Beispiel: wenn nur jedes 1000te Haus ab und zu abbrennt und man ein Haus versichern müssten - müsste man trotz dem den ganzen Wert des Hauses Versicherungsprämie und Sicherheit zurücklegen. Wenn man aber 1000 Häuser versichern will dann braucht man immer noch nur die Sicherheit von einem Haus und die Versicherungsprämie sinkt auf ein Tausendstel.

Das gleiche gilt für Schätzungen in Arbeitspaketen - schon wenn man nur 5 Arbeitspakete hintereinander abarbeiten muss sinkt die Wahrscheinlichkeit drastisch, dass alle Verspätungen im gleichen Projekt auftreten. In der Praxis kann man daher den Projektpuffer typischerweise um 50% kürzen. Dabei entsteht ein Risiko für die Termineinhaltung die aber <5% liegt. Das sind Erfahrungswerte und können von Situation zu Situation variieren.



#1 longest chain of tasks



#2 50% buffer highlighted (green)



#3 aggregate the buffer at the end



#4 cut the buffer in half



Das Bild zeigt die typische Wahrscheinlichkeitsfunktion von Schätzungen und die schrittweise Anwendung in der Projektplanung. Hierbei wird vor allem die kritische Kette betrachtet (also die längste Kette von Arbeitspaketen und Berücksichtigung der Ressourcenverfügbarkeit) - daher auch der Name Critical Chain.

In der Praxis gewöhnen sich Systeme an diese Verkürzung und nach einer gewissen Lernzeit schwenkt man um auf eine Verkürzung der Arbeitspakete um 66% ohne Anpassung des Endtermins. Wichtig ist, dass man einen Projektpuffer erhält, der ca. 30% der Projektdauer umfasst.

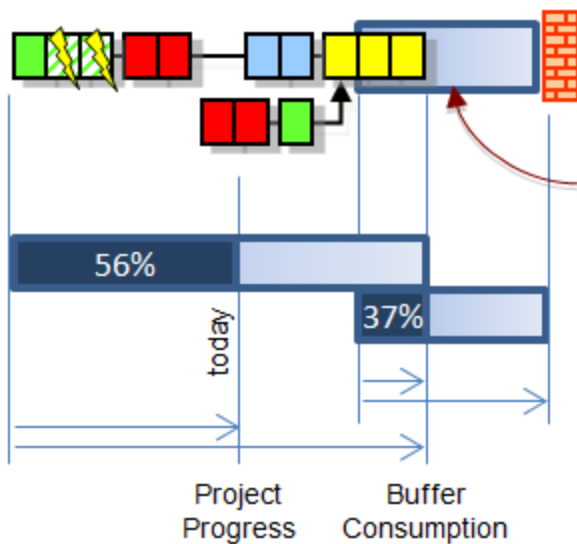
Execution Control oder operative Projektsteuerung

Kennen sie Projektampeln? Ja! Kennen sie die Farbe "Melonengrün"? Nein! Melonengrün heisst nichts anderes als das Projekt ist "grün" - aber je tiefer man bohrt desto "roter" wird es 😊

Wenn man einen Projektpuffer hat kann man endlich operativ aussagekräftige Projektampeln definieren. O.k. es ist keine Ampel mehr sondern eine Fieberkurve - aber die Farben sind die gleichen.

Das Prinzip ist einfach:

- Wenn der Fortschritt auf der kritischen Kette größer ist als der Pufferverbrauch > grün
- Wenn der Fortschritt auf der kritischen Kette kleiner ist als der Pufferverbrauch > rot
- Der Übergang von grün auf rot ist gelb (ist zwar komisch aber so ist es halt bei Ampeln)
- Wenn der Puffer verbraucht ist > schwarz = tot



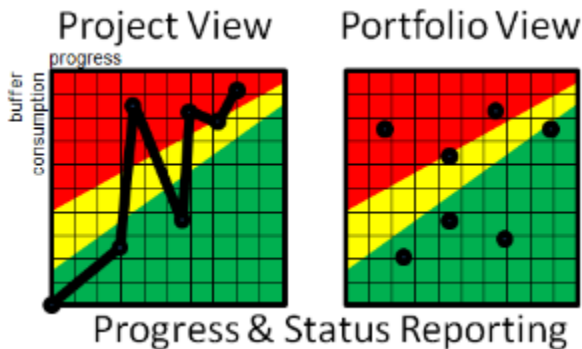
33% of the plan
is a project buffer
placed before the
due date



status "green" → project is progressing
faster than buffer consumption

status "red" → project is progressing more
slowly than buffer consumption

In der Praxis wird das in eine Fiberkurve eingetragen. Jedes Projekt beginnt links unten bei Fortschritt 0% und Pufferverbrauch 0% und wandert über die Zeit nach rechts oben 100% Fortschritt und wenn möglich ein bisschen Puffer übrig.



Progress & Status Reporting

Die Projektsicht dient vor allem dazu das Projektteam auf Kurs zu halten. Die Wirkungen sind typischerweise:

- das Team erhält schnell Feedback über den Fortschritt und echten Status
- wenn das Projekt in die rote Zone gerät fokussiert sich das Team auf die Rückgewinnung des Puffers (selbständig)
- die Projektprobleme werden schneller sichtbar
- die Projektpläne werden steuerbarer
- die Verantwortlichkeit des Team für den Termin steigt
- das Team wird gestärkt

Die Portfoliosicht dient vor allem den Stakeholdern den Führungsebenen. Die Wirkungen sind typischerweise:

- transparente und visuelle, schnell zu erfassende Information über den Zustand des kompletten Portfolios
- objektiver Status täglich
- Vertrauen in die Projekte und Teams steigt und damit weniger Reporting und µManagement
- Information über den Belastungszustand des Unternehmens - wenn mehr als 10% der Projekte rot sind ist das System nicht mehr selbststeuernd - dann muss die Last reduziert werden

Und das gibt es noch viel mehr was plötzlich passiert

- damit ein Projekt an Ressourcen kommt - braucht es einen Projektplan. Ein guter Plan besteht aus Arbeitspaketen und Verantwortlichen die diese liefern. Um Arbeitspakete zu bekommen braucht man ein sauberes Konzept. Um an ein sauberes Konzept zu kommen muss man den Auftrag klären > Critical Chain wirkt wie ein Katalysator der die ganze guten Projektmanagement-Tugenden zum tragen bringt.
- Wenn man die Ressourcen operativ immer so zuteilt, dass das Projekt das am "rotesten" ist die Ressourcen bekommt - dann ist

sichergestellt, dass immer dem Projekt mit der größten Schieflage geholfen wird. Wenn der Work-In-Progress richtig begrenzt ist (s.o.) dann ist so sicher gestellt, dass 95% der Projekte den Termin halten.

- Wenn man die Ressourcen operativ je nach Schieflage den Projekten zuteilt kommt es (neben den Projektpuffer) zu einer weiteren Aggregation der Risiken. Und damit kann man den Projektpuffer weiter verringern! Ja das passiert tatsächlich.
- Wenn man auswertet, welche Arbeitspakete wie viel Puffer fressen (konsumieren) und abfragt was der Grund hierfür war hat man den Schlüssen für echte kontinuierliche Verbesserung. Man muss nur das Team raussuchen dessen Arbeitspakete den meisten Puffer fressen und genau hier muss man ansetzen um Durchlaufzeit nachhaltig zu verringern.
- Um den Fortschritt und Pufferverbrauch auszuwerten muss man täglich (zweitäglich) die Restlaufzeit der Arbeitspakete rückmelden. Dafür gibt es Software die hier hilft. Als Dank bekommt man täglich aktuelle Projektpläne.
- Weil es Puffer gibt ist es auch nicht schlimm, wenn mal ein Arbeitspaket hinzu kommt.
- Wenn ein Arbeitspaket nicht auf der kritischen Kette liegt oder nicht im Engpass dann kann man es einfach machen. Change Request werden damit extrem einfach.
- u.s.w. u.s.w.

Hintergrund

[Critical Chain](#) wurde 1997 von [E. Goldratt](#) beschrieben und seither in weit über 2000 Firmen weltweit eingeführt. Es gibt unzählige [Erfolgsberichte](#) mit z.T. unglaublichen Effekten (nicht wundern! staunen).

Und nun?

- Wo finde ich eine Übersicht über die Erfolgsgeschichten? z.B. bei [Realization](#), [VISTEM](#), [Speed4Projects](#) oder googlen
- Wo finde ich [Literatur](#): E. Goldratt "das Ziel" und "die kritische Kette" oder U. Techt "Goldratt und die Theory-of-Constraints"
- Wo gibts Seminare zu Critical Chain? [Speed4Projects](#), [TOC-Institut](#) oder [Projekthaus-Stuttgart](#) oder Management-Circle oder googlen
- Wie kann ich checken, wie viel Potential in meinem Unternehmen steckt? [5min-Quickcheck!](#)
- Wie kann ich es ausprobieren? Es gibt ein [Selbstlern-Kit!](#)
- Was sagt die GPM dazu? Auch die guten "alten" haben noch was drauf - es gibt eine [Fachgruppe Critical Chain](#) - die ist sogar recht aktiv
- Was sagt PMI dazu? Im PM-Book-Guide v2 auf Seite 174 steht "Critical Chain ist eine gute Idee Projekte zu planen" - na ja das ist ein Grund dafür, dass es OpenPM gibt!
- Gibt es eine Übersicht über Critical Chain? Ja - [Critical Chain on an Page](#) (o.k. auch von mir - und die Bilder sind die gleichen)

Beiträge auf openPM zu Critical Chain

erscheinen hoffentlich bald viele ...