1001 Nacht
Das XP Überlebenstraining

Prof. Dr. Nikolaus Wulff
Fachhochschule Münster

Eine kleine Märchenstunde

... Denn ist nicht das Schreckliche wie das Wunderbare zumeist die Folge des Unbeachteten und Unbedachten? Und wächst nicht das Große aus den Kleinigkeiten?

So ist es in den alten Märchen und so ist es auch in unser aller Leben...

Scherazade

Der Esel und der Ochse

- ... der Esel ermunterte den Ochsen sich krank zu stellen, worauf der Ochse am nächsten Tag im Stall verblieb und bei gutem Futter wieder zu Kräften kommen sollte.
- Am nächsten Morgen aber wurde der Esel – statt des Ochsen –, vom Bauern vor den Pflug gespannt und musste dessen Arbeit verrichten...

> ScrumMaster:
> Sei kein Esel, mach' dich nicht für Andere zum Ochsen!

Der Fischer, der eine Flasche fand

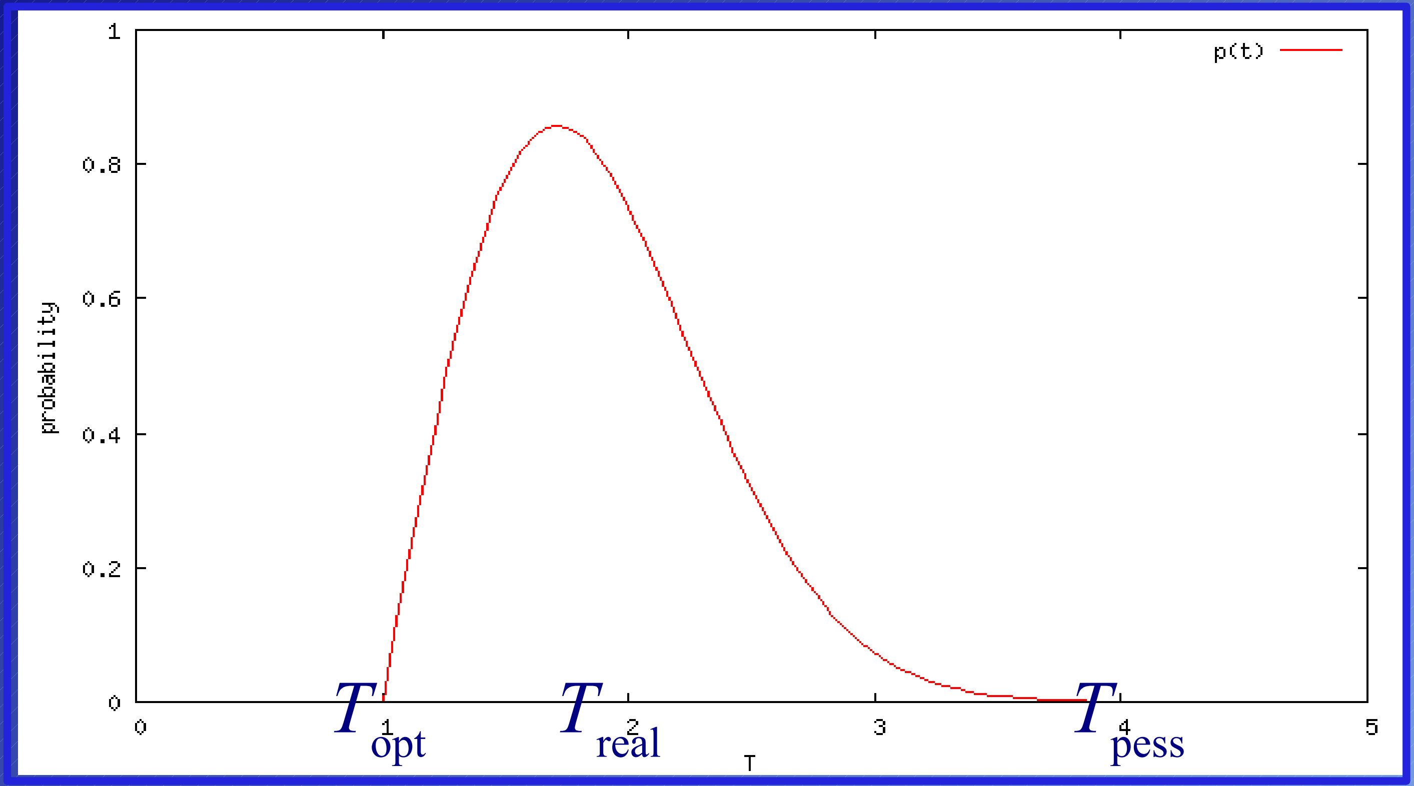
... O König der Zeiten, weißt du wie groß die Entfernung ist zwischen meiner Stadt und der deinen?

Und der König sagte: Drei Stunden ritten wir.

O König, wenn du träumst, so wache auf, denn um von hier nach deiner Stadt zu reisen, brauchst du ein ganzes Jahr.
... und wenn es wirklich gelang in so kurzer Zeit, so geschah das wohl mit Geisterhilfe, oder weil die Stadt verzaubert war.

Woher kommen Termine?

Fertigungstermine schätzen



- Wahrscheinlichkeitsverteilung *p(t)* ein Arbeitspaket zwischen dem optimistischen Termin $T_{opt} = 1$ und dem pessimistischen Termin $T_{pess} = 4$ fertigzustellen.

Wann ist der Fertigstellungstermin?

- Die Wahrscheinlichkeit *P(T)* das Paket zu irgend einem Zeitpunkt $T_{\text{opt}} < T < T_{\text{pess}}$ zu beenden ist

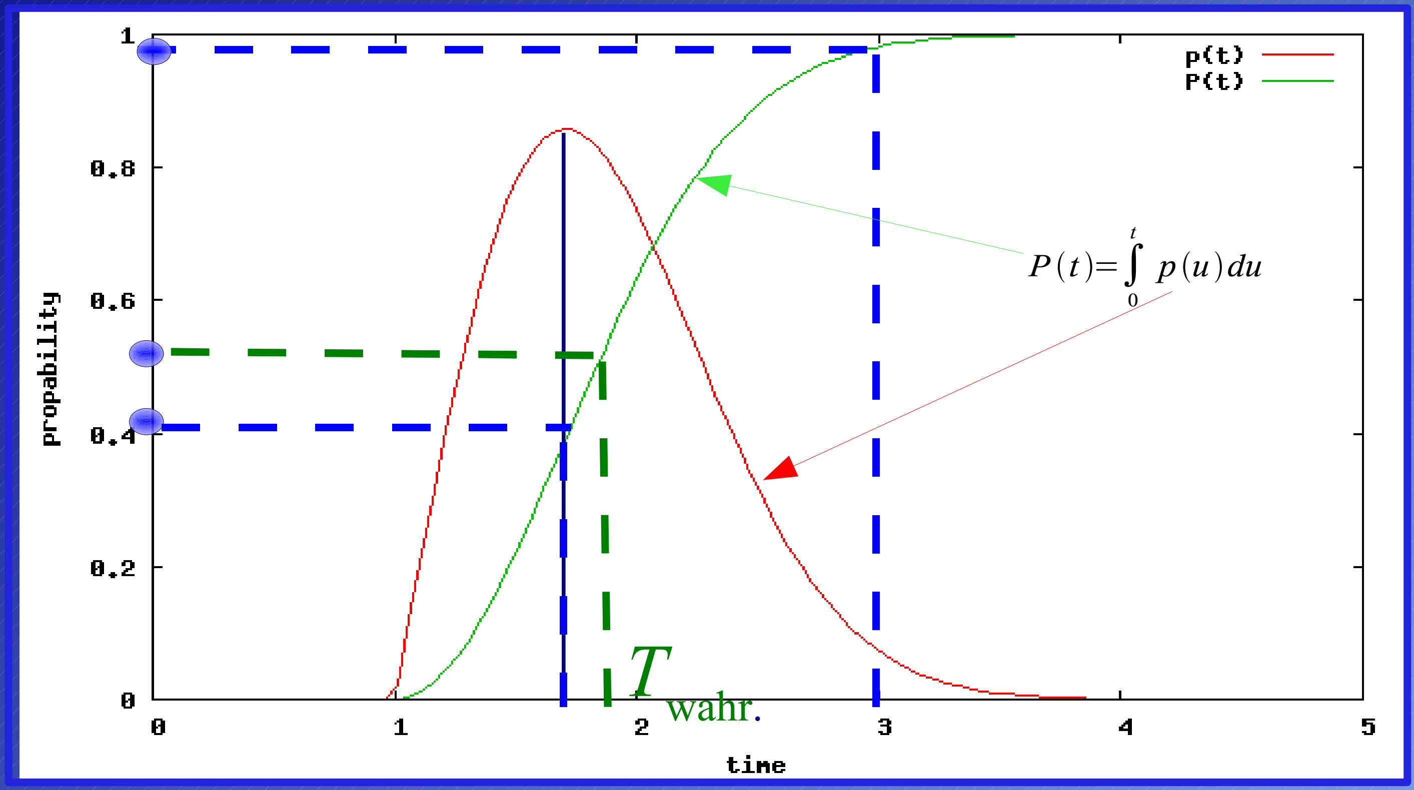
$$P(T) = \int_0^T p(t)\,dt$$

- und der wahrscheinlichste Endtermin T_{wahr} ist

$$T_{wahr.} = \int_0^\infty t\,p(t)\,dt$$

- Auf jedem Fall ist die Chance zu T_{opt} fertig zu werden gleich Null! (und zu T_{pess} gleich Eins)
- Wenn wir doch nur *p(t)* kennen würden...

Mögliche Endtermine



- Der realistische Termin 1.7 hat nur eine Chance von ~43%. Auch dieser Termin ist sehr optimistisch!
- Der wahrscheinliche Termin T_{wahr}=1.9 hat eine 54% Chance und mit 96% wird der Termin t=3 gehalten.

Nährungen für p(t) bzw. T_{wahr}

- In der Realität ist p(t) unbekannt.
- Wenn gar nichts bekannt ist, wähle eine Gleichverteilung $p(t) = (t - t_{pess.})/(t_{opt} + t_{pess})$.

$$T_{wahr} = \frac{t_{opt} + t_{pess}}{2}$$

- Eine bessere Approximation ist die Drei-Punkte-Form:

$$T_{wahr} = \frac{t_{opt} + 4 * t_{real} + t_{pess}}{6}$$

Simpson-Integration

Falsche Aussagen

- Das Beispiel zeigt, es ist sehr schwer den Zeitpunkt genau vorherzusagen.
- Definitiv falsch ist es den optimistischen Termin als Auslieferungstermin zu wählen. Dieser hat gar keine realistische Chance!
- Auch der wahrscheinlichste Termin ist schwer zu halten. Es gibt lange Ausläufer, so dass in fast 2/3 aller Fälle der Termin nicht zu halten ist. Wohl dem PL der sich und seinen Kunden darauf vorbereitet hat...

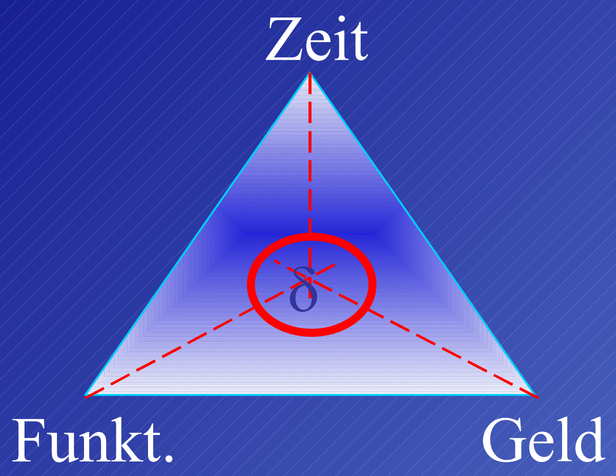
Aladin und die Wunderlampe

... Aladin war ein Kind, das mehr in seinen Träumen lebte als im Alltag, und er glaubte, dass sich ihm die Welt durch Wunder, nicht aber durch emsiges Wirken in den Werkstätten erschließen würde ...

- Noch heute glauben viele Manager und Projektleiter an Wunder und verkennen:

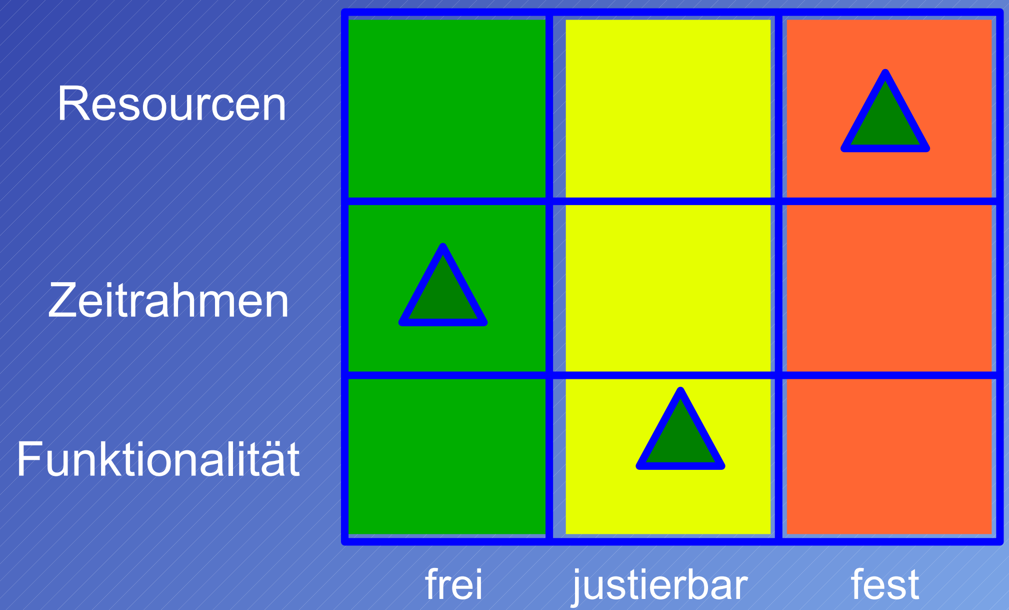
$$\boxed{\text{Verrichtete Arbeit} = \text{Leistung} * \text{Zeit}}$$

Zielkonflikt: Die Quadratur des Kreises



Die unmögliche
Punktlandung

Microsoft Solution
Framework (1997)



Projekte sind
Kompromisse

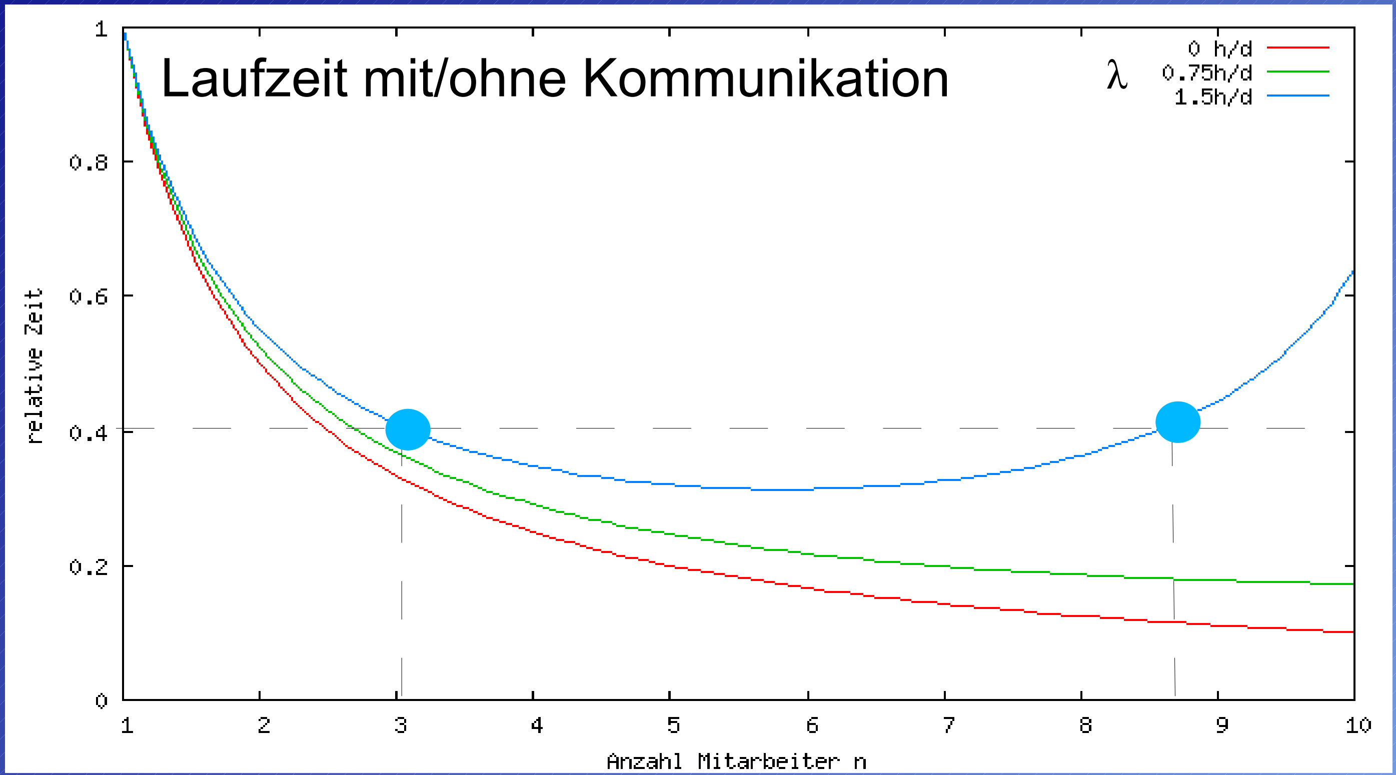
Zeit und Teamgröße

- Die benötigte Zeit für ein Arbeitspaket ist: $\boxed{T \sim \frac{1}{n}}$
- Die effektive Arbeitsleistung wird verringert um den „Verlustaufwand“ für die Kommunikation der Mitglieder untereinander: $\boxed{n \rightarrow n_{eff} = n - \lambda \frac{n(n-1)}{2}}$
- Die Zeit $T(n)$ zur Erstellung eines Arbeitspaketes als Funktion der Teamgröße n und der relativen täglichen Kommunikation $\lambda = x/8h$ beträgt:

$$\boxed{T(n) = \frac{T(1)}{n[1 - \lambda(n-1)/2]}}$$

Entwicklungszeit
für *n* Teammitglieder
λ = Komplexitätsfaktor

Laufzeit versus Teamgröße



- Ein 3 Mann Team leistet in der selben Zeit genauso viel wie ein 9 Mann Team, wenn ein Abstimmungs-bedarf von 1.5 Stunden pro Tag besteht!

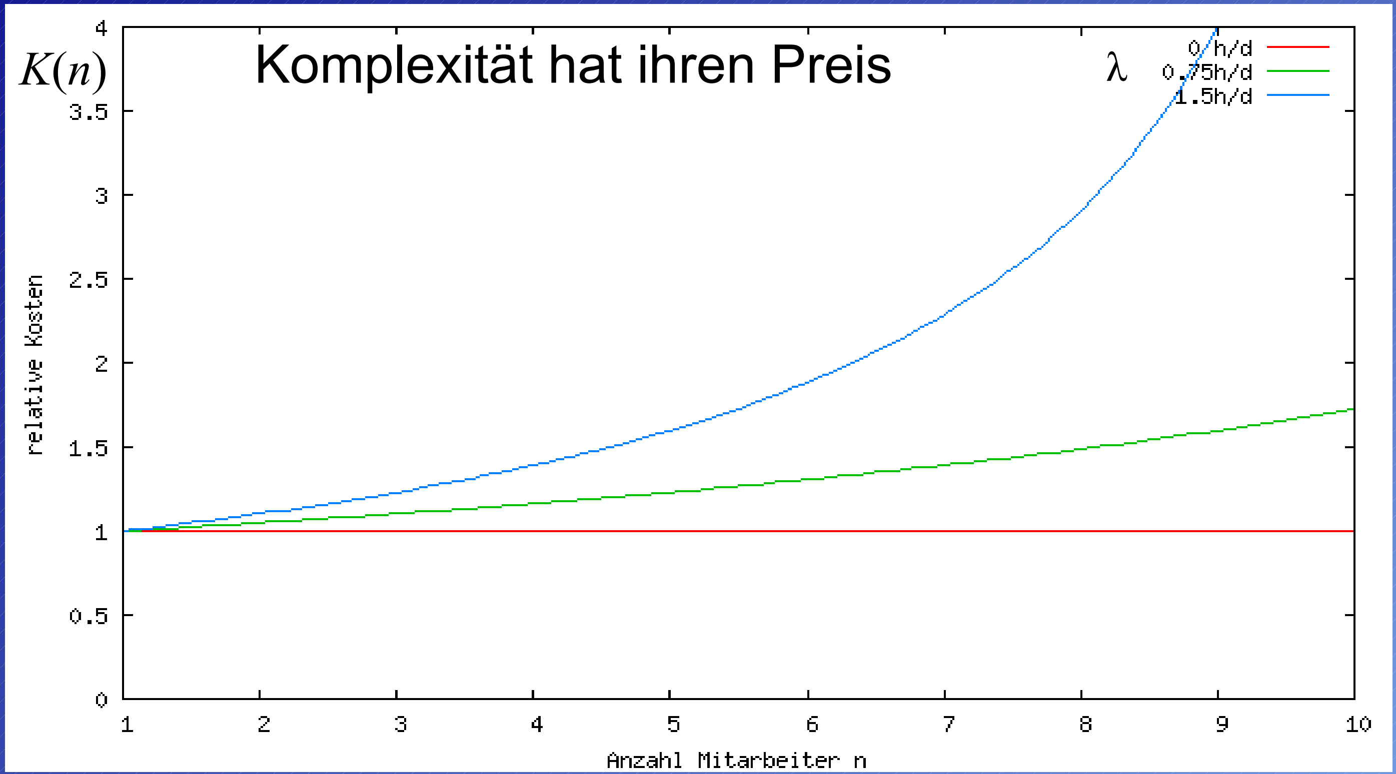
Minimale Laufzeit

- Das Modell gestattet die Bestimmung der Teamgröße, um ein Arbeitspaket in minimaler Zeit zu erstellen. $\boxed{\frac{dT(n)}{dn}=0} \Rightarrow \boxed{n_{opt}=\frac{1+2/\lambda}{2}}$

Optimale Teamgröße für minimale Entwicklungszeit
λ = Komplexitätsfaktor

- In Phasen hoher Kommunikation (Analyse, Systemarchitektur...) mit ~2 h Abstimmung/Tag liegt die Teamgröße bei n_{opt} = 4 - 5 Mitarbeitern.
- Bei λ =1h/Tag liegt die optimale Teamgröße bei n_{opt} = 8 - 9 Mitarbeitern.

Kosten können explodieren



- Mit steigender Komplexität steigen die Lohn-kosten für ein Arbeitspaket überproportional an. Das „Chinesen Prinzip“ funktioniert nicht...

Kommunikation eliminieren?

- Die Kommunikation lässt sich nicht eliminieren.
  - Dies ist nur bei völliger Unabhängigkeit der Mitarbeiter voneinander möglich.
  - Dies mag bei „Spargelstechern und Fließband-arbeitern“ möglich sein, jedoch
- nicht bei Softwareentwicklern!

- Ohne ausreichende Kommunikation der Entwickler leidet die Qualität und Funktionalität der Software.

Teamarbeit hat ihren Preis

- Softwareentwicklung ist ein kreativer Prozess und erfordert Teamarbeit.
- Wichtig ist die Kommunikation im Team.
- Kommunikation hat einen Preis, der häufig in der Planung nicht berücksichtigt wird.
- Das Zeit/Kosten-Verhalten ist nichtlinear.
- Ansätze SWE wie Fließbandarbeit zu planen und zu steuern versagen deshalb leider immer wieder.

Optimale Teambildung

- Wird ein Projekt geschickt aufgesetzt, so lassen sich die Kosten der Kommunikation minimieren bei einem gleichzeitigen Qualitätsgewinn.
- Ein Projektleiter kann hier ähnlich vorgehen wie ein Softwarearchitekt, der in einer Schichten-architektur die Schnittstellen optimiert.
- Nach dem „Teile-und-Herrsche-Vorgehen" wird das Projekt in kleinere geeignete Subprojekte zerlegt, um Abhängigkeiten zu minimieren.
  - Es gibt eine natürliche kleinste Paketquantelung...

Kosten und Teamgröße

- Angenommen die Lohnkosten sind der wichtigste (einzige) Faktor bei der Softwareentwicklung.
- Die naive Annahme ist, dass bei Verdopplung der Teamgröße eine Halbierungr der Entwicklungszeit bei konstanten Entwicklungskosten eintritt.
- Die Effektivität des Teams nimmt mit wachsender Größe ab, so dass sich die Entwicklungskosten *K(n)=n*T(n)* trotz der (scheinbaren) Zeitersparnis zunehmend erhöhen!

$$K(n) = \frac{K(1)}{1 - \lambda(n-1)/2}$$

Entwicklungskosten bei *n* Teammitgliedern
λ = Komplexitätsfaktor

Ein Gedankenexperiment

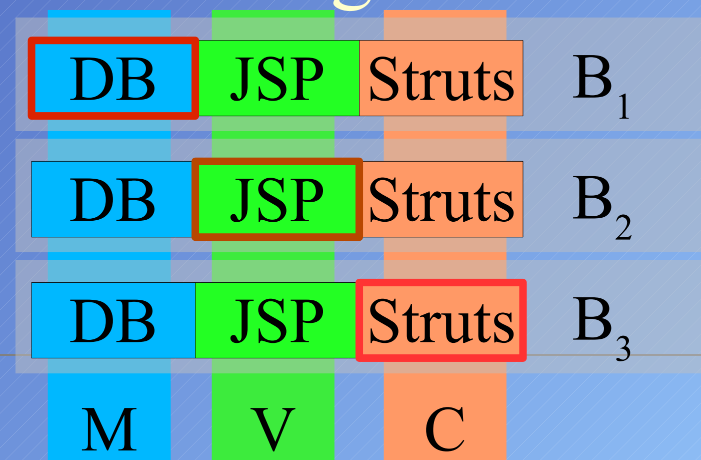
- Ein Produkt, bestehend aus drei Anwendungsfällen, soll mit Web-Technologie entwickelt werden.
- Es wird eine MVC II Architektur bestehend aus JSPs, Struts-Controller und einer DB verwendet.
- Es liegen vier (gleichgroße) Arbeitspakete vor:
  – Die Systemarchitektur und die drei Anwendungsfälle.
- Es werden zwei Projektteams A und B aufgesetzt.
- Projekt A startet gleich mit 9 Mitarbeitern.
- Projekt B startet mit einem 3-köpfigen Architekturteam und stockt dann um 6 Mitarbeiter auf, in dem es drei Subteams B_1, B_2 und B_3 bildet.

Die Team B Strategie

- Das Architektur-Team beschäftigt sich mit dem generischem Framework für JSP, Struts und DB.
- Zeit und Kosten für ein Arbeitspaket betragen:

$$T_{B_0} = \frac{T_1}{3(1-\lambda)} \qquad K_{B_0} = \frac{K_1}{1-\lambda}$$

- Anschließend werden die neuen Mitglieder der drei Subprojekte von den Architekten integriert und sie teilen sich auf.



Je nach Erfordernis agieren eher drei fachliche oder drei technische Subteams.

Team A versus Team B

- Team A hat von Anfang an die volle Teamstärke.
- Alle Mitarbeiter besitzen ähnliche Qualifikation.
- Gesamtzeit und -kosten der Entwicklung*) sind:

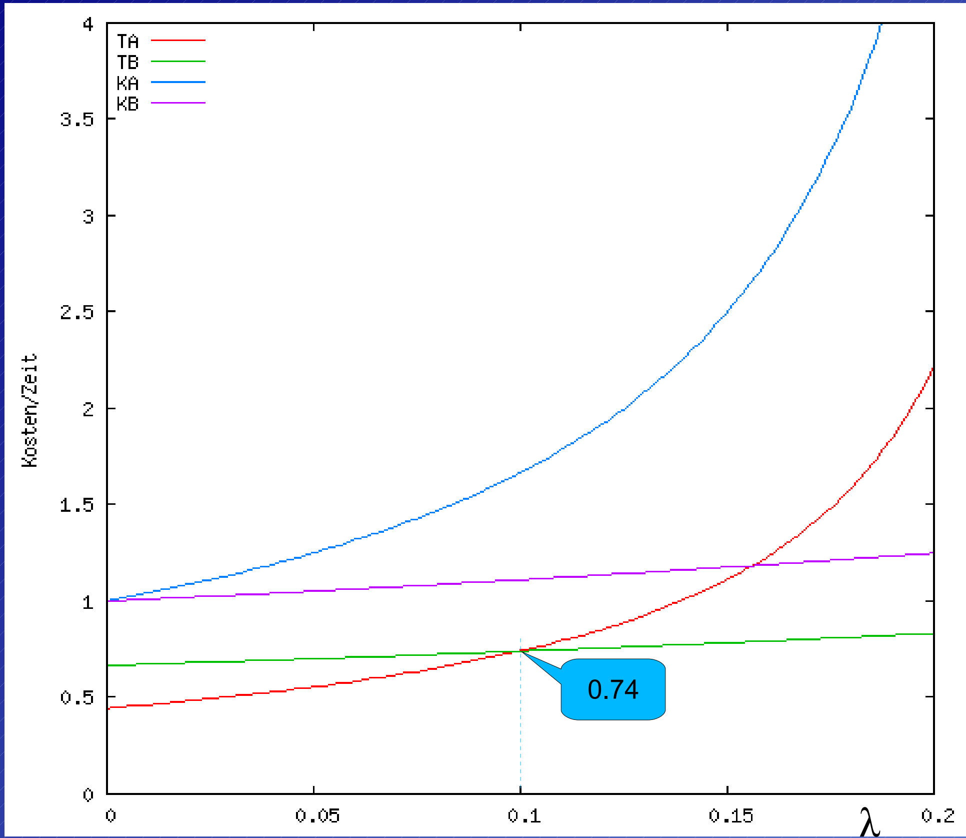
$$T_A = \frac{4\,T_1}{9(1-4\lambda)} \rightarrow 0.44 \qquad K_A = \frac{4\,K_1}{1-4\lambda}$$

$$T_B = \frac{2\,T_1}{3(1-\lambda)} \rightarrow 0.66 \qquad K_B = \frac{4\,K_1}{1-\lambda}$$

- Für $\lambda \rightarrow 0$ benötigt das B Team 1.5 mal länger als das A Team. Der genaue Verlauf hängt vom λ-Parameter ab, d. h. wie „problemträchtig“ die Entwicklung ist.

*) Die Übergangsphase des 3-er zum 9-er Team wurde vernachlässigt...

Vergleich von Team A und B



Verlauf von Zeit und Kosten versus λ.

$$\delta T_A = \frac{0.74}{0.44} = 1.68$$

$$\delta T_B = \frac{0.74}{0.66} = 1.12$$

- Team A hat im besten Fall die kürzere Laufzeit 0.4 während Team B 0.6 Zeiteinheiten benötigt.
- Bei einem komplexen Projekt schlägt schnell der λ-Faktor zu.
- Bei λ = 48min/d überholt Team B das A Team.
- Die Kurven von Team B sind insgesamt flacher und λ unemfindlicher.

Brook's Gesetz

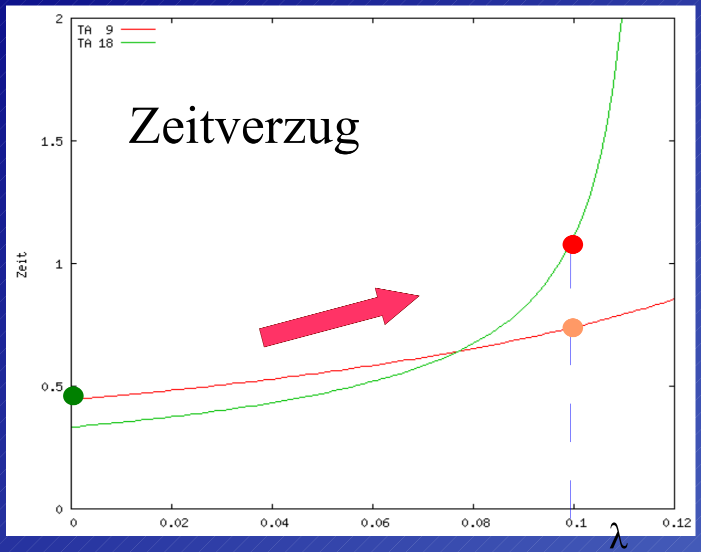
- Nachdem die Hälfte der Anwendung realisiert ist, signalisiert Team A dem Management, dass es nicht rechtzeitig fertig werden wird. (Zeitpunkt?!)
- Es wird entschieden Team A zu verdoppeln auf 18 Mitarbeiter, um den Termin zu halten...

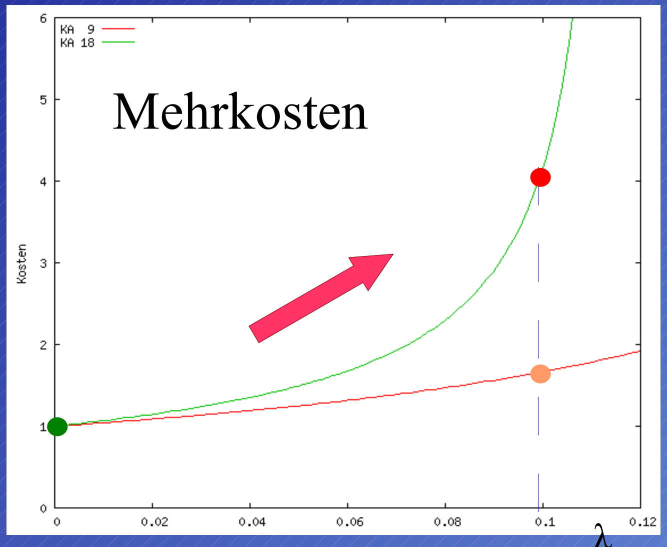
$$T_A = \frac{2T_1}{9(1-4\lambda)} + \frac{2T_1}{18(1-8.5\lambda)} \qquad K_A = \frac{2K_1}{1-4\lambda} + \frac{2K_1}{1-8.5\lambda}$$

- Bei λ=0.1 dauert das A Projekt nun 2.5 mal länger und ist 4 mal teurer als die Planwerte des A Teams.

Adding manpower to a late software project makes it later.

Ein Horrorszenario





- Die Verdoppelung des Teams lässt den Termin endgültig platzen.
- Warnzeichen waren schon bei 25% T_{soll} absehbar...
- An die Kosten und den Schaden ist gar nicht zu denken.
- Dieses Modell sieht aus wie ein „Sandkastenspiel“ aber es wird leider jeden Tag erneut Realität...

Nur ein Modell?

- Es handelt sich um ein sehr vereinfachtes Modell, das mit Vorsicht interpretiert werden muss.
- Die generellen Trendaussagen sind hingegen richtig.
- Die geometrische Reihe beginnt linear und entwickelt sich dann hochgradig nichtlinear:

$$\frac{1}{1-x} = \sum_{j=0}^{\infty} x^j = 1 + x + x^2 + x^3 + \ldots \rightarrow 1 + x \,,\; \textit{für } x \ll 1$$

- Menschen denken in linearen Zusammenhängen – die Wirkung ist direkt proportional zur Ursache –, exponentielle/hyperbolische Verstärkung ist entgegen unserer Evolution und schwer fass- und planbar.

Das war' s...

- Märchen gehören zum Projektgeschäft
- Jedoch es gilt daraus zu lernen.

Der Weise tut am Anfang,
was der Narr am Ende tut.