

# Verschätzt!

Warum wir beim Schätzen meistens daneben liegen...  
und wie es besser geht

Martin Grotz, MATHEMA GmbH





# Agenda

- Definitionen
- Probleme
- Lösungen



# Über was reden wir nicht

- Einzelne Arbeitspakete für einen Sprint abschätzen
- Agile Schätzmethoden
- Riesenprojekte



# Definitionen





# Definitionen

## Target/Ziel

Gewünschtes Geschäftsziel: "wir brauchen X, damit Y"

## Commitment

Zusage: "Wir liefern X zum Zeitpunkt T in Qualität Q"

## Estimate/Schätzung

Fragen: "Wie lange wird X dauern? Wie viel wird X kosten?"



# Schätzung vs. Plan

## Schätzung

- vorurteilsfrei
- faktenbasiert
- analytisch

## Plan

- ergebnisoptimiert
- was wollen wir erreichen
- wie wollen wir es erreichen



# Nutzen einer guten Schätzung

Gute Schätzungen bringen:

- Echte Statusinformation
- Höhere Qualität
- Einfachere Kommunikation
- Planungssicherheit (Kunde und wir!)
- Vertrauen
- Erkannte Abweichungen liefern konkreten Anlass zur Risikobewertung



# Probleme







# Das Ziel der Schätzung

- "Herausfinden, ob Ziel, Plan und Aufwand nah genug beieinander liegen, um mit Hilfe der Projektsteuerung realistisch verknüpft werden zu können"
- Dem Projektmanager eine solide Basis für Entscheidungen liefern



## Wie sicher bist du dir?

- Aussagen wie "zu 90% sicher" machen nur Sinn, wenn es eine Faktenbasis gibt
- Sicherheit wird fast immer zu optimistisch angegeben (Realität: Leute denken 90%, sind aber eher 30% Treffsicherheit)
- Schätzung: je unsicherer, desto größer muss die Spannweite sein



# Wahrscheinlichkeiten und Gefahren

- Exakte Schätzung: "Es dauert 62 PT" 💣
- Erfolgchance von Projekten: Glockenkurve? 💣
- Es gibt eine untere Grenze, wie gut es laufen kann, **aber keine obere Grenze, wie schlecht**
- "Diseconomy of scale"





# "Die Projektsteuerung rettet alles!"

- Projektsteuerung bringt Schätzung, Ziel und Commitment zusammen
- Gute Projektsteuerung kann bis zu 20% Fehler bei der Schätzung ausgleichen
- Je größer die Abweichungen, desto mehr Energie und Arbeit braucht die Steuerung



# Zuviel oder zuwenig

- zu groß schätzen?
  - Parkinson's Law (Arbeit füllt immer die zur Verfügung stehende Zeit)
  - "Gold-plating" Gefahr
- zu klein schätzen?
  - Projektpläne sind wertlos
  - Steuerungsaufwand steigt (Krisenmeetings, Kommunikation mit dem Kunden, ...)
  - Qualität sinkt
  - Entwickler schätzen eh schon immer zu optimistisch



# Systemisches Problem

- Praktisch alle Schätzungen sind zu klein!
- Durchschnittliche echte Projektdauer: Schätzung **+120%**
- Durchschnittliche echte Projektkosten: Schätzung **+100%**

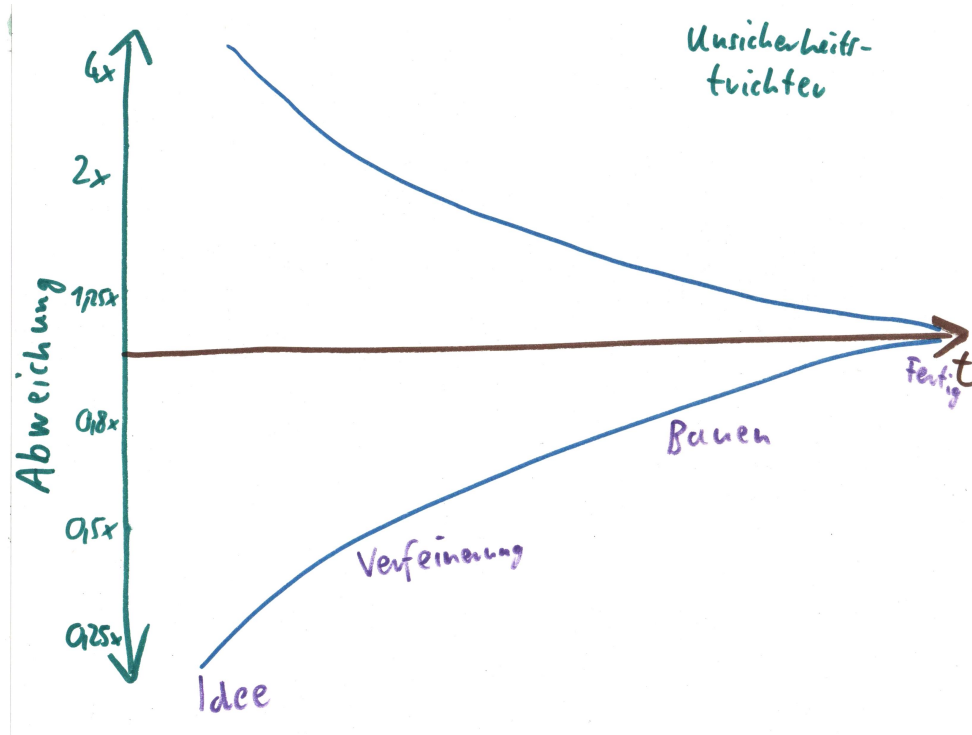


# Fehlerquellen

1. Unzureichende Informationen für die genaue Schätzung
2. Falsche Einschätzung der eigenen Fähigkeiten
3. Zuviel Veränderung im Projekt/in den Requirements
4. Schlechter Schätzprozess



# Unsicherheitstrichter







# Unsicherheitstrichter

- best-case!
- den Trichter schmaler bekommen ist kein Automatismus!
- Unsicherheitsfaktoren immer mitdenken
- Spannungsfeld: Commitment (zu) früh im Trichter abgeben



## Eigene Fähigkeiten

- Schätzung basiert z.B. auf "Seniors machen es"
- Umsetzung wird aber nicht nur von Seniors gemacht
- Abweichung zwischen Schätzung und echter Dauer - muss nicht schlimm sein, muss aber berücksichtigt werden (Plan!, Abrechnung)



# Zuviel Veränderung

- Unklare Requirements
- Sich ändernde Requirements
- NASA plant immer 40% dafür extra ein



# Wir haben da was vergessen... Schätzung

Nebenaufgaben werden oft vergessen:

- Setup bei jedem Entwickler
- Testdaten generieren
- Setup CI/CD
- Deployment auf dem Endsystem
- Onboarding
- Kommunikation mit dem Kunden
- Regelmäßige Meetings
- Änderungen nach Reviews
- ...



## Wir haben da was vergessen... Plan

Manche Dinge erhöhen den Aufwand nicht, aber die Projektdauer schon:

- Urlaub
- Krankheit
- Fortbildung
- Hardware-Probleme
- ...



# Schlechter Schätzprozess

- Unbegründeter Optimismus
- Druck von außen
- Zu viele Stellschrauben bei den Schätz-Faktoren
- Zu hohe (angebliche) Präzision
- Zu schnell abgeschätzt



# Lösungen





# Verfahren

- Zählen und Rechnen, Historische Daten
- Expertenschätzung, Gruppenreview
- Toolgestützt
- Formale Methoden





# Zählen und Rechnen

- man "zählt" etwas, das stark mit dem Aufwand korreliert (z.B. Anzahl an Formularen, Anzahl an anzubindenden Services, ...)
- man kennt den Aufwand pro Ding
- Gesamtaufwand:  $\text{Anzahl} * \text{Aufwand}$
- Ideal: Es gibt mindestens 20 davon, damit sich eventuelle Fehler rausmitteln können



# Historische Daten

- Vorbedingung für Zählen und Rechnen: Man hat Daten aus der Vergangenheit
- Vorbedingung für nach und nach besser werdende Schätzungen: Man hat Daten aus der Vergangenheit
- Je besser man alte Daten erfasst hat (aus dem gleichen Projekt und aus ähnlichen Projekten), desto mehr Möglichkeiten hat man
- Regelmäßiger Abgleich Schätzung und echte Ergebnisse, um Fehler und Fehlerquellen zu finden



# Expertenschätzung

- "hinsetzen, nachdenken, abschätzen"
- "Umsetzungsexperte" vs. "Schätzexperte"
- Best case, worst case, most likely case
- Tipp: sich Zeit nehmen ("1% Regel"), nicht sofort auf Zuruf Einschätzung abgeben



## Expected case berechnen

- aus Best case, worst case, most likely case für spätere Rechnungen den Expected case pro Paket berechnen:

$$ExpectedCase = \frac{BestCase + (3 * MostLikelyCase) + (2 * WorstCase)}{6}$$



## Bottom-up

- je feiner die Aufgliederung in Einzelpakete ist, desto weniger wird vergessen
- Checklisten benutzen
- je früher im Unsicherheitstrichter, desto größer sind die Pakete. Das muss sich auch in der Schätzung widerspiegeln



# Mathematische Tücken

- einfach nur die Best Case und Worst Case Werte addieren klappt nicht!
- Stattdessen ein bisschen Mathe:
- $StdDev_p = \frac{(WorstCase_p - BestCase_p)}{TeilerJeNachVertrauen}$
- Danach:  $Varianz_p = StdDev_p^2$
- Am Ende:  $StdDev = \sqrt{\sum Varianz_p}$



# Teiler je nach Vertrauen

"% wirklich in geschätzter Spannbreite"	Teiler in Standardabweichung
10	0,25
20	0,51
30	0,77
40	1,00
50	1,40
60	1,70
70	2,10
80	2,60
90	3,30
99,7	6,00



# Konfidenzintervalle

- Mit wieviel Wahrscheinlichkeit möchte ich unter dem Wert bleiben?
- Daraus bestimmt sich der Faktor für die Gesamt-Standardabweichung
- Die Einheiten müssen alle gleich sein (h, PT, Wochen, ...)





# Konfidenzintervalle

■  $\sum ExpectedCase + (Faktor * StdDev)$

Percentage confident	Faktor
10	-1,28
20	-0,84
30	-0,52
40	-0,25
50	0
60	0,25
70	0,52
80	0,84
90	1,28
98	2



# Geschafft

$$\begin{aligned}
 {}^1\sin\left(3t_2 + \frac{\pi}{6}\right) &= A \sin\left(3t_2 + \frac{\pi}{6}\right); \\
 &= \frac{1}{2}ky_2^2; E_0 = E - E_p = \frac{1}{2}k(A^2 - y_2^2) \\
 &= \frac{1}{2}k(A^2 - y_2^2) \Rightarrow y_2 = A \sqrt{\frac{V}{2}} = \frac{4}{3} \cdot 10^{-1} \text{ V} \\
 E_p &= E_{p_{\max}} \Rightarrow \sin^2\left(3t_p + \frac{\pi}{3}\right) = 1 \Rightarrow \sin \\
 &= \sin\left(\frac{\pi}{2} + n\pi\right); n = 0, 1, 2, \dots \\
 y) * z &= \left[\frac{1}{2}(x + y - xy + 1)\right] * z = \\
 &+ xy - xyz + z + 1 = \frac{1}{2}\left[\frac{1}{2}(x + y \right. \\
 y * z) &= x * \left[\frac{1}{2}(y + z - yz + 1)\right] = \\
 x(y + z - yz + 1 + 1) &= (x * y) * \\
 x * y &= \frac{1}{2}(x + y - xy + 1) \\
 &= \int_{-a}^0 x^2 e^{ax} dx = \frac{1}{a} \left[ x^2 e^{ax} \right]_{-a}^0 - \frac{2}{a} \int_{-a}^0 x e^{ax} dx \\
 &= -a^2 - \frac{2}{a} \left[ \frac{1}{a} (x e^{ax}) \right]_{-a}^0 - \frac{1}{a} \int_{-a}^0 e^{ax} dx \\
 &+ \frac{2}{a^2} \left[ \frac{1}{a} (e^{ax}) \right]_{-a}^0 = -ae^{-a^2} - \frac{2}{a} e^{-a^2} \\
 &= \frac{1}{a^3} e^{a^2} [2e^{a^2} - 2 - 2a^2 - a^4] \\
 Q_{\text{total}} &= Q_1 + Q_2 = 3\epsilon_0 \frac{S}{d_1} U_0 \\
 C_1 = C_2 &= \epsilon_0 \frac{S}{d_1} = 8,85 \text{ pF}
 \end{aligned}$$

In der Praxis: Tools benutzen (z.B. ein Excel-Sheet)



## Review in der Gruppe

- Umsetzungsexperten UND Schätzexperten
- erst jeder für sich, dann austauschen und besprechen
- Formaler Prozess: Wideband Delphi



# Wideband Delphi

0. Koordinator und Schätzexperten bestimmen
1. Koordinator verschickt die Spezifikation und ein Schätzformular
2. Meeting mit allen, um Unklarheiten in der Spezifikation zu besprechen
3. Jeder Experte füllt das Schätzformular **anonym** aus
4. Koordinator sammelt die Ergebnisse ein, führt sie zusammen und verteilt die Menge aller Schätzungen
5. Meeting mit allen, um extreme Abweichungen in den Schätzungen zu besprechen
6. Schritt 3-5 wiederholen, bis es keine extremen Abweichungen mehr gibt
7. Meeting mit allen, um die endgültige gemeinsame Abschätzung einstimmig anzunehmen



# Toolgestützt

- NDepend, JArchitect, ... für bestehende Software
- Simulation des Projektverlaufs
- Sanity Check von mit anderen Methoden erstellten Schätzungen
- Achtung: SHISHO ("Shit in, shit out")



# Formale Methoden

## COCOMO

- Constructive Cost Model
- Mathematische Formeln und viele (zu viele?) Rechenfaktoren
- Schwierig bei vielen Anforderungsänderungen

## Function Point Analyse

- ISO/IEC 20926
- Fokus auf fachlich-funktionale Anforderungen
- Achtung: Das Drumherum nicht vergessen



# Estimate Sanity Check

- Wurde ein definierter Prozess zur Erstellung genutzt?
- War die Erstellung frei von externem Druck/externer Einflussnahme?
- Wenn Sachen verhandelt wurde, ging es nur um Eingabedaten für die Schätzung?
- Entspricht die Präzision der Angabe der Projektphase?  
(Unsicherheitstrichter)
- Wurden unterschiedliche Schätztechniken eingesetzt, die zu ähnlichen Ergebnissen kamen?
- Ist die angenommene Produktivität der Projektbeteiligten aus der Vergangenheit begründbar?
- Ist die Dauer des Plans mindestens 2x der Aufwand?
- Waren die Leute, die später an der Umsetzung beteiligt sind, auch an der Schätzung beteiligt?
- Wurde die Schätzung nochmal von mindestens einem Experten geprüft?
- Enthält die Schätzung einen Puffer für Projektrisiken?
- Gehört die Schätzung zu einer Reihe von immer genaueren Schätzungen im Projektverlauf?
- Wurde an alle Aufgaben innerhalb des Projekts und innerhalb jedes Arbeitspakets gedacht?

## Wertung

- 10+ → super
- 7-9 → tauglich für die Projektsteuerung, wahrscheinlich zu optimistisch geschätzt
- 0-6 → in der Praxis nutzlos

*The original "Estimate Sanity Check" is from Software Estimation by Steve McConnell (Microsoft Press, 2006) and is copyrighted in 2006 by Steve McConnell. All Rights Reserved. Permission to copy this quiz is granted provided that this copyright notice is included.*



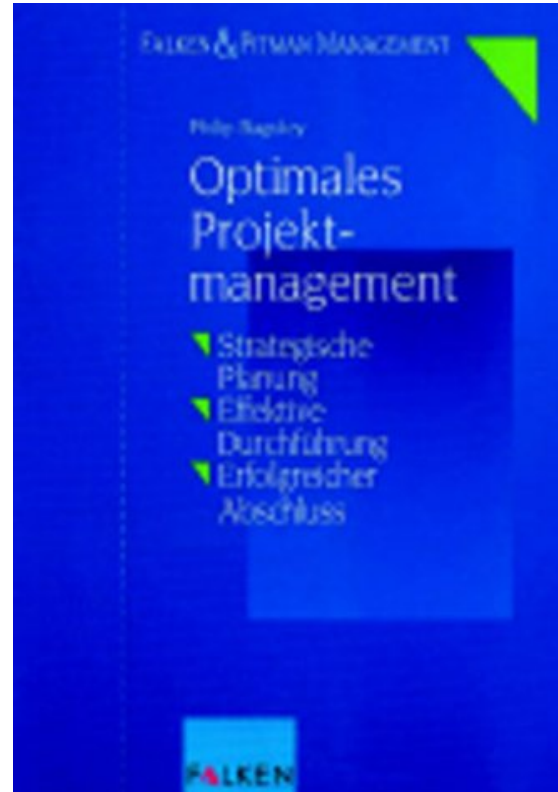
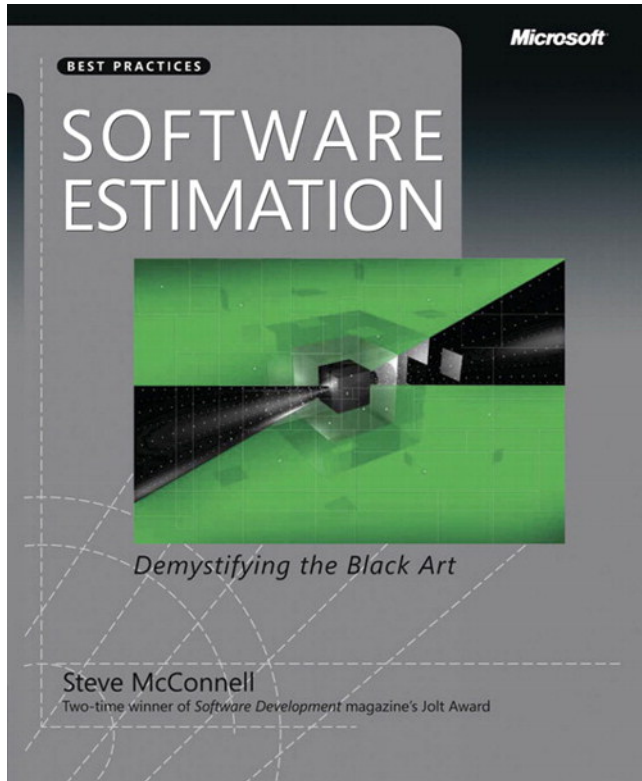
# Kommunikation

- keine falsche Präzision!
- Je unsicherer, desto größer muss die Bandbreite sein
- Abgleich mit dem Plan
- die unwahrscheinlichen Fälle weglassen





# Bücher





# Kontakt



E-Mail [martin.grotz@mathema.de](mailto:martin.grotz@mathema.de)

LinkedIn <https://www.linkedin.com/in/martin-grotz-6780332b5/>

MATHEMA  
Github [MATHEMA-GmbH](#)

Privates Github [groma84](#)