



# Mehr als »Pi mal Daumen« Formeln für Projektmanager

Im Alltagsleben eines Projektleiters lässt sich nur schwer mit festen Kennzahlen oder DIN-Werten arbeiten. Dennoch gibt es sie tatsächlich: Projektmanagement-Formeln, die sich in der Praxis sinnvoll anwenden lassen.



**Ralf C. Adam**

arbeitet als Executive Producer für Gameforge.

Ralf ist seit mehr als 17 Jahren in der Spielebranche tätig und hat in dieser Zeit als Producer und Projektleiter für Firmen wie Sunflowers, Infogrames/Atari, JoWood und Phenomic sowie als Freelancer für Disney, Ubisoft u.a. gearbeitet. Seit Ende 2008 betreut er mit seinem Producing-Team die externen Studios bei Gameforge.  
ralf.adam@gameforge.de

**WWW**  
makinggames.de  
• Projektmanagement mit Scrum

Die meisten Projektmanagement-Bücher sind vollgestopft mit Zahlen, Tabellen, Prozessschaubildern, wilden Abkürzungen sowie DIN-Vorschriften und -Normen. Jedoch zeigt sich nach der Lektüre solcher Fachliteratur meist schnell, dass sich das gedruckte Wissen in der harten Realität nur schwer umsetzen lässt: Projektmanagement besteht eben nicht nur aus Checklisten und Gantt-Diagrammen. Man hat es mit Menschen zu tun, und kein Projekt ist wie das andere (was im Übrigen per Definition ein Kriterium für ein Projekt ist). Das gilt in der klassischen IT-Branche, und es gilt um ein Vielfaches mehr in der Spieleindustrie, wo neben den ohnehin schon zahlreichen weichen, schwer kalkulierbaren Faktoren noch dieses unplanbare Element »Spielspaß« hinzukommt. »Wir machen eben keine SAP-Anwendungen für Banken« ist ein häufig gehörter Satz auf jeder Entwicklerkonferenz.

Dennoch gibt es sie tatsächlich: Praktische Anwendungstips und Formeln, die funktionieren und sich als zuverlässige »Silver Bullets« für den Patronengürtel des Projektleiters erwiesen haben. Einige davon soll dieser Artikel aufzeigen. Doch Achtung! Für alle nachfolgend aufgeführten Angaben, Formeln und Werte gilt der oberste Grundsatz im Projektmanagement: Gesunder Menschenverstand schlägt jede Regel!

## Dauer vs. Aufwand

Viele Projektmanager kennen das Problem: Wenn Teammitglieder gefragt werden, wie lange die Erledigung eines Arbeitspakets dauert, dann erhalten sie darauf zwar relativ präzise und wohlgedachte Zeitschätzungen, die aber am Ende dann trotzdem nicht eingehalten werden. Als Gründe kommen von den Mitarbei-

tern oft Aussagen wie »Ich musste ja noch in das Weekly-Meeting« oder »XY hat noch meine Hilfe benötigt«.

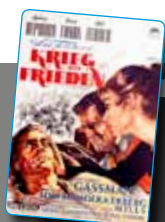
Das Problem liegt darin, dass viele Leute Aufwand und Dauer verwechseln bzw. bei einer Zeiteinschätzung eben nur die Zeit angeben, die es braucht, den Task tatsächlich umzusetzen. Wenn jedoch beispielsweise ein Coder sagt, er benötigt für die Implementierung eines Features fünf Tage, dann kann das eben durchaus heißen, dass er effektiv 40 Stunden daran programmieren muss. Da aber kein Mitarbeiter acht Stunden am Tag fünf Tage die Woche ausschließlich an seinen Arbeitspaketen arbeitet, stehen diese fünf Tage eben für den reinen Arbeitsaufwand, nicht aber für die eigentliche Dauer der Fertigstellung. Um in einem Projektplan trotzdem auf halbwegs verlässliche Zeiteinschätzungen für die einzelnen Arbeitspakete zu kommen, hat sich folgende Formel als nützliche Faustregel erwiesen:

$$\text{Dauer} = \text{Aufwand} + 30\%$$

Wenn ein Mitarbeiter also eine Zeiteinschätzung von fünf Tagen für einen Task angibt und man als Projektmanager davon ausgehen kann, dass es sich hierbei eher um den reinen Aufwand handelt, dann sollte man dem Arbeitspaket eine Dauer von 6,5 Tagen einräumen.

## Arbeiten mit Pufferzeiten

Nun sieht es auf den ersten Blick vielleicht verlockend aus, diese oben beschriebenen 30% Aufschlag einfach am Ende eines Milestones als Puffer zu reservieren, anstatt ihn auf die Dauer aufzuschlagen. Dies wäre jedoch grob fahrlässig: Puffer sollten immer für Bug-Fixing, Poli-



Der schnellste Speedrun in Half-Life 2 dauerte nur **1 Stunde und 36 Minuten**.  
Der Spielfilm Krieg und Frieden (1968) hat eine Laufzeit von über **8 Stunden**.



Die teuerste Konsole der Welt kostet **323.000 US-Dollar**: eine auf drei Exemplare limitierte, vergoldete und diamantbesetzte Playstation 3.



shing und Performance-Optimierung reserviert werden. Niemals aber für die Fertigstellung von Features, die es im eigentlichen Zeitrahmen nicht geschafft haben! Chris Williams, Director of Product Development bei Microsoft, sagt hierzu treffend: »Using up the buffer for feature work, and then saying »We made the milestone« is just lying to yourself! When will the bugs in those features be fixed?«

Wie viel Puffer aber braucht man in einem Projekt? Hier gibt es weniger verlässliche Formeln als vielmehr Erfahrungswerte. Bei einem vierwöchigen Sprint bzw. einer vierwöchigen Milestone-Phase sollte man als Entwickler mindestens eine Woche Puffer einplanen, ab sechs Wochen Arbeit empfehlen sich sogar zwei Wochen Puffer. Das bedeutet, dass der interne Abgabetermin scharf zwei Wochen vor dem externen Abgabetermin liegen und entsprechend geplant sein muss – kein Arbeitspaket und keine Feature-Implementierung darf im Puffer liegen!

Außerdem hat es sich als äußerst nützlich erwiesen, Puffer zu einem Milestone-Abgabetermin aufzuteilen. Da erfahrungsgemäß zu jeder Milestone- oder Sprint-Version Feedback aus der QA oder vom Publisher folgt, kann es durchaus sinnvoll sein, eine zweiwöchige Pufferphase so aufzuteilen, dass vor und nach Abgabetermin jeweils eine Woche Polishing- und Bugfixing-Zeit reserviert wird, anstatt die zehn Arbeitstage komplett vor dem Milestone zu verbrauchen. Andernfalls müssen nämlich aus der Milestone-Abnahme resultierende Change Requests im nächsten Sprint mit aufgefangen werden, was in der Zeitplanung meist nicht berücksichtigt ist.

### Die Größe von Arbeitspaketen

Eine bei der Erstellung des Projektplans immer wieder auftauchende Frage lautet: »Wie groß oder klein darf ein Arbeitspaket eigentlich sein?« Ein Anwalt würde hierauf wahrscheinlich antworten: »Das kommt darauf an.« Und tatsächlich kann die Gesamtprojektgröße hier durchaus eine gewisse Rolle spielen. Meine Erfahrung ist in diesem Falle allerdings, dass – egal ob es sich um ein Drei-Monats- oder Drei-Jahres-Projekt handelt – ein Task niemals länger als fünf Tage dauern und idealerweise nicht kleiner als ein Tag sein sollte.

Die erste Zahl erklärt sich aus der reinen Definition eines Tasks: Ein Arbeitspaket ist ein Teil eines Projekts, der nicht mehr weiter sinnvoll aufgegliedert werden kann. Arbeitspakete, die größer als eine Woche sind, lassen sich eigentlich immer sinnvoll weiter herunterbrechen. Außerdem wird bei großen Arbeitspaketen die

### Tip: Unfertige Tasks aufteilen

Oft kommt es vor, dass ein Task von einem Mitarbeiter angefangen, aber erst zu einem späteren Zeitpunkt beendet werden kann. Das führt schnell dazu, dass im Projektplan viele Arbeitspakete auf »50% abgeschlossen« gesetzt sind und damit quasi in der Luft hängen. Der Projektplan lässt sich immer schwerer überschauen, und halbfertige Tasks wandern jede Woche weiter nach hinten, bis sie schließlich ganz in Vergessenheit geraten, obwohl sie noch nicht fertig gestellt wurden.

Aus diesem Grund empfiehlt es sich, Tasks aufzusplitten, sobald ein Mitarbeiter die Arbeit daran zu einem bestimmten Zeitpunkt nicht abschließen kann und sich einem anderen Arbeitspaket zuwenden muss. Anstatt also z. B. einen 4-Tages-Task über Wochen als »50% abgeschlossen«-Leiche mit sich herumzuschleppen, sollte man den Task in zwei 2-Tages-Pakete aufteilen, eins davon auf 100% setzen und das andere beispielsweise durch Priorisierung im Projektplan nach hinten schieben.



Aus eins mach zwei: Teilen Sie den Task (hier in MS-Project) in zwei Sub-Tasks auf, setzen sie den ersten auf »100% abgeschlossen« und geben Sie dem zweiten eine entsprechend niedrige Priorität.

Projektplanung unscharf und das Tracking enorm schwierig. Wenn ich als Projektleiter erst nach 20 Tagen erfahre, dass ein Arbeitspaket noch nicht fertig gestellt wurde, ist das Kind meist schon in den Brunnen gefallen und der Milestone gerissen.

Umgekehrt machen 0,25-Tages-Tasks das Tracking ebenfalls unnötig kompliziert, immerhin reden wir hier von zwei Stunden Dauer. Und da – wie oben beschrieben – Dauer nicht dem Aufwand entspricht, reden wir am Ende von Arbeitspaketen auf Mikroebene, die in ihrer Granularität schon fast eher detaillierten Arbeitsanweisungen entsprechen.

### Dreifach-Schätzung

Eine im Projektmanagement sehr beliebte Methode, bei der Aufwandseinschätzung von Arbeitspaketen noch verlässlichere Zeitangaben zu bekommen, besteht in der sogenannten Dreifach-Schätzung: Der Mitarbeiter überlegt sich zu jedem von ihm zu erfüllenden Task, wie lange es im Normalfall dauern wird, das Arbeitspaket zu erledigen (Most Likely). Darüber hinaus schätzt er aber außerdem noch den Wert, wie lange es im besten Fall dauern könnte (Best Case) sowie im nicht eben unwahrscheinlichen Fall, dass alles, was nur schiefgehen kann, auch schiefgehen wird – frei nach dem berühmten Murphy's Law (Worst Case).

Viele Projektmanager ermitteln dann aus dieser Dreifach-Schätzung den einfachen Mittel-

Apple konnte zwischen 1995 und 1997 insgesamt nur **42.000** Pippin-Konsolen verkaufen. Nintendo verkauft aktuell

jede Woche **328.000** DS-Handhelds.

**80 Prozent** der spielenden Frauen machen das auf der Wii. Die Wii-Quote bei den männlichen Gamern beträgt **40 Prozent**.

Der SNK Neo-Geo AES und der Atari 2600 hatten mit jeweils **13 Jahren** die bisher längsten Konsolenlebenszyklen.

**Tipp: Die 5 Ws bei der Planung**

Eine Projektplanung ist nur so gut wie die darin enthaltenen Zeitschätzungen. Um hier auf möglichst valide und verlässliche Angaben zu kommen empfiehlt es sich, bei der Schätzung für jeden Task folgende fünf Fragen zu stellen:

- Wer (Ausführender) macht...
- ...was (Aufgabe, Arbeitspaket)
- ...bis wann (Abgabetermin/Dauer)
- ...mit welchem wie gemessenen Ergebnis (Ziel- und Erfolgskriterien)
- ... wozu (wird diese Aufgabe später gebraucht)

wert. Tatsächlich haben jedoch statistische Untersuchungen gezeigt, dass eine 87,5%ige Chance besteht, dass sich die tatsächliche Projektdauer von der durchschnittlichen Schätzung hin zur Worst-Case-Schätzung verschiebt. Aus diesem Grund empfiehlt es sich, auf eine der beiden nachfolgenden Formeln zurückzugreifen, die beide das oben beschriebene Phänomen berücksichtigen und annähernd ähnliche Werte liefern (siehe auch Extra-Kasten):

**Variante 1**  
 (aus »Die nackte Wahrheit über Projektmanagement«/Wilfried Reiter):  
**Worst-Case-Schätzung minus Mittelwert geteilt durch 3.**

**Variante 2**  
 (aus »Game Producer's Handbook«/Dan Irish):  
**2 mal Best-Case-Schätzung plus 3 mal Worst-Case-Schätzung plus Most-Likley geteilt durch 6.**

**Dreifach-Schätzung (Beispiele)**

**Variante 1**  
 Formel:  
 Worst Case – Mittelwert / 3  
 Beispiel:  
 • Best-Case: 4 Tage, Most-Likely: 10 Tage, Worst-Case: 16 Tage  
 • Mittelwert = 10 Tage  
 • Worst-Case 16 – Mittelwert 10 = 6  
 •  $6 / 3 = 2$   
 Hohe Wahrscheinlichkeit, dass die Projektdauer um 2 Tage von der Most-Likely-Schätzung nach oben hin abweicht = 12 Tage.

**Variante 2**  
 Formel:  
 $(2 \times \text{Best-Case} + 3 \times \text{Worst-Case} + \text{Most-Likely}) / 6$   
 Beispiel:  
 • Best-Case: 10 Tage, Most-Likley: 15 Tage, Worst-Case: 25 Tage  
 •  $(2 \times 10 + 3 \times 25 + 15) / 6 = 18,33$   
 Hohe Wahrscheinlichkeit, dass die Projektdauer um 3,33 Tage von der Most-Likely-Schätzung nach oben hin abweicht = 18,33 Tage.

Die Dreifach-Schätzung hat noch einen weiteren praktischen Vorteil: Dadurch, dass der Mitarbeiter gezwungen ist, die verschiedenen Szenarien genauer zu durchdenken, wird schon seine Most-Likely-Schätzung viel präziser, weil er für die verschiedenen Szenarien automatisch viel mehr Parameter berücksichtigt als bei einer einfachen »Wie lange wird dieser Task wohl dauern?«-Schätzung aus der hohlen Hand.

Bei längeren Projekten kann es außerdem hilfreich sein, nach jedem Milestone die abgegebenen Schätzungen jedes Mitarbeiters für die zurückliegenden Wochen zu evaluieren und die oben aufgeführten Formeln individuell anzupassen. Manche Teammitglieder tendieren beispielsweise dazu, gerade bei der Worst-Case-Schätzung übervorsichtig zu sein, während sie bei anderen fast schon automatisch die wahrscheinlichste Zeitangabe ist. Durch diese Art der Feinjustierung lassen sich die Zeitschätzungen im Projektplan über den Projektgesamtverlauf hin immer genauer schärfen.

**Der PI-Faktor**

Ein weiteres nützliches Tool, auf das eigentlich kein gewissenhafter Projektleiter verzichten kann, ist eine Risiko-Liste. In der werden alle aktuell im Projekt erkannten Risiken inklusive Maßnahmen zur Gegensteuerung, Action-Points im Falle eines Eintretens des Risikos sowie eines Verantwortlichen pro Risiko erfasst und verwaltet. Die Priorisierung dieser Risiken erfolgt allerdings oftmals nach keinem bestimmten System oder Muster. In manchen Risiko-Listen werden die Risiken chronologisch in der Reihenfolge ihres Sichtbarwerdens aufgeführt, in anderen werden gefühlte Aus-dem-Bauch-heraus-Prioritäten vergeben.

Eine genauere Methode, um Risiken gezielt zu priorisieren, bietet der sogenannte PI-Faktor. »P« steht in diesem Fall für »Probability« (Wahrscheinlichkeit: Wie hoch ist die Chance, dass das Risiko eintritt?) und »I« für »Impact« (Auswirkung: Wie gravierend sind die Auswirkungen auf das Projekt, falls das Risiko eintritt?). Jedem Risiko werden nun für diese beiden Faktoren zunächst Werte zwischen 0% und 100% zugewiesen (in 20%-Intervallen). Anschließend wird nach folgender Formel der PI-Faktor bestimmt:

$(P \times I) \times 0,01$

Mit den auf diese Weise erhaltenen Werten lassen sich die Risiken sinnvoll nach ihrer Priorität sortieren. Hierbei empfiehlt es sich, eine eigene PI-Matrix anzulegen und individuell für das Pro-

Legend of Zelda: Ocarina of Time für den Nintendo 64 ist mit **99 von 100 Punkten** das am besten bewertete Spiel aller Zeiten auf metacritic.com. Das schlechteste Spiel ist Big Rigs: Over the Road Racing (2003) mit **8 Punkten**.



Mario, Nintendos italienischer Klempner, taucht in über **200 Spielen** auf. Er ist nach dem früheren Vermieter des Nintendo-Hauptquartiers in den USA benannt: Mario Seagle.



DRAFT – RISK MANAGEMENT PLAN					
PROBABILITY (P)	IMPACT (I)				
	20%	40%	60%	80%	100%
20%	4%	8%	12%	16%	20%
40%	8%	16%	24%	32%	40%
60%	12%	24%	36%	48%	60%
80%	16%	32%	48%	64%	80%
100%	20%	40%	60%	80%	100%

Risk	Notes	Owner	(P) Factor	(I) Factor	(PI) Factor
Das externe, ausländische Grafikstudio stellt nicht, wie versprochen, sein A-Team und liefert minderwertige Qualität.					
Der Name für das Spiel steht noch nicht final und es muss noch Marken-Recherche betrieben werden. Ohne finalen Namen, kein finales Logo, keine URL etc.					
Durch schlechte bzw. fehlende Art-Content-Verwaltung könnten Projektverzögerungen entstehen.					
Rückfragen an das Game Design können nicht zeitnah beantwortet werden, weil einer der beiden Game Designer externer Freelancer ist.					
Coder könnten aus dem Projekt abgezogen und auf laufende Spiele im Betrieb geworfen werden, falls dort aktueller Bedarf besteht.					
Feature Cut führt zu Balancing-Problemen; das Elder-Gameplay im Spiel reicht nicht aus – das Spiel könnte zu schnell zu Ende sein.					
Schlechtes Missions- und Level Design verwehrt den leichten Zugang zum Spiel, ebenso leidet der Spielspaß.					
Datenaufwand könnte zu Performance-Problemen führen.					
Umsetzung der kontextsensitiven Dialogboxen könnte sich als problematisch erweisen.					
Wir haben noch keinen UI-Designer im Team.					

Erstellen Sie sich Ihre eigene Risk-Matrix und gewichten Sie mithilfe des PI-Faktors für Ihr Projekt individuell die Bedrohung, die von jedem Risiko ausgeht.

jekt zu bestimmen, welcher PI-Level welche Dimension an Bedrohung darstellt, zum Beispiel über einen Ampel-Farbcode.

### Ein Wort zum Schluss

Allen hier aufgeführten Tipps ist gemein, dass sie sich mehr oder weniger unabhängig von der eingesetzten Projektmanagement-Methode anwenden lassen. Egal ob in einem agilen Prozess wie Scrum oder im guten alten Wasserfall-Modell: Projektplanung besteht im Kern immer darin, sich zu fragen, wer im Team welches Arbeitspaket mit welchem zu erzielenden Ergebnis erledigen muss, wie lange dies dauert (bzw. wie komplex diese Aufgabe ist) und wie die Abhängigkeiten und Prioritäten sind (siehe Extrakasten). Und da – wie bereits zu Beginn des Artikels erwähnt – jedes Projektteam aus völlig unterschiedlichen Individuen besteht und anders

»tickt«, möchte ich Sie ermutigen, alles hier Geschriebene zu hinterfragen, nach Belieben zu adaptieren und anzupassen und Ihre ganz eigenen, persönlichen Projektmanagement-Methoden zu entwickeln. Denn Projektmanagement ist ein weites Feld ...

Ralf C. Adam

Bei Tetris auf dem Gamboy kann der Spieler maximal **Level 20** erreichen: Hier fällt alle zwei Frames ein Puzzlesteinchen – mehr schafft der **8 KB** Arbeitsspeicher nicht.



Die Unterhaltungssoftware Selbstkontrolle (USK) hat im Jahr 2009 **3.100 Spiele** geprüft. **92 Prozent** der Titel wurden mit »Ab 16« oder niedriger freigegeben.



Assassin's Creed 2 hat **87.206 Zeilen** vertonte Dialoge. Zum Vergleich: Die Bibel hat **31.171 Verse**.

