

# Typische Projektlügen, Teil 1

## Warum Projektplanung scheitert

Jeder, der in der Spiele-Entwicklung schon einmal mit Projektplanung zu tun hatte, weiß: Das funktioniert eh nicht! Doch über die Gründe, warum »es« so oft nicht funktioniert, sind sich die Wenigsten im Klaren.



Ralf C. Adam

arbeitet als freier Producer, Autor und Game Design Consultant.

Ralf ist seit 14 Jahren in der Spielebranche tätig und hat in dieser Zeit als Producer und Projektleiter für Firmen wie Sunflowers, Infogrames/Atari, Jowood und Phenomic gearbeitet. Seit nunmehr drei Jahren bietet er seine Dienste als »Gun for hire« an, und berät dabei sowohl Entwicklerteams auf der einen, als auch Publisher auf der anderen Seite – aktuell unter anderem Atari, Blue Byte, dtp und GameForge. [ralf@tigerteam-productions.de](mailto:ralf@tigerteam-productions.de)

**A**ls die Standish Group 1994 zum ersten Mal die im so genannten CHAOS-Report zusammengefasste Studie zu Projektmanagement durchführte, kam sie zu dem Ergebnis, dass in der IT-Branche nur 16,2 Prozent aller Projekte »on-time« und »on-budget«, also pünktlich und innerhalb des vereinbarten Kostenrahmens entwickelt werden. 31,1 Prozent aller Projekte kommen zu spät und/oder laufen finanziell aus dem Ruder. Und volle 52,7 Prozent aller Projekte werden noch vor Fertigstellung eingestampft. In der Spiele-Entwicklung dürften diese Zahlen sogar noch wesentlich dramatischer ausfallen. Immerhin verstehen unsere Kollegen von SAP und Co unter kleinen Projekten solche, die mit fünf Teammitgliedern und einer Laufzeit von weniger als drei Monaten auskommen. An mittleren Projekten arbeiten sechs bis 30 Teammitglieder bis zu einem Jahr. Und alles darüber hinaus fällt schon unter den Begriff Großprojekt. Wir haben es im Games-Bereich inzwischen also fast ausschließlich mit solchen Mammut-Entwicklungen zu tun. Und die haben – wie der DFB-Pokal – ihre ganz eigenen Gesetze.

### Alles Lüge

Jedes Teammitglied, egal ob Projektleiter, Lead- oder Aushilfstester, sieht sich bei der Spiele-Entwicklung tagtäglich mit Aussagen konfrontiert, die sich im Nachhinein als glatte Lügen herausstellen. Der Publisher verspricht dem Studioleiter bei Vertragsunterschrift vollen Support – und lässt ihn bei der ersten Milestone-Zahlung im Regen stehen. Der Coder verspricht dem Technical Director diesen einen wichtigen Task, von dem alles abhängt, bis Freitag fertig gestellt zu haben – und zwei Wochen später wartet der Rest des Teams immer noch darauf und kann nicht weiterarbeiten. Das Marketing verspricht, dass sie nur noch diesen einen hoch aufgelösten Freisteller des Orks für die GC benötigen – und zwei Tage vor Messebeginn rauscht eine Anforderungsliste über 200 Screenshots, zehn zusätzliche Hi-Res-Artworks und einen Video-Trailer herein (und das Ganze natürlich bis gestern).

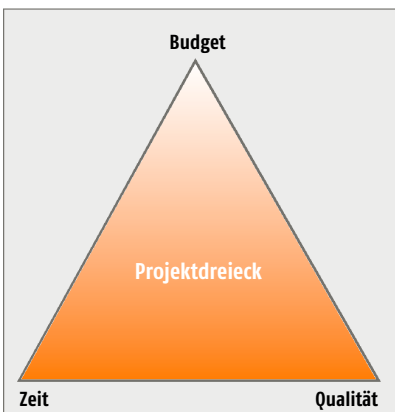
Irgendwann verzweifelt auch der abgebrühteste Projektleiter und nimmt dieses Verhalten

als gegeben hin. Der Gewöhnungseffekt führt schlussendlich dazu, dass all diese Unwahrheiten zu Wahrheiten werden, die man einfach akzeptieren muss. Im nächsten Schritt werden dann Schuldige oder zumindest Gründe für das unvermeidliche Scheitern gesucht. Zu den beliebtesten Ausflüchten zählen dabei: Der Projektmanager/Producer war zu unerfahren, die technischen Probleme haben uns zurückgeworfen, oder – in unserer Branche immer noch die gängigste Verteidigungsmethode – das Budget, die Zeit und die Ressourcen, die uns zur Verfügung standen, waren viel zu knapp bemessen (meist mit dem Zusatz: »Wenn wir das Budget von Blizzard hätten, ja dann! Dann könnten wir...«). Tatsache ist aber: Auch Projekte mit 20, 40 oder 200 Millionen Dollar Budget brauchen doppelt so lange oder kosten doppelt so viel wie zunächst geplant – und scheitern genauso oft wie kleinere (meist dann nur etwas lauter).

### Projektmanagement ist Zeitverschwendung!?

Eine Aussage, die man immer wieder von Projektleitern in der Spiele-Entwicklung hört, lautet: »Ich hab's ja versucht, aber es klappt nicht«. Der in mühevoller, monatelanger Arbeit ausgefüllte Projektplan wird beim ersten Teammeeting von den Teammitgliedern als »bürokratischer Schwachsinn« in der Luft zerrissen. Schließlich sei eine Spiele-Entwicklung ein »organischer Prozess« und »so ein Plan kostet nur unnötig Zeit«. Und mit der Realität habe das alles sowieso schon mal gar nichts zu tun. Nach zwei bis drei Wochen, in denen der pflichtbewusste Projektleiter noch versucht, den Tasks nachzulaufen, dabei aber vom gesamten Team boykottiert wird, findet keine Aktualisierung des Projektplans mehr statt. Spätestens nach einem Monat verschwindet er dann auf Nimmerwiedersehen ganz in der Schublade – es sei denn, man muss ihn noch halbherzig pflegen (oder zumindest so tun, als ob), da der »nervige« Publisher darauf besteht.

Auch die anderen Versuche, mehr Struktur und organisiertes Vorgehen zu etablieren, lässt das Team gnadenlos gegen die Wand fahren. Die



Das »magische« Projektdreieck – die meisten Entwickler sind schon froh, wenn sie auch nur eines der drei Ziele halbwegs treffen.

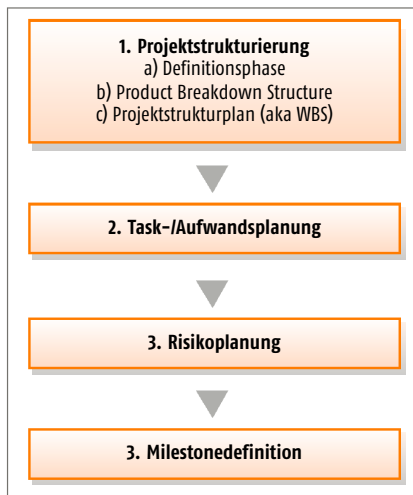
aus der sündhaft teuren, auf der letzten GDC erstandenen Projektmanagement-Bibel herauskopierten Prozessformulare ernten nur schallendes Gelächter beim Team. Und das zum ersten (und letzten) Mal abgehaltene Projekt-Kick-Off-Meeting endet mit einem allgemeinen Genöle über diese »Selbstbeweihräucherungsveranstaltung«. Die Lehren, die der Projektleiter daraus zieht, sind eindeutig: Projektplanung ist Zeitverschwendung, killt die Motivation im Team, und alles, was in Büchern über Projektmanagement steht, ist in der Praxis sowieso nicht anwendbar. Eine nachvollziehbare Reaktion – aber leider mit den falschen Rückschlüssen.

### Es zählt nicht die Methode, sondern die Adaption

Die erste und vielleicht wichtigste Regel überhaupt, wenn es um Projektmanagement geht, lautet: Was sagt mir mein gesunder Menschenverstand? Jeder Projektverantwortliche, der versucht, eine von irgendjemand erlernte oder irgendwoher abgekupferte Vorgehensweise unreflektiert über sein eigenes Team zu »stülpen«, kann damit nur scheitern. Der Trick liegt nicht in der Methode, sondern in der Anpassung! Jedes Team ist anders – und jedes Spiel ist anders (und damit auch der Weg, dieses Spiel zu entwickeln). Das liegt in der Natur der Sache, denn ein wesentliches Unterscheidungsmerkmal eines Projekts im Gegensatz zu einem Prozess besteht darin, dass es sich bei einem Projekt immer um eine Arbeit handelt, bei der etwas komplett Neues entsteht. Beispiel: Einen neuen Porsche zu entwickeln ist ein Projekt, tausende davon in einer Fabrik zu bauen dagegen nicht. Und der besondere Faktor des Spielspaßes unterscheidet unsere Branche noch zusätzlich von jeder anderen Form der Projektentwicklung. Somit sieht sich ein Game-Developer bei jedem neuen Spiel mit der alten Star-Trek-Herausforderung konfrontiert: »To boldly go where no one has gone before«.

Damit wären wir auch schon bei der zweiten Regel, die von niemand Geringerem als Albert Einstein stammt – und unter anderem auch in Noah Falsteins und Hal Barwoods »400 Project« ([www.theinspiracy.com](http://www.theinspiracy.com)) als Game-Design-Regel zitiert wird: »Mache die Dinge so einfach wie möglich, aber nicht einfacher!« Bei jeder Technik, die ein Projektleiter einsetzen möchte, sollte er zunächst hinterfragen, ob sie die Ergebnisse liefert, die er benötigt – und ob es nicht auch ohne sie gehen würde. Wer für ein Drei-Monatsprojekt mit zwei Mitarbeitern einen Projektplan mit Tasks auf Minutenbasis herunterbricht, wird damit höchstwahrscheinlich genauso sein Ziel verfehlen, wie jemand, der für eine 30-Monatsentwicklung mit einem 50-Mann-Team nur 100 Tasks im Projektplan aufführt.

Ein weiterer, grundlegender Fehler, der in den oben aufgeführten Beispielen gemacht wurde (und auch in der Praxis regelmäßig gemacht wird), besteht in der Nichteinbeziehung des restlichen Teams. Wer sich als Projektleiter in seinen sprichwörtlichen Elfenbeinturm zurückzieht



Projektplanung kann nur funktionieren, wenn sie in der richtigen Reihenfolge angegangen wird.

und sich dort (womöglich auch noch ungefragt) vermeintlich tolle Prozesse und Workflows ausdenkt, muss sich nicht wundern, dass diese hinterher im Team keinen Anklang finden und schlicht ignoriert werden. Actio gleich Reactio – oder, wie man bei uns sagt: Wie man in den Wald hineinruft, so schallt es heraus. Anstatt also in Eigenregie ein tolles Regelwerk darüber aufzustellen, wie »der Hase in Zukunft bei uns zu laufen hat«, sollte man zunächst mal seine Teammitglieder fragen, wo sie denn die größten Probleme und Reibungsverluste in der aktuellen Arbeitsweise sehen. Hierfür eignet sich dann in der Tat hervorragend das viel beschworene Kick-Off-Meeting. Aber eben nicht, um die versammelte Mannschaft vor vollendete Tatsachen zu stellen, sondern um gemeinsam darüber zu diskutieren, wie man das nächste Projekt sinnvoller gestaltet. Wie weit sollte man Tasks granulieren? Welche Tools sollte man für welchen Work-



Pünktlich zu Weihnachten und in hervorragender Qualität: Anno 1701 von Sunflowers und Related Designs gilt zu Recht als Musterbeispiel für gute Projektplanung.

flow einsetzen? Wie werden Verzögerungen im Projekt gehandhabt? Wie wird miteinander kommuniziert? Wie oft finden wann warum welche (Team-)Meetings statt? Wer wüsste die Antworten hierauf besser, als die eigenen Mitarbeiter – schließlich schlagen die sich tagtäglich mit diesen Fragen herum.

Viele Projektleiter schrecken vor dieser Form der Involvierung des gesamten Teams zurück, da sie befürchten, dass am Ende nicht unbedingt die effizienteste Methode steht – und dies kann auch durchaus passieren. Hier gilt dann im Zweifel aber: Akzeptanz kommt vor Methode! Denn was nutzt die schönste Methode, wenn sie vom Team nicht akzeptiert und damit auch automatisch nicht (oder nur halbherzig und damit ineffizient) umgesetzt wird?

**Die Sache mit den Milestones ...**

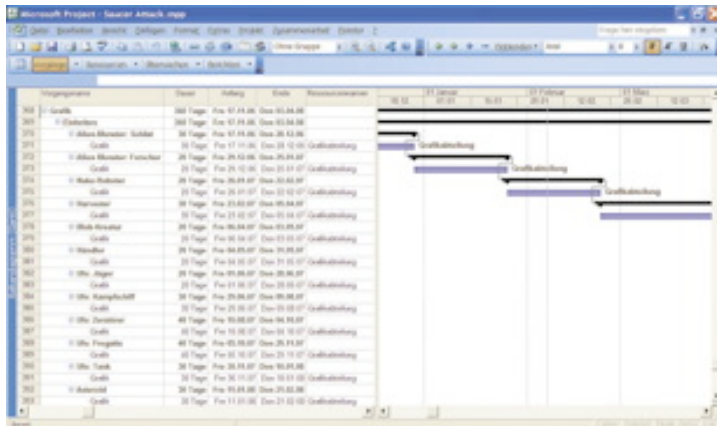
Fast jeder Spiele-Entwickler – es sei denn, er genießt den seltenen Luxus, sein Spiel aus eigener Tasche finanzieren zu können – ist auf Gedeih und Verderb von der pünktlichen Lieferung seiner Milestones abhängig, da hieran die Zahlungen des Publishers (und damit das eigene Überleben) geknüpft sind. Aus diesem Umstand entsteht allerdings schnell eine verhängnisvolle Spirale aus Lügen und Selbstbetrug – getreu dem

Motto, dass nicht sein kann, was nicht sein darf. Ein Spielchen bei dem, so muss man fairerweise hinzufügen, nicht nur der Developer, sondern meist auch der Publisher selbst nur allzu gerne mitmacht. Der fatale Kreislauf beginnt fast immer vermeintlich harmlos: Der erste Milestone hat einen leichten Verzug und kommt nicht pünktlich. Anstatt nun die Situation zu analysieren, die Planung zu überprüfen und gegebenenfalls sogar über die Machbarkeit des geplanten Release-Datums nachzudenken, beginnt stattdessen ein sich gegenseitiges Beruhigen und Herunterspielen der Probleme: Die Verzögerung hat ja noch nichts zu bedeuten, so früh im Projekt, man liegt ja noch voll im Soll, die Milestones sind sowieso mehr grobe Richtwerte und im Zweifel wirft man später einfach mehr Ressourcen auf das Projekt, um den Verzug wieder einzuholen. Tatsächlich ist eigentlich schon zu diesem frühen Zeitpunkt klar, dass das Spiel niemals zum ursprünglich angestrebten Datum veröffentlicht werden kann. Und alle Beteiligten wissen das auch. Nur traut niemand sich es offen auszusprechen. Zumal im weiteren Verlauf des Projekts der Verzug zunächst geschickt verschleiert wird: Die nächsten Milestones kommen plötzlich wieder pünktlich – und damit ist die Entwicklung automatisch wieder »on track«. Oder man redet es sich zumindest ein. Dass Milestone 2 lediglich aus Marketing-Assets besteht, die nichts über den eigentlichen Entwicklungsstand aussagen, wird geflissentlich übersehen. Das böse Erwachen erfolgt dann meist erst gegen Projektende zur Alpha oder Beta, wenn es bereits zu spät ist, noch wirkungsvolle Maßnahmen zu ergreifen. Wo aber liegen die tiefliegenden Wurzeln dieses Übels?

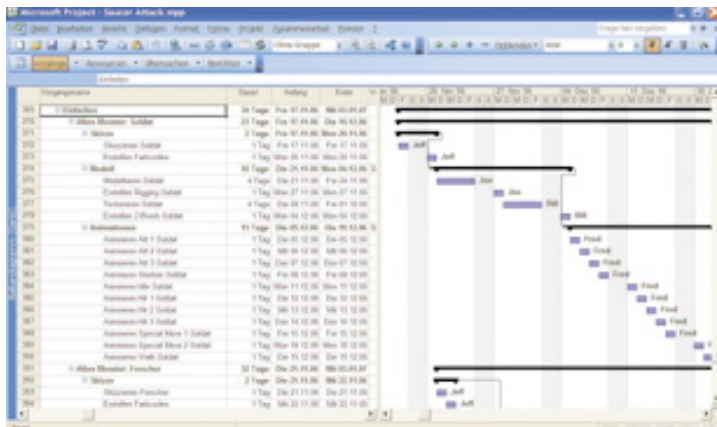
**Wir lügen uns einen Termin zurecht**

Fast jeder, der schon einmal einen Vortrag über Projektmanagement besucht hat, kennt das berühmte Projektdreieck (siehe Abbildung) mit den drei Eckpunkten Qualität, Zeit/Termine, Kosten/Budget. Jeder Publisher besteht natürlich darauf, dass das von ihm in Auftrag gegebene Spiel in allen drei Punkten zu seiner Zufriedenheit fertig gestellt wird, sprich in den fast gebetsmühlenartig beschworenen Faktoren »on Time«, »on Quality«, »on Budget«. Hierzu fällt mir ein Spruch aus der Werbung ein: »Das sind ja drei Wünsche auf einmal – das geht nun wirklich nicht!« Und tatsächlich zeigt die Erfahrung, dass jeder, der auf Teufel-komm-raus versucht, alle drei Ziele zu erfüllen, in mindestens zweien versagt (in der Spiele-Entwicklung in der Regel krasse Überziehung der Zeit- sowie der Budgetvorgaben).

Der logische Umkehrschluss daraus lautet: Statt zu versuchen, alle drei einzuhalten, sollte man sich von vornherein (gemeinsam mit dem Publisher) auf einen Punkt einigen, bei dem man bereit ist, Kompromisse einzugehen. Dieser Punkt muss dann natürlich entsprechend besonders genau gesteuert und gemanagt werden. Bei obigem Beispiel wäre es zum Beispiel sinnvoll



Ein Projektplan des Grauens: Tasks, die 30 Tage und länger dauern und weder in Einzelaufgaben runtergebrochen noch auf individuelle Ressourcen verteilt wurden.



Derselbe Plan, allerdings wesentlich feiner granuliert und unter Berücksichtigung der entsprechenden Abhängigkeiten sowie der Reihenfolge der Abarbeitung.



gewesen, schon zu diesem frühen Zeitpunkt über das Streichen von Features (= Qualität) nachzudenken oder die Zeiten – und damit das Masterdatum – entsprechend anzupassen. Nur leider traut sich in frühen Projektphasen meist niemand, diese heiligen Kühe anzurühren. Aber: Features herauszunehmen, wenn sie schon zu 90 Prozent implementiert sind, führt nicht zu Zeitgewinn, sondern im Gegenteil mit hoher Wahrscheinlichkeit zu noch mehr Chaos – und Verzögerung.

Da aber besonders in unserer Branche in den allermeisten Fällen am ehesten die Zeitvorgaben nicht eingehalten werden, begehen gerade die Publisher den Denkfehler, dass dieser Punkt die meisten Vorgaben benötigt und setzen Termine und Milestones, noch ehe das Team überhaupt mit seiner Planung begonnen hat. Meiner Erfahrung nach ist dies eines der krassesten Projektvergehen überhaupt! Es führt zwangsläufig und automatisch dazu, dass der Developer – schließlich ist er auf die Milestones und damit die Zahlungen angewiesen – seine Planung auf die Vorgaben hin »zurechtbiegt«. Wenn ein Programmierer also einen Task mit zehn Tagen eingeschätzt hat, dies aber nicht in die Gesamtplanung zur Erfüllung von Milestone X passt, wird der Projektleiter hingehen und ihn fragen, ob er es nicht auch in fünf Tagen schaffen könnte. Und ich möchte den Programmierer sehen, der sich bei einer solchen Fragestellung nicht bei seiner Ehre gepackt fühlt und in bester Scotty-Manner antwortet wird: »Ich schaffe es auch in zwei, Captain!« Leider werden es dann am Ende erfahrungsgemäß doch eher 15.

### Wie treffe ich ein bewegliches Ziel?

Eine Planung sollte deshalb niemals und unter keinen Umständen auf Basis von bestehenden Terminen geschehen, sondern immer auf Basis von Ressourcen und Qualität. Am Anfang stehen also nicht die Milestones, am Anfang steht immer(!) die Definition des angestrebten Ergebnisses, in aller Regel also das komplett ausgearbeitete Game-Design, inklusive (und das wird oft und gerne vergessen) einer Definition der Qualitätskriterien. Dann erfolgt die Zeiteinschätzung der einzelnen Tasks und danach die Planung des Ablaufs anhand der vorhandenen Ressourcen. Hieraus erst ergeben sich die Termine wie Milestones, Alpha, Beta etc. Wer das Pferd vom Schwanz her aufzäumt, kann sich eine Projektplanung gleich schenken.

Da wir aber in keiner idealen Welt leben, gibt es natürlich Vorgaben an ein Projekt – und diese sind meist zwangsläufig terminlicher Natur (wenn das Spiel zu Weihnachten 2008 nicht kommen kann, dann wird der Publisher gar nicht erst den Vertrag unterschreiben). Nachdem man also in der oben beschriebenen Reihenfolge eine realistische Planung erstellt hat, sollte man diese der Wunschplanung bzw. den Vorgaben des Publisher gegenüberstellen. Jetzt – und nur jetzt – lassen sich Maßnahmen sinnvoll beschließen. Müssen Features gestrichen werden? Reichen die aktuell zur Verfü-



Projekt-Kick-Off und Team-Meetings sind nur effizient, wenn sie strukturiert durchgeführt werden – eine Moderation sowie eine Agenda sind hierfür unverzichtbar.

gung stehenden Ressourcen aus? Diese Entscheidungen müssen zu Beginn der Produktion getroffen werden, nicht im Laufe oder gar gegen Ende des Projekts. Wenn der Developer zu diesem Zeitpunkt erkennt, dass er mehr Mitarbeiter benötigt (was fast automatisch höhere Kosten bedeutet), dann muss er das dem Publisher sofort sagen. Viele Entwickler trauen sich das allerdings nicht aus Sorge, den Auftrag dann nicht zu bekommen. Sie starten lieber das Projekt und warten den berühmten »Point of no return« für den Publisher ab, also den Zeitpunkt, zu dem der Auftraggeber schon so viel Geld investiert hat, dass er nicht mehr zurück kann. Dann bringen sie die Probleme auf den Tisch und erklären, dass sie mehr Ressourcen benötigen. Ein verständliches Vorgehen, bedenkt man die Uneinsichtigkeit mancher Publisher – aber leider kein sinnvolles. Der Publisher wird sich verständlicherweise verschaukelt vorkommen und oftmals die benötigten Ressourcen auch nicht genehmigen. Was dazu führt, dass der Rest des Teams in einen monatelangen Crunsh-Mode wechseln muss und ausbrennt.

Und selbst wenn der Publisher die Budgeterhöhung genehmigt, also mitten in die laufende Entwicklung neue Ressourcen zu werfen, führt das trotzdem nur selten zu den gewünschten Ergebnissen. Neue Mitarbeiter müssen erstmal eingearbeitet werden und schaffen es nicht, ihre Effektivität von 0 auf 100 in zwei Tagen zu steigern. Und das weit größere Problem besteht im Finden der benötigten Mitarbeiter, wenn man nicht schon Monate vorher anfängt, nach ihnen zu suchen.

Ralf C. Adam

### In der nächsten Ausgabe...

In Teil 2 unserer Serie beschäftigen wir uns unter anderem mit der oft fehlinterpretierten Rolle des Projektleiters, klassischen Denkfehlern beim Abschätzen von Arbeitsaufwänden sowie Publishern, die die Projektarbeit sabotieren.

### Weiterführende Quellen

- Wilfried Reiter: »Die nackte Wahrheit über Projektmanagement«, Orell Füssli 2003 (ISBN: 3-280-05018-9)
- Erik Bethke: »Game Development and Production«, Wordware Publishing 2003 (ISBN: 1-55622-951-8)
- Steve McConnell: »Rapid Development«, Microsoft Press 1996 (ISBN: 1-55615-900-5)
- Steve McConnell: »Software Project Survival Guide«, Microsoft Press 1998 (ISBN: 1-57231-621-7)
- Dan Irish: »The Game Producer's Handbook«, Thomson 2005 (ISBN: 1-59200-617-5)