

## Vorwort

Dem Projektmanagement stehen heute viele bewährte Standards (Prozesse, Techniken, Werkzeuge etc.) zur Verfügung. Obwohl beispielsweise die Erfolgsquote von IT-Projekten in den letzten Jahren etwas gestiegen ist, scheitern oder kriseln immer noch viele Projekte. Zwar sind Projekte trotz aller Standards immer wieder individuelle Herausforderungen, doch die Projekte werden zunehmend schneller, größer, inhaltlich und technisch komplexer, räumlich stärker verteilt, haben dynamischere Ziele sowie instabilere Anforderungen und Rahmenbedingungen. Die Ansprüche steigen also, erfordern möglicherweise immer mehr spezifischere Antworten und damit agilere Vorgehens- und Handlungsweisen.

In dieses Gebiet sind wir mit der Tagung interPM 2006 unter dem Titel »Agiles Projektmanagement« vorgestoßen.

Die interPM ist eine seit mehreren Jahren stattfindende interdisziplinäre Tagung zum Thema Projektmanagement. Sie wird getragen durch die Fachgruppe Projektmanagement in der Gesellschaft für Informatik e.V. (GI) und der Deutschen Gesellschaft für Projektmanagement e.V. (GPM) in Zusammenarbeit mit dem Berufsverband Deutscher Soziologinnen und Soziologen e.V. (BDS). Anders als andere Tagungen ist die interPM ein lebendiger Ort der Diskussion und des Gedanken- und Erfahrungsaustausches, da parallel zu klassischen Vortragseinheiten ein OpenSpace läuft, d.h. durch die Teilnehmer selbst initiierte und gestaltete Workshops zu speziellen Themen und Fragen.

In diesem Buch sind die eingereichten und im Vorwege ausgewählten Beiträge der interPM 2006 enthalten. Insofern gibt dieses Buch nur einen Teil dessen wieder, was auf der interPM vorgetragen, berichtet und diskutiert worden ist. Die Beiträge haben sehr unterschiedliche Ausgangspunkte, Zielrichtungen und Schwerpunkte, um das Themengebiet Agilität von verschiedenen Perspektiven betrachten zu können. Es sind Beiträge von gestandenen Praktikern, Projekt- und Praxisberichte ebenso vertreten wie wissenschaftliche und theoretische Arbeiten, Studien und Ähnliches.

Geprägt durch die Gesellschaft für Informatik e.V. als Mitveranstalter beziehen sich einige Beiträge ausschließlich auf den Bereich der Softwareentwicklung.

Durch den interdisziplinären Ansatz brachte die interPM 2006 jedoch Vertreter unterschiedlicher Berufsgruppen in den Dialog über mehr Agilität im Projektmanagement. Ziel war es, die Perspektiven, Ansätze und Konzepte verschiedener Disziplinen zum Projektmanagement (z.B. Betriebs- und Ingenieurwissenschaften, Soziologie, Psychologie, Informatik, Beratungs- und Bildungsinstitutionen, PM-Schulen und -verbände) einander gegenüberzustellen, um diese zu diskutieren, zu reflektieren und zu verbinden mit dem Ziel, etwas »Neues« entstehen zu lassen.

Vielen herzlichen Dank also an alle Unterstützer und Beitragenden, die die Tagung und dieses Buch ermöglicht und mitgestaltet haben! Ganz besonders bedanke ich mich bei meinen Mit-Organisatoren Andreas Frick, Gerrit Kerber, Dr. Roland Marré und Dr. Erich Behrendt sowie bei Christa Preisendanz vom dpunkt.verlag.

*Bernd Oestereich*

Hamburg, April 2006

# Inhalt

<b>Mit erfahrungsgeleitetem Handeln zu mehr Agilität im Projektmanagement</b> <i>Prof. Dr. Fritz Böhle</i>	<b>1</b>
<b>Transaktionskosten in Softwareprojekten – Wege zur nachhaltigen Kundenbeziehung</b> <i>Dr. Ralf Zarsteck</i>	<b>13</b>
<b>Die kreative Projektphase, das dritte Prozessmodell neben Critical Path und Critical Chain</b> <i>Dr.-Ing. Dieter Geckler</i>	<b>29</b>
<b>Agile Prozesse durch erfolgreiches Change Leadership</b> <i>Gustav Giest</i>	<b>47</b>
<b>Timeboxing – das Rückgrat agiler Projekte</b> <i>Bernd Oestereich</i>	<b>53</b>
<b>Wann sind agile Softwareprojekte erfolgreich? Ergebnisse einer empirischen Studie</b> <i>Dr. Ralph Trittman</i>	<b>69</b>
<b>Methodische Aufwandsschätzung aus Sicht eines agilen Projektmanagements</b> <i>Dr. Hans-Jürgen Plewan</i>	<b>83</b>
<b>Integratives fachkonzeptionelles Projektmodell zur Analyse von SCRUM</b> <i>Prof. Dr. Jörg Becker · Dipl.-Wirt.-Inform. Michael Thygs</i>	<b>101</b>
<b>Agile Projektmanagementmethoden für Produktlinien</b> <i>Dr. Florian Harzenetter · Tobias Stenzel · Ludger Meyer</i>	<b>115</b>
<b>Agile Softwareentwicklung nach Winston Royce</b> <i>Jens Himmelreich</i>	<b>123</b>
<b>Risikomanagement in Projekten Eine entscheidungsorientierte Betrachtung</b> <i>Prof. Dr. Nino Grau · Prof. Dr. Ulrich Vossebein</i>	<b>135</b>

<b>Messung und Erklärung der Agilität von Unternehmen aus dem Blickwinkel des Multiprojekt-Managements</b> <b>Eine konzeptionelle Analyse und empirische Validierung</b> <i>Dipl.-Wi.-Ing. Henning Dammer · Prof. Dr. Hans Georg Gemünden</i>	<b>145</b>
<b>Technik als Kommunikationsanlass in agilem Projektmanagement</b> <i>Dr. Hans-Dieter Schat</i>	<b>165</b>
<b>Der Ingenieur als agiler Projektmanager und der Soziologe als agiler Dienstleister im Softwarewareentwicklungsprozess – Erfahrungsbericht</b> <i>Dipl.-Ing. Tom Tiltmann · Dr. phil. Leonie Petry · Prof. Dr.-Ing. Klaus Henning</i>	<b>181</b>
<b>Projektorientierter Wissensmarkt</b> <b>Agiles Projektmanagement durch einen marktbezogenen Ansatz des Wissensmanagements</b> <i>Dr.-Ing. Wolfgang Arens-Fischer</i>	<b>195</b>
<b>Agiles Projektmanagement und die sieben Todsünden</b> <b>Ein »neuer« Blick auf die menschliche Seite der Projektarbeit</b> <i>Dipl.-Oec. Rainer Fassnacht</i>	<b>215</b>
<b>Projekterfolgssicherung durch Sozialkompetenz</b> <b>Agiles Projektmanagement über fachliche Qualifikationen hinaus</b> <i>Dipl.-Ing. (FH) Joachim Schmid</i>	<b>223</b>
<b>Unternehmenskultur und IT-Projektmanagement</b> <b>Ergebnisse einer empirischen Studie</b> <i>Dipl.-Inf. (FH), MBA Markus Steidle</i>	<b>231</b>
<b>Mitarbeiterbeteiligung als Basis erfolgreicher Projekte – das Klinikencafé</b> <i>Dipl.-Soz. Hartmut O. Genz</i>	<b>245</b>
<b>Interdisziplinäre Prognosesysteme im Projektmanagement</b> <b>Methodische Erstellung und Anwendung zur Unternehmensdiagnose</b> <i>Dipl.-Wirt.-Inf. Torben Hansen · Prof. Dr. Peter Loos</i>	<b>257</b>
<b>Agile Projektmanagement-Prinzipien – Ein evolutionärer Management-Ansatz?</b> <b>»Projektmanagement 2. Ordnung« als Referenzmodell</b> <i>Dipl.-Ing. Manfred Saynisch</i>	<b>273</b>
<b>Umgang mit Komplexität im Projektmanagement</b> <i>Dipl.-Inform. Alexander Haasper</i>	<b>305</b>
<b>Autoren</b>	<b>325</b>

# Mit erfahrungsgelitetem Handeln zu mehr Agilität im Projektmanagement

Prof. Dr. Fritz Böhle

Philosophisch-Sozialwissenschaftliche Fakultät, Universität Augsburg

## Zusammenfassung

*Dass Projekte in ihrem zeitlichen Ablauf, den Kosten und/oder dem inhaltlichen Ergebnis von den ursprünglichen Zielsetzungen und Planungen abweichen, ist keine Seltenheit. Oft ist es eher der Normalfall (vgl. Fröhlich 2002). Bisher sind zwei Reaktionen hierauf vorherrschend: die Optimierung der Planung und die Verdrängung des Unplanbaren. Beides sei kurz näher erläutert, um daran anschließend eine andere, bisher kaum beachtete Strategie vorzustellen.*

## 1 Das Leitbild der Planung

### 1.1 Die Optimierung der Planung und die Verdrängung des Unplanbaren

Vor allem in der wissenschaftlichen Diskussion, Ausbildung und Beratung werden die Ursachen für Probleme bei der Durchführung von Projekten in einer unzureichenden Projektplanung und -steuerung gesehen. Dementsprechend richten sich hier die Bemühungen darauf, Modelle und Tools für die Planung und Steuerung von Projekten zu entwickeln. Dabei kann unterschieden werden zwischen organisatorischen, technischen und personellen Ansätzen. Beispiele für organisatorische Ansätze sind das Wasserfallmodell oder die Festlegung von Meilensteinen u. Ä. Beispiel für technische Ansätze sind informations- und kommunikationstechnisch gestützte Planungs- und Steuerungs-Tools (vgl. als Überblick Schelle 1999). Personelle Ansätze richten sich demgegenüber primär auf die Humanressourcen und beinhalten vor allem Ausbildungs- und Schulungskonzepte. Problemen in der Praxis wird dementsprechend mit einer Verfeinerung und Weiterentwicklung organisatorischer Modelle, technischer Tools sowie von Aus-

bildungs- und Schulungskonzepten begegnet. Die Lösung wird darin gesehen, dass die »Praxis« die hierdurch bereitgestellten Hilfsmittel nutzt und »richtig« anwendet. Misserfolge werden vor diesem Hintergrund oft auf eine unzulängliche Anwendung der vorgeschlagenen Verfahren bei der Projektplanung und -steuerung zurückgeführt.

Obwohl mittlerweile nicht nur in der Theorie, sondern auch in der Praxis ein vergleichsweise hohes Niveau an Planungs- und Steuerungsmodellen, Tools sowie Ausbildungs- und Schulungskonzepten erreicht ist, sind die Erfolge in der Praxis dennoch nicht zufrieden stellend. Die Strategie der Herstellung von Planbarkeit ist angesichts der Bedeutung von Kundenorientierung und Flexibilität nicht beliebig durchsetzbar und/oder entpuppt sich als Problem- und Kostenverlagerung. Trotz Optimierung der Projektplanung und -steuerung treten in der Praxis immer wieder nicht vollständig kontrollierbare Ereignisse und Einflussgrößen bei der Realisierung von Projekten auf. Dies führt jedoch in der Regel nicht dazu, die eingeschlagenen Wege der Projektplanung und -steuerung zu überdenken – im Gegenteil: Zumeist verstärkt es eher das Bestreben zu weiterer Optimierung. Die Ursachen für Abweichungen von Geplantem und Schwierigkeiten in der konkreten Projektsteuerung werden dabei als etwas angesehen, was durch eine bessere Planung grundsätzlich in den Griff zu bekommen ist, oder sie werden für zukünftige Entwicklungen als nicht relevant ausgewiesen. Gerade Letzteres hat den Effekt, dass eine grundsätzlichere Auseinandersetzung mit den Ursachen und Grenzen der Planung und Planbarkeit von Projekten unterbleibt.

## 1.2 Die Verdrängung des Nicht-Planbaren

Neben dem Versuch, die Planung zu optimieren, ist eine andere Strategie, Einflussgrößen und Ereignisse, die zu Abweichungen von Planungen und Schwierigkeiten bei der Projektsteuerung führen, als »Sonderfall« zu betrachten. In dieser Sicht lassen sich aus »Fehlentwicklungen« keine generellen Folgerungen ableiten, und in Folge ihrer »Einmaligkeit« bieten sie dafür auch keinen Anlass. Beim nächsten Mal, so die Annahme, werden die Bedingungen wieder »normal« sein. Unsere Untersuchungen mit der GPM Deutsche Gesellschaft für Projektmanagement e.V. in der Automobilindustrie zeigen, dass in diesem Zusammenhang die Suche nach einem »Schuldigen« weit verbreitet ist (vgl. Pander u. Wagner 2005). Dies können sowohl der Projektleiter oder Projektmitarbeiter wie auch Kunden und auch von den jeweiligen Projekten direkt und indirekt Betroffene sein. Die Ursachen für Fehlentwicklungen werden damit personalisiert und individualisiert.

Wenn doch Zweifel daran aufkommen, dass die in der Praxis auftretenden Grenzen der Planung und Planbarkeit überwunden und ausgeschaltet werden können, bleibt dies weithin folgenlos, da kaum eine ernst zu nehmende Alternative zu den eingeschlagenen Lösungswegen zu bestehen scheint. Etwas überspitzt ausgedrückt: Was nützt der Zweifel, wenn er eher zu Handlungsunfähigkeit als

zu Handlungsfähigkeit führt? So sind es letztlich auch die äußeren Zwänge – die notwendige kostenmäßige Kalkulation und frühzeitige Definition des Ergebnisses sowie die Kontrolle und Dokumentation von Projektverläufen –, die oft wider besseres Wissen (oder sollte man sagen: »Ahnungen«) dazu (ver-)führen, am Leitbild der Planung und Planbarkeit von Projekten festzuhalten.

## 2 Erfahrungsgelitetes Projektmanagement

Im Folgenden sei eine Form des Projektmanagements vorgestellt, die sich zum einen vom Leitbild der Planung und Planbarkeit unterscheidet, zum anderen aber auch auf mehr Agilität im Projektmanagement zielt. Grundlage hierfür ist das Konzept des erfahrungsgeliteten Handelns. Dies sei zunächst kurz näher erläutert (2.1), um daran anschließend die sich hieraus ergebenden Konsequenzen für die Anlage und Durchführung von Projekten näher zu umreißen. Im Einzelnen betrifft dies die Projektorganisation (2.2 und 2.3), die konkrete Durchführung und Steuerung von Projekten (2.4) sowie die Kooperation zwischen den am Projekt beteiligten Personen (2.5).

### 2.1 Erfahrungsgelitetes Handeln

Das Konzept des erfahrungsgeliteten Handelns wurde in Untersuchungen zur Arbeit in der industriellen Produktion entwickelt. Hier zeigt sich, dass auch bei fortschreitender Automatisierung – entgegen ursprünglichen Prognosen – menschliche Arbeit unverzichtbar bleibt und sich die Anforderungen an menschliche Arbeit keineswegs auf einfache Restarbeiten beschränken. Ein Grund hierfür liegt in Grenzen der wissenschaftlich-technischen Beherrschung konkreter Produktionsabläufe. Je mehr die Technisierung industrieller Produktion fortschreitet, umso mehr zeigt sich, dass die wissenschaftlich-technische Durchdringung von Produktionsprozessen nur begrenzt möglich ist. Die wissenschaftliche Abstraktion, die Berechenbarkeit und Kontrollierbarkeit konkreter Gegebenheiten stößt in dem Maße an Grenzen, wie Einflussgrößen und Rahmenbedingungen vielfältiger und komplexer werden. Abweichungen vom geplanten Verlauf und Unwägbarkeiten werden damit zur Normalität (vgl. Böhle 1998).

Eine wichtige Aufgabe menschlicher Arbeit in hoch technisierten Prozessen besteht darin, wissenschaftlich-technisch nicht erfassbare und ausschaltbare Unwägbarkeiten durch regulierende Eingriffe zu bewältigen. Nach einer weit verbreiteten Auffassung ist hierfür ein besonderes, wissenschaftlich fundiertes Fachwissen und die Fähigkeit zu abstraktem analytisch-logischem Denken sowie planmäßig-systematisches Vorgehen notwendig. Wie empirische Untersuchungen jedoch zeigen, trifft dies nur bedingt zu: Theoretisches Wissen, abstraktes Denken und systematisches Vorgehen sind zwar notwendig, reichen aber zur Bewältigung der Unwägbarkeiten hoch technisierter Systeme nicht aus. Neben dem Fachwis-

sen ist auch ein besonderes Erfahrungswissen notwendig. Dieses Erfahrungswissen besteht nicht nur in detaillierten Kenntnissen praktischer Gegebenheiten oder bestimmten Arbeitsroutinen. Vielmehr handelt es sich um ein Wissen, das auf einer besonderen Arbeitsweise beruht und hierin eingebunden ist. Im Unterschied zum planmäßigen Handeln lässt sich diese Arbeitsweise als ein erfahrungsgeleitetes Handeln bezeichnen. Betont wird damit die besondere Rolle des Erfahrungsmachens und der sinnlichen Wahrnehmung als Grundlage sowohl des Erwerbs von Wissens als auch der Regulierung des Handelns.

Im Unterschied zu einem planmäßigen Handeln sind beim erfahrungsgeleiteten Handeln Planung und Durchführung praktischen Handelns unmittelbar miteinander verschränkt, charakteristisch ist daher nicht der Grundsatz: »Erst denken (planen), dann handeln«, sondern vielmehr ein schrittweises, exploratives Vorgehen, bei dem die Handlungsziele im Vollzug praktischen Handelns konkretisiert werden und einzelne Handlungsschritte erst schrittweise in Reaktion auf das jeweils vorangegangene Ergebnis entwickelt werden. Eine solche Vorgehensweise ist beim erfahrungsgeleiteten Handeln verbunden mit einer komplexen sinnlichen Wahrnehmung, die sich nicht nur auf eindeutig definierte oder messbare, sondern auch auf diffuse und vielschichtige Informationsquellen richtet und auch mit Vorstellungen (Imaginationen) über aktuell nicht sinnlich Wahrnehmbares verbunden wird. Subjektives Empfinden und Gefühle sind hierbei im Unterschied zu einem planmäßigen Handeln nicht ausgeschaltet, sondern vielmehr wichtige Grundlagen der Wahrnehmung und Beurteilung des Wahrgenommenen. In der Praxis wird eine solche Wahrnehmung oft als »Gespür« beschrieben. Sinnliche Wahrnehmungen solcher Art sind verbunden mit assoziativem anschaulichem Denken. Denken erfolgt nicht nur in Begriffen, sondern vor allem in Form von Bildern und assoziativen Verknüpfungen eines bestimmten Ereignisses mit ähnlichen selbst erfahrenen und erlebten Situationen. Und schließlich werden auch Dinge und Sachverhalte nicht nur als leblose Objekte, die nach Gesetzmäßigkeiten funktionieren, begriffen, sondern mit etwas Lebendigem verglichen, mit dem man in einen Dialog tritt und zu dem man eine bestimmte persönliche Beziehung entwickelt (vgl. zum Konzept des erfahrungsgeleiteten Handelns Böhle u. Schulze 1997; Böhle 1999; Bauer u. a. 2002, Kap. I).

Mit dem Konzept des erfahrungsgeleiteten Handelns wird die grundsätzliche Überlegenheit eines planmäßigen Handelns relativiert. Es macht deutlich, dass in der Praxis auftretende Grenzen der Planung und Planbarkeit nicht notwendigerweise zu einem Verlust an Handlungsfähigkeit führen. Dies ist nur dann der Fall, wenn allein das planmäßige Handeln als effektiv und angemessen betrachtet wird und dementsprechend Abweichungen hiervon als Indiz für Unzulänglichkeiten und für »muddling through« gelten. Das Konzept des erfahrungsgeleiteten Handelns, für das sich empirische Belege nicht nur in der industriellen Produktion, sondern auch bei Experten in unterschiedlichen Professionen finden (vgl. Schön 2002), macht demgegenüber darauf aufmerksam, dass in der Praxis nicht nur

planmäßiges Handeln, sondern auch noch »andere« Handlungsstrategien notwendig (!) sind und je nach situativen Gegebenheiten entwickelt und genutzt werden müssen. Nicht-planmäßiges Vorgehen ist daher nicht notwendigerweise eine »Abweichung« vom Ideal der Planung, sondern weist auf der Grundlage des erfahrungsgeleiteten Handelns eine eigene Systematik und Handlungslogik auf. Es erweist sich speziell in Situationen als angemessen und dem planmäßigen Handeln überlegen, in denen nicht-antizipierbare Ereignisse sowie nicht berechenbare und nicht kontrollierbare Einflussgrößen auftreten.

Im Folgenden sei gezeigt, in welcher Weise das Konzept des erfahrungsgeleiteten Handelns für das Projektmanagement neue Perspektiven eröffnet. Um Missverständnisse zu vermeiden: Das Leitbild des planmäßigen Handelns wird damit nicht hinfällig, es wird jedoch im Sinne eines Sowohl-als-auch ergänzt und modifiziert. Ebenso wie beim Umgang mit technischen Systemen ist auch beim Projektmanagement sowohl planmäßiges als auch erfahrungsgeleitetes Handeln notwendig. Da der bisherige Schwerpunkt des Projektmanagements auf dem planmäßigen Handeln liegt, seien im Folgenden primär die Konsequenzen, die sich aus dem Konzept des erfahrungsgeleiteten Handelns für das Projektmanagement ergeben, aufgezeigt.

## 2.2 Anerkennung der Grenzen der Planbarkeit

Im Unterschied zum Leitbild der Planung und Planbarkeit beruht das erfahrungsgeleitete Handeln auf der grundsätzlichen Anerkennung von Grenzen der Planbarkeit konkreter Gegebenheiten. Diese werden nicht als Planungsdefizit oder Sonderfall begriffen, sondern vielmehr als Normalfall. Auch bei bester Planung ist demzufolge mit nicht-antizipierbaren Ereignissen und nicht vollständig kontrollierbaren und berechenbaren Einflussgrößen zu rechnen.

Betrachtet man die Abwicklung von Projekten nicht nur als einen sachlich-technischen, sondern auch als einen sozialen und interessenpolitischen Prozess, so sind Probleme in der Kommunikation und Konflikte strukturell (!) angelegt und somit ein Tatbestand, mit dem grundsätzlich zu rechnen ist. Dies ist jedoch nicht im Sinne einer Personalisierung und Individualisierung von Planungs- und Steuerungsproblemen misszuverstehen (vgl. 1.2). Es geht hier nicht primär um individuelle Verhaltensweisen (obwohl diese auch eine Rolle spielen), sondern vor allem um (mögliche) Konflikte und Missverständnisse, die beispielsweise aus unterschiedlichen beruflichen Orientierungen (etwa bei interdisziplinärer Zusammenarbeit und bei abteilungs- und betriebsübergreifenden Projekten) und aus der Einbindung der am Projekt beteiligten Personen in unterschiedliche institutionelle und organisatorische Zusammenhänge, denen sie (auch) verpflichtet sind, resultieren.

Unwägbarkeiten im Projektmanagement resultieren in der Praxis nicht nur aus sozialen, sondern auch aus sachlich-technischen Gegebenheiten. Bekannt ist, dass sich einfach erscheinende technische Aufgaben im konkreten Fall oft als

höchst kompliziert erweisen oder notwendige Ressourcen sachlich und/oder zeitlich nicht in der erforderlichen Weise verfügbar sind. Und schließlich ist gerade bei neuen Projekten grundsätzlich damit zu rechnen, dass nicht sämtliche Einflussgrößen ex ante angemessen berücksichtigt bzw. vorhergesehen werden können. Insbesondere unter den Bedingungen innovativer Entwicklungen und eines dynamischen Umfeldes ist daher die Planung von Projekten grundsätzlich (!) ein Vorhaben, das immer nur annähernd erreicht werden kann, da die internen und externen Bedingungen, unter denen Projekte stattfinden, nur begrenzt den Kriterien der Planbarkeit im Sinne berechenbarer, antizipierbarer und kontrollierbarer Gegebenheiten entsprechen. Dabei ist auch zu berücksichtigen, dass Projekte zumeist nicht nach dem Modell linearer Kausalität ablaufen, sondern einen systemischen Charakter aufweisen, Abweichungen von Plangrößen lassen sich daher auch in ihren Wirkungen nur sehr schwer exakt abschätzen und eingrenzen.

### 2.3 Offene Planung

Auf der Grundlage des erfahrungsgeleiteten Handelns erfolgt kein Verzicht auf Planung. Die Planung erhält hier jedoch eher den Stellenwert eines Rahmens, der im konkreten Handeln (erst) ausgefüllt wird. Dies bezieht sich nicht nur auf die Festlegung einzelner Arbeitsschritte und die Verfügung über Ressourcen, sondern auch auf die Definition von Zielen und Ergebnissen. Werden Grenzen der Planbarkeit anerkannt, hat dies auch Auswirkungen auf die Definition der Projektziele. Es bleibt nicht nur offen, in welcher Weise ein bestimmtes Ziel erreicht wird, sondern auch, wie das Ergebnis eines bestimmten Projekts aussieht. Folgt man dem Konzept des erfahrungsgeleiteten Handelns, so ist dies kein Mangel. Es wird hiermit vielmehr dem Tatbestand Rechnung getragen, dass gerade bei innovativen Projekten praktisches Handeln nicht nur der Vollzug vorangegangener Planungen ist, sondern erst im (!) praktischen Handeln sowohl das Prozedere als auch die (möglichen) Ergebnisse entstehen. Dies beschränkt sich nicht auf die Konkretisierung und/oder Überprüfung des Realitätsgehalts vorangegangener Zielsetzungen, sondern beinhaltet auch die Entdeckung neuer Möglichkeiten und Ziele. Im Rahmen einer offenen Planung werden daher nicht nur die konkrete Durchführung, sondern auch die anzustrebenden Ergebnisse auf einige Fixpunkte beschränkt. Damit wird beispielsweise bei der Einführung von IuK-Systemen berücksichtigt, dass der Kunde zumeist ex ante kaum präzise formulieren kann, was er benötigt, sondern sich dies erst im Prozess der Einführung von IuK-Systemen (oder entsprechender Umstellungen anderer Systeme) ergibt.

Kundenwünsche während des Projektverlaufs sind so gesehen keine Störgröße oder auf unzuverlässiges Kundenverhalten zurückzuführen, sondern entsprechen der »Natur der Sache«. Im praktischen Vollzug wird vielfach erst erkennbar, was tatsächlich benötigt wird und was als geeignet erscheint. Umgekehrt gilt aber auch, dass externe Kunden oder die von Projekten direkt oder indirekt Betroffene

nen innerhalb des Betriebs auch Abweichungen von ursprünglichen Zielsetzungen nicht per se als Unzuverlässigkeit und Planungsdefizit abbuchen, sondern prüfen, inwieweit sich damit nicht nur negative, sondern vor allem auch positive Effekte verbinden. Dies kann sich sowohl auf inhaltliche als auch auf zeitliche Effekte und nicht zuletzt auch auf Kosten beziehen. Das ist nicht als Freibrief für Planlosigkeit und Beliebigkeit (miss-) zu verstehen. Kriterien für die Beurteilung sind (bleiben) auch hier die Relation von Kosten und Ertrag sowie die Nützlichkeit des Ergebnisses. Entscheidend ist jedoch, dass dies nicht allein an den ursprünglichen Planungen, sondern an den faktischen Ergebnissen gemessen wird.

## 2.4 Situative Projektsteuerung

Die Steuerung des Projektverlaufs ist in der Regel die Aufgabe von Projektleitern oder ähnlichen Positionen. Dabei ist die Bewältigung sachlich-technischer und ökonomischer Aufgaben zu einem großen Teil abhängig von der Interaktion und Kommunikation mit den an den Projekten Beteiligten. Ein planmäßiges Handeln, das sich auf die inhaltliche und zeitliche Festlegung und Verteilung von Aufgaben sowie die Kontrolle der Einhaltung getroffener Vereinbarungen richtet, reicht hierfür nicht aus. Dies ist mittlerweile ein weithin bekannter Sachverhalt; gefordert und gefördert werden daher in der Praxis neben fachlichen soziale Kompetenzen (vgl. Brodbeck u. Frese 1994).

Dabei geht es in der Praxis jedoch nicht nur um ergänzende »soft skills«, die sich darauf richten, durch »soziales Geschick« die planmäßige Steuerung praktisch um- und durchzusetzen. Notwendig ist vielmehr auch (!) die Verbindung sozialer Kompetenzen mit einem erfahrungsgeleiteten Handeln, das sich darauf richtet, (gemeinsame) Lösungswege zu erarbeiten sowie unterschiedliche Interessen auszutarieren. Im Folgenden sei dies anhand von Ergebnissen aus einer Untersuchung zur Rolle erfahrungsgeleiteter Arbeit im Rahmen betriebsübergreifender Projekte näher ausgeführt (vgl. Heidling, Meil u. Rose 2003).

Bei der Interaktion und Kommunikation mit den an dem Projekt beteiligten Personen ist vor allem bei betriebsübergreifenden Projekten ein flexibler Umgang mit jeweils betriebsspezifischen Kulturen notwendig. In Verhandlungssituationen mit Vertretern verschiedener Unternehmen (Gleiches gilt im Prinzip auch für unterschiedliche betriebsinterne Abteilungen) müssen von den Projektleitern ein Interessenabgleich und Interessenausgleich sowie hierauf bezogene Lösungen erarbeitet werden. Dies setzt voraus, dass die unterschiedlichen Positionen ernst genommen werden und die Projektleiter in der Lage sind, sich situativ auf die verschiedenen, ex ante vielfach nicht absehbaren und einschätzbaren Positionen einzustellen. Sie müssen somit in der Lage sein, auch ihre eigenen Zielsetzungen und Vorstellungen an die jeweilige Situation anzupassen und zu modifizieren. Zu moderieren sind dabei Prozesse, die häufig auch durch ungleich verteilte Machtpotenziale – etwa zwischen Herstellern und Zulieferern, zwischen Systemliefe-

ranten und den ihnen zuliefernden Unternehmen – geprägt sind. Die Projektleiter müssen somit in der Lage sein, eigene Interessen zu vertreten und durchzusetzen, aber auch Kompromisse einzugehen, angesichts der realen Machtkonstellation die eigenen Handlungsmöglichkeiten einzuschätzen und die Projektsteuerung hieran auszurichten. Gerade solche Interessen- und Machtkonstellationen sind in ihren konkreten Wirkungen schwer absehbar und verlangen in hohem Maße die Fähigkeit zu einem situativen Ausbalancieren von Zuständigkeiten, Aufgaben und Ressourcenverteilung.

Ein weiteres Merkmal des hierbei praktizierten erfahrungsgeleiteten Handelns ist die Fähigkeit, sich von Projekten und Ergebnissen »ein Bild zu machen« bzw. in Prozessketten optimale Ergebnisse »vorherzusehen«. Erreicht wird dies durch die Bildung von Analogien und Bezügen zu früheren (Projekt-)Erfahrungen. Auch für die einzelnen Projektmitarbeiter ist dies wichtig. Sie sind in der Regel nur für einzelne Teilkomponenten eines Gesamtproduktes verantwortlich; sie müssen jedoch in der Lage sein, die Integration ihrer Komponente in das Gesamtprodukt zu gewährleisten und sich entsprechend hieran zu orientieren. Bildhafte Vorstellungen des Gesamtproduktes bis hin zu »Visionen« werden hier von Projektmitarbeitern wie Projektleitern als hilfreich, wenn nicht unverzichtbar eingeschätzt.

Sowohl für die Projektleiter als auch für die Mitarbeiter spielt ein analytisches, logisch-formales Denken für die Bewältigung einzelner (Teil-)Aufgaben ebenso wie für das Verstehen tätigkeitsübergreifender Prozesse eine wichtige Rolle. Zugleich ist aber ebenso von Bedeutung eine Offenheit für unterschiedliche Sicht- und Denkweisen und nicht-lineare, prozesshaft-vernetzte Entwicklungen. Die Projektleiter müssen in der Lage sein, die komplexen Wirkungen und Rückwirkungen einzelner Entscheidungen und Handlungen auch ohne exakte Informationen ein- und abzuschätzen. Gleiches gilt auch für die Antizipation zukünftiger Entwicklungen auf der Grundlage von »sticky informations«. Analytisches Denken wird hier ergänzt durch die Assoziation vergleichbarer Situationen, die Aktualisierung eines entsprechenden Erfahrungswissens, des Weiteren aber auch durch das Sicheinlassen auf »prospektive« Erfahrungen und imaginative (praktische) Auseinandersetzungen mit möglichen Entwicklungsszenarien. Man analysiert dabei nicht nur zukünftige Entwicklungen, sondern stellt sie sich konkret vor und erlebt sie. Typisch hierfür sind Äußerungen wie: »Ich sehe das wie in einem Film«, oder: »Ich spiele das nicht nur in Gedanken durch, sondern versetze mich in die Situation hinein.« Und schließlich ist es in der Interaktion und Kommunikation sowohl für Projektleiter als auch für Mitarbeiter notwendig, die sachlichen Arbeitsbeziehungen durch wechselseitiges Vertrauen sowie durch Verständnis für unterschiedliche Arbeitsstile und ggf. Stärken und Schwächen der am Projekt Beteiligten zu ergänzen. Die sachlich-instrumentelle Kooperation bedarf hierzu einer Ergänzung durch eine empathische Kooperation (Spieß 1996).

## 2.5 Informelle, erfahrungsgelitete Kooperation

Die Abstimmung zwischen den am Projekt Beteiligten erfolgt in der Regel in Form gemeinsamer Besprechungen. Die hier stattfindenden Meetings finden vor allem in der Anfangsphase zur Abstimmung von Projektzielen und einzelnen (Teil-)Aufgaben statt; im Verlauf des Projektes dienen sie zur Präsentation und Überprüfung von Zwischenergebnissen sowie zur Abstimmung des weiteren Vorgehens. Zumeist werden hierzu so genannte Meilensteine im Projektverlauf festgelegt. In der Praxis kommt es jedoch trotz solcher gemeinsamer Abstimmungen sowohl zu Unstimmigkeiten in einzelnen Teilergebnissen wie auch zu (Fehl-)Entwicklungen, die erst am Ende des Projektes erkennbar werden. Obwohl in den Meetings über den Projektverlauf und über auftretende Probleme berichtet und diskutiert wird, entstehen offenbar Missverständnisse und kommen wichtige Sachverhalte nicht zur Sprache (vgl. Grüter, Breuer u. Wollenberg 2001).

Wie eine genaue Betrachtung zeigt, resultiert dies nicht nur aus Unzulänglichkeiten der beteiligten Partner oder einer unzureichenden Organisation der Meetings. Wichtige Ursachen hierfür liegen vielmehr darin, dass solche Besprechungen im Normalfall von den eigentlichen Projektarbeiten räumlich und sachlich getrennt stattfinden und daher die Sachverhalte, um die es geht, nicht unmittelbar, sondern nur durch Beschreibungen präsentiert werden. Des Weiteren dienen solche Meetings in erster Linie der Abstimmung und somit der Planung und Koordination von (Teil-)Aufgaben und weit weniger einer kooperativen Suche nach Problemlösungen oder gar ihrer gemeinsamen Durchführung.

Diese Form der Kooperation entspricht dem Modell des planmäßigen Handelns. Dies zeigt sich auch darin, dass im Normalfall solche Meetings durch formelle Regelungen, in denen Zeitpunkt, Ort, Teilnehmer und Inhalt festgelegt sind, organisiert werden. Je mehr sie in dieser Weise vorbereitet werden, umso höher scheint ihre Effizienz. Dies trifft durchaus zu, gilt aber nur für solche Sachverhalte, die sich exakt beschreiben, abstimmen und planen lassen. Ein »Gefühl« oder eine »Ahnung«, dass möglicherweise etwas schief läuft, und Schwierigkeiten, die sich nicht präzise definieren lassen, können hier nur sehr schwer thematisiert oder gar diskutiert werden. Des Weiteren sind aktuell auftretende Probleme und Abstimmungserfordernisse oft erst mit erheblichem zeitlichem Abstand mitteilbar; sie werden dann zu einem Zeitpunkt besprochen, wenn es u.U. »schon zu spät« ist, oder sie verlieren ihre Brisanz, weil bereits in Folge von Zeit- und Kostendruck andere (Hilfs-)Lösungen gewählt wurden.

Das Konzept des erfahrungsgeliteten Handelns macht demgegenüber auf eine andere mögliche und notwendige Form der Kooperation in Projekten aufmerksam: die informelle Kooperation in der alltäglichen Arbeit (vgl. Böhle u. Bolte 2002). Sie erfolgt nicht nach festgelegten Regeln, sondern situativ je nach Anlass. Durch ein aktuell auftretendes Erfordernis oder auch Bedürfnis nach Abstimmung bzw. gegenseitiger Hilfe bestimmen sich hier Zeitpunkt, Partner

und Inhalt. Diese Form der Kooperation ist daher grundsätzlich »informell« und muss von den am Projekt Beteiligten selbst hergestellt werden. Des Weiteren erfolgt die Kooperation nicht abgesondert von den laufenden (Projekt-)Arbeiten, sondern ist unmittelbar in sie integriert. Sie findet statt am Arbeitsplatz zumindest eines Beteiligten und/oder »vor Ort« an den Sachverhalten, um die es geht. Dies hat zur Folge, dass die jeweiligen Sachverhalte nicht nur beschrieben werden, sondern auch unmittelbar demonstrierbar und erfahrbar sind. Auf dieser Grundlage wird es möglich, ein wechselseitiges Verständnis über die jeweiligen Projektarbeiten und über die Kontexte, in denen sie sich vollziehen, durch eigene Erfahrungen im Unterschied zu Beschreibungen aufzubauen. Wie die Praxis zeigt, können auf dieser Grundlage nicht nur aktuell auftretende Probleme rasch bewältigt, sondern vor allem Probleme aufgegriffen und oft auch erst entdeckt werden, die sich nicht präzise definieren und/oder in ihren Wirkungen exakt abschätzen lassen (Böhle u. Bolte 2002; Porschen 2002).

Für die Organisation und Steuerung von Projekten folgt hieraus, dass neben formell geregelten Meetings Freiräume und Möglichkeiten geschaffen werden müssen, durch die es möglich ist, je nach Bedarf informell in den laufenden Projektarbeiten zu kooperieren. Die formell geregelten Meetings dürfen daher nicht – wie es oft geschieht – als Ersatz für eine solche informelle Kooperation angesehen werden. Sie sind ohne Zweifel ein wichtiges Instrument zur Projektsteuerung. Ihre Leistungen richten sich jedoch primär auf das, was planbar ist. Die Bewältigung des Nicht-Planbaren erfordert demgegenüber nicht nur eine erfahrungsgeleitete Projektorganisation und -steuerung, sondern auch eine erfahrungsgeleitete Kooperation zwischen den an Projekten beteiligten Personen.

## Literatur

- Bauer, H. G.; Böhle, F.; Munz, C.; Pfeiffer, S.; Woicke, P. (2002): Hightech-Gespür – Erfahrungsgeleitetes Arbeiten und Lernen in hoch technisierten Arbeitsbereichen. Ergebnisse eines Modellversuchs beruflicher Bildung in der Chemischen Industrie. W. Bertelsmann, Bielefeld.
- Böhle, F. (1998): Technik und Arbeit – Neue Antworten auf »alte Fragen«. In: Soziale Welt, Heft 3, 49. Jg, S. 233-252.
- Böhle, F. (1999): Arbeit – Subjektivität und Sinnlichkeit, Paradoxien des modernen Arbeitsbegriffs. In: G. Schmidt (Hrsg.): Kein Ende der Arbeitsgesellschaft, Berlin, S. 89-109.
- Böhle, F. u. A. Bolte (2002): Die Entdeckung des Informellen – Der schwierige Umgang mit Kooperation im Arbeitsalltag. Campus, Frankfurt/New York.
- Böhle, F. u. H. Schulze (1997): Subjektivierendes Arbeitshandeln – Zur Überwindung einer gespaltenen Subjektivität. In: Ch. Schachtner (Hrsg.): Technik und Subjektivität. Frankfurt, S. 26-46.
- Brodbeck, F. C. u. M. Frese (Hrsg.) (1994): Produktivität und Qualität in Softwareprojekten. Oldenbourg Verlag, München/Wien.

- Fröhlich, A. W. (2002): *Mythos Projekt – Projekte gehören abgeschafft. Ein Plädoyer*. Galileo Business, Bonn.
- Grüter, B., Breuer, H. u. A. Wollenberg (2001): *Genese von Wissen in aufgabenorientierten Gruppen – Eine Fallstudie zur Wissensarbeit in der kommerziellen Softwareentwicklung*. In: E. H. Witte (Hrsg.): *Leistungsverbesserungen in aufgabenorientierten Kleingruppen*. Lengerich, S. 150-180.
- Heidling, E., Meil, P. u. H. Rose (2003): *Neue Anforderungen an Kompetenzen erfahrungsgeleiteten Arbeitens und selbstgesteuerten Lernens bei industriellen Fachkräften unter den Bedingungen verteilter Arbeit*. In: Universität Augsburg/ISF München: *Neue Anforderungen an industrielle Fachkräfte, hektographierter Forschungsbericht*, Augsburg.
- Pander, S. u. R. Wagner (2005): *Unternehmensübergreifende Zusammenarbeit in der Automobilentwicklung – durch erfahrungsgeleitete Kooperation die Grenzen der Planbarkeit überwinden*. Hampp, München/Mering.
- Porschen, S. (2002): *Erfahrungsgeleitete Kooperation im Arbeitsalltag. Neue Anforderungen an Ingenieure*. Reihe: ISF München Forschungsberichte, München.
- Schelle, H. (1999): *Projekte zum Erfolg führen – Projektmanagement systematisch und kompakt*. C. H. Beck, München.
- Schön, D. A. (2002): *The Reflective Practitioner – How Professionals Think in Action*. Ashgate Publishing Ltd., Aldershot, UK.
- Spieß, E. (1996): *Kooperatives Handeln in Organisationen*. Hampp, München/Mering.

# Timeboxing – das Rückgrat agiler Projekte

Bernd Oestereich

Geschäftsführer oose Innovative Informatik GmbH

## Zusammenfassung

*Seit 1998 unterstützt die Firma oose Innovative Informatik GmbH Unternehmen bei der Einführung iterativer und agiler Projektmanagementverfahren. Während die Einführung in kleineren Projekten mit bis ca. 10 Mitarbeitern gewöhnlich unproblematisch ist, erfordern größere Projekte, beispielsweise mit 80 oder 200 Projektmitarbeiterinnen, ausgeklügelte Prinzipien und Verfahren. In diesem Beitrag geht es um spezielle Managementverfahren, die helfen, Großprojekte agiler zu machen.*

*In kleinen Projekten kann eine einzelne Person, gewöhnlich die Projektleitung, direkt die Vorgehensweisen beeinflussen. Großprojekte sind durch deutlich mehr Eigendynamik einzelner Personen und Gruppen und chaotischere Strukturen und Abläufe gekennzeichnet. Andererseits lässt sich Agilität in Großprojekten auch nicht einfach zentral anordnen. Vielmehr geht es darum, einen konzeptionellen, aufbau- und ablauforganisatorischen Rahmen zu schaffen, der die einzelnen Personen und Einheiten zu mehr Agilität, Eigenverantwortung und Ergebnisorientierung anregt und der andererseits die Gesamtprojektorganisation von den auf dieser Ebene auftretenden chaotischen<sup>1</sup> Phänomenen abschirmt. Eine Gratwanderung zwischen ausreichend Freiraum (als notwendige Voraussetzung für Agilität) und ausreichend Systematik und Reproduzierbarkeit, so dass das Gesamtprojekt beherrschbar bleibt, d.h. Planung, Steuerung und Controlling mit angemessenem Aufwand laufen können.*

---

1. Chaotisch ist hier wertfrei, d.h. nicht negativ, gemeint.

*Die langjährigen Erfahrungen von oose mit vielen mittleren und großen Softwareprojekten zeigen, dass Timeboxing-Verfahren das Rückgrat hierfür sind.*

*Iteratives Vorgehen, die frühe und kontinuierliche Lieferung brauchbarer Software, vorrangige Lieferung der Features mit dem größten Kundennutzen, integriertes Risikomanagement, schnelles und häufiges Feedback, regelmäßige Retrospektiven, ein hoher Grad an Testautomatisierung und tägliche Build- und Smoketests sind wichtige Prinzipien agilen Softwareprojektmanagements.*

*Dabei wird der Begriff »iteratives Vorgehen« im Zusammenhang mit agilen Softwareprojekten ebenso häufig wie unterschiedlich gebraucht. Der Erfolg iterativen Vorgehens basiert unserer Erfahrung nach auf der Definition einer Iteration als Timebox – dann lassen sich die eben genannten gängigen Prinzipien agilen Softwareprojektmanagements konsistent und skalierbar auch in Großprojekte integrieren.*

## **1 Iterationen planen und steuern**

### **1.1 Grundsätzlicher Aufbau einer Iteration**

Abbildung 1 zeigt den prinzipiellen Ablauf einer Iteration, aus welchen Elementen sie besteht und wie diese inhaltlich und zeitlich zusammenhängen. Alle Iterationen sind gleichartig aufgebaut und folgen dem selben Schema. Grundsätzlich wird eine Iteration unterteilt in einen Fortschrittsabschnitt und einem Orientierungsabschnitt.

In dem Fortschritts- oder auch Arbeitsabschnitt findet die eigentliche Entwicklung statt. Hier werden Anforderungen abgestimmt und umgesetzt, Lösungen entwickelt etc. (siehe Abb. 1).

Im Gegensatz dazu entwickelt sich das Projekt in dem Orientierungsabschnitt inhaltlich nicht weiter. Hier geht es zunächst einmal um einen systematischen Rückblick auf die in der Iteration getane Arbeit. Was sollte erreicht werden? Was wurde tatsächlich erreicht? Darauf aufbauend werden dann Antworten zu folgenden Fragen gesucht: Was lernen wir daraus? Was können wir zukünftig besser machen? Was bedeuten die Erkenntnisse des Rückblicks für die weitere Planung und das weitere Vorgehen? Ein wichtiger Teil dieses Abschnittes ist also eine *Retrospektive*. Erst danach beginnt dann die nächste Iteration.

Was genau in diesen beiden Abschnitten Fortschritt und Orientierung passiert und zu tun ist, wird im Folgenden erläutert.

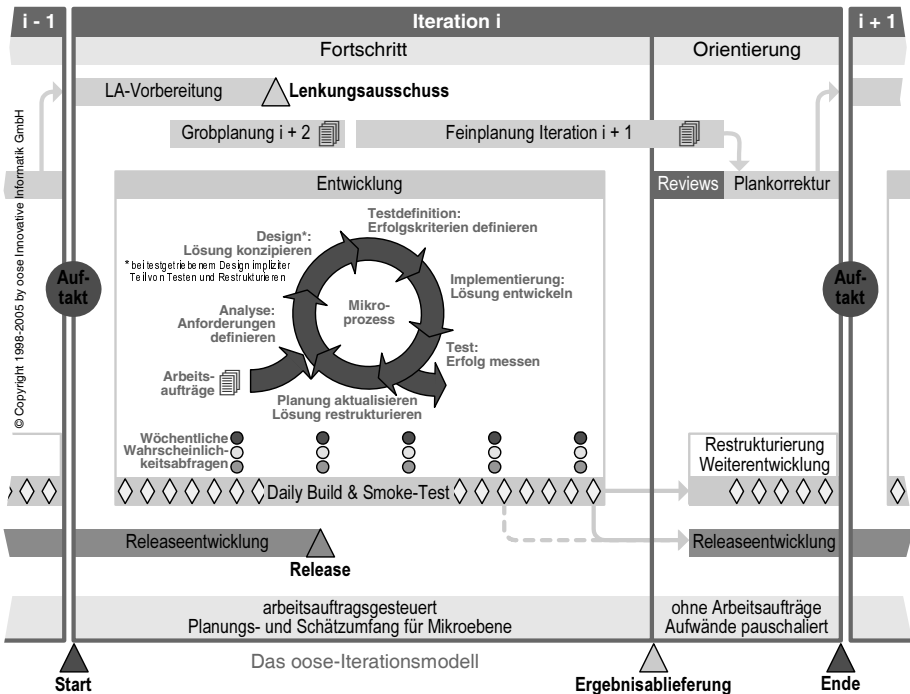


Abb. 1 Prinzipieller Aufbau einer Iteration nach dem oose-Iterationsmodell

## 1.2 Grobplanung i + 2

Anders als bei einigen anderen agilen Verfahren, gehen wir hier davon aus, dass bereits zu Anfang ein oberflächlicher, aber zeitlich vollumfänglicher Projektplan (Makroplan) erarbeitet wird. Die Gründe dafür sind in [Oes 2005] beschrieben.

Aus der Makroplanung geht hervor, welche grundsätzlichen Ergebnisse mit einer Iteration erreicht werden sollen. Es werden iterationspezifische und für den Auftraggeber nachvollziehbare und an seinem geschäftlichen Nutzen orientierte Ziele beschrieben.

Diese Planung ist für die Mikroplanung (Planung einer einzelnen Iteration) so weit herunterzubrechen, dass jeder einzelne Mitarbeiter in einer Iteration genau erfährt, was er zu tun hat und wie die von ihm erwarteten Ergebnisse aussehen. Diese Detaillierung erfolgt in zwei Schritten, der Grobplanung für die jeweils übernächste Iteration (i+2) und der Feinplanung für die jeweils direkt folgende Iteration (i+1). Die Gesamtaufgabe bzw. die Ziele der Iterationen werden dabei in besser überschaubare Einzelaufgaben zerlegt.

Dabei muss jede Teilaufgabe so bemessen sein, dass sie innerhalb einer Iteration realistisch zu bewältigen ist. Arbeitsaufträge, deren Dauer eine Iteration

überschreiten, sind in kleinere Einheiten zu unterteilen. Eine gute Größe ist, pro Projektmitarbeiter ca. 2 – 3 Arbeitsaufträge in einer Iteration vorzusehen.

Ziel der Grobplanung einer Iteration ist es,

- alle Arbeitsaufträge zu identifizieren, d.h. mit einem Namen zu versehen,
- das erwartete Ergebnis jedes einzelnen Arbeitsauftrages ganz grob mit 1 – 2 Sätzen zu benennen,
- den mit dem Arbeitsauftrag verbundenen *Aufwand grob* zu *schätzen*,
- ggf. bereits eine Verantwortliche und weitere Mitarbeiterinnen, die den Arbeitsauftrag umsetzen sollen, zu benennen.

Die Arbeitsaufträge befinden sich am Ende dieser Planungsaktivität im Status *Entwurf*, d.h., der Arbeitsauftrag ist noch unvollständig oder noch nicht abschließend abgestimmt.

Verantwortlich für die Grobplanung ist die Projektleitung, wobei die Planung je nach Projektgröße unterschiedlich organisiert werden kann.

### 1.3 Feinplanung Iteration i + 1

Nachdem Arbeitsaufträge bereits zwei Iterationen im Voraus (Iteration i – 2) identifiziert und grob skizziert wurden, werden diese in der unmittelbar vorangehenden Iteration (i – 1) so weit konkretisiert, dass die Mitarbeiter im Projekt unmittelbar damit arbeiten können. Die Aufgabe ist entsprechend zu präzisieren. Die erwarteten Ergebnisse sind eindeutig, unmissverständlich und soweit möglich objektiv überprüfbar zu beschreiben. Die vorgesehenen Mitarbeiterinnen sind zu benennen, welche Verantwortung bzw. Teilaufgabe sie haben und mit welchem Aufwand sie eingeplant sind. Daraus ergibt sich dann auch der zu planende Gesamtaufwand des Arbeitsauftrages.

**Arbeitsaufträge detaillieren.** Jeder Arbeitsauftrag hat genau einen Ergebnisverantwortlichen und ggf. weitere Mitarbeiter. Der voraussichtlich zu leistende Aufwand sollte soweit möglich von der Ergebnisverantwortlichen, ggf. gemeinsam mit den übrigen Beteiligten, geschätzt werden.

Die Beschreibung eines Arbeitsauftrages umfasst mindestens:

- Geplanter Start- und Endtermin
- Geschätzter Aufwand in Personentagen (PT)
- Für das Ergebnis verantwortliche Person
- Weitere beteiligte Personen und ggf. ihre jeweiligen Rollen oder Aufgaben
- Eine oder mehrere Personen, die nach Ausführung des Arbeitsauftrages den Erfolg bewerten und feststellen (Gutachter)
- Beschreibung des erwarteten Ergebnisses (Form und Inhalt) nach möglichst objektivierbaren Kriterien
- Hinweise zu verwendenden Hilfsmittel, Werkzeuge etc.

Arbeitsaufträge (Task Cards) sollten schriftlich formuliert werden, wobei dies sehr einfach gehandhabt werden kann, beispielsweise mit Karteikarten, auf einem Flipchart oder auf Pinwand-Karten, oder etwas aufwändiger mit speziellen Werkzeugen (Datenbank, Tabellenkalkulationsprogramm, Wiki, Textverarbeitung, spezielle Software).

Wichtig bei der Planung ist, dass die Arbeitsaufträge sinnvoll, realistisch und innerhalb der Iteration umsetzbar sind. Arbeitsaufträge gehen grundsätzlich nicht über eine Iteration hinaus. Größere Aufgaben müssen also in Teilaufgaben zerlegt werden, die innerhalb einer Iteration zu bewältigen sind.

#### 1.4 Iterationsauftakt

Eine Iteration wird durch eine spezielle Auftaktveranstaltung demonstrativ eingeleitet. Die Projektleitung stellt dar, was in der Iteration konkret erreicht werden soll. Die einzelnen Mitarbeiter und Teams erfahren dadurch, wie sich ihre konkreten Aufgaben in das Ganze einfügen und welche Bedeutung ihre Arbeit hat. Außer dem Projektteam sollten auch Benutzer und Vertreter des Auftraggebers dabei sein, damit sie erfahren, welche Ergebnisse sie am Ende der Iteration erwarten dürfen. Außerdem lernen sich somit die verschiedenen beteiligten Parteien frühzeitig kennen bzw. haben regelmäßig Gelegenheit zum persönlichen Gespräch.

**Ziele aktiv kommunizieren.** Beim Auftakt sollen alle Beteiligten auf gemeinsame Ziele eingeschworen werden. Zusätzlich zu den Arbeitsaufträgen kann die Projektleitung auch besondere iterationsspezifische oder teambezogene Ziele kommunizieren. Alle Beteiligten erhalten Gelegenheit, sich zur Planung zu äußern. Mit der Auftaktveranstaltung wird die Planung transparent gemacht, für deren Akzeptanz geworben und die Motivation positiv beeinflusst.

**Druck aufbauen.** Durch die Festlegung von Zielen in Gegenwart des Auftraggebers und der zukünftigen Systembenutzer wird diesen ein Versprechen gegeben: *»In x Wochen, am Ende der Iteration, werden wir folgende Features realisiert haben.«* Durch dieses Versprechen und der damit verbundenen Erwartungshaltung entsteht auch ein wichtiger psychologischer Druck auf das Projektteam.

Es reicht nicht aus, dass sich die Projektleitung und die Auftraggeberin über die Ziele und Teilziele des Projektes einig sind und austauschen. Ziele sind wichtig für die Arbeit des Projektteams und müssen immer wieder allen Projektbeteiligten deutlich gemacht werden. Mitarbeiter, die keine Projektziele kennen, legen sich ersatzweise ihre eigenen Ziele zurecht. Oftmals sind dies Ziele über eine gute Arbeitsqualität. Die meisten Entwicklerinnen haben den Ehrgeiz, technisch und konzeptionell eine exzellente Software zu entwickeln. Dabei können andere mögliche Ziele, wie sie Auftraggeber und Projektleitung abgestimmt haben, beispielsweise Termine, wirtschaftliche Ziele und inhaltlich-fachliche Prioritäten, unbeachtet bleiben.

Ziele landen nicht automatisch beim Team, auch sollte die Kommunikation von Zielen nicht dem Zufall überlassen werden. Ziele sollten daher aktiv vermittelt, d.h. möglicherweise sogar gelehrt werden.

## 1.5 Der Iterations-Mikroprozess

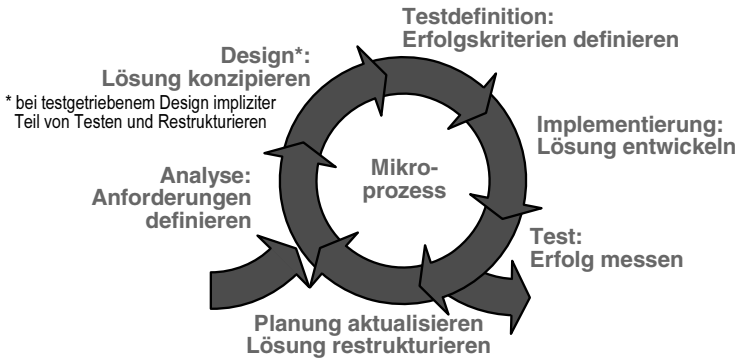
Nach dem Iterationsaufakt werden die vorbereiteten Arbeitsaufträge umgesetzt, d.h. die eigentliche Entwicklung betrieben. Hier finden jetzt alle elementaren Entwicklungsaktivitäten statt:

- Analyse: Anforderungen definieren
- Design: Lösungen konzipieren
- Testdefinition: Erfolgskriterien definieren
- Implementierung: Die Lösungen erstellen
- Test: Den Erfolg messen, die Lösung evaluieren
- Die Evaluierungserkenntnisse verarbeiten, d.h. ggf. die Lösungen restrukturieren und die Planung aktualisieren

**Waschmaschinenmodell.** Wie der Mikroprozess konkret ausgestaltet wird, bleibt den einzelnen Teams und Mitarbeitern überlassen. Das Mikroprozessmodell (auch Waschmaschinenmodell genannt) soll lediglich die prinzipiellen Aktivitäten anzeigen. Es ist nicht notwendig, dass für jeden Arbeitsauftrag dieser Zyklus genau einmal (d.h. eine Umdrehung) durchlaufen wird. Dies hängt vielmehr vom konkreten Arbeitsauftrag ab. Wenn es sinnvoll ist, besteht ein Arbeitsauftrag ausschließlich beispielsweise in der Anforderungsdefinition und -abstimmung oder dient ausschließlich dem Test. Ebenso gut kann es im Rahmen eines Arbeitsauftrages sinnvoll sein, mehrere vollständige Zyklen zu durchlaufen (siehe Abb. 2).

**Testgetriebenes Vorgehen.** Das Mikroprozessmodell kann außerdem auch variiert werden, es zeigt nur eine gängige Abfolge elementarer Aktivitäten. Bei einem testgetriebenen Design beispielsweise wird es die Aktivität Design nur bedingt geben. Bei dieser Vorgehensweise ist das Design quasi ein Abfallprodukt der Testdefinition sowie möglicher Restrukturierungen.

**Testgetriebene Anforderungsdefinition.** Ebenso ist es denkbar, Anforderungsdefinition und Testdefinition als dasselbe aufzufassen. Ein Test ist letztendlich nichts anderes als eine zu erfüllende Anforderung. Statt in natürlicher Sprache ist es prinzipiell auch denkbar, für jede Anforderung sofort den entsprechenden Test zu programmieren. Dann sind diese Tests die Spezifikation der Anforderungen. Auch diese Vorgehensweise ist mit dem hier beschriebenen Iterationsmodell vereinbar. In dem Mikroprozessmodell ist dies lediglich nicht dargestellt, da diese Vorgehensweise in der Praxis selten anzutreffen ist. Meistens werden Anforderungen in natürlicher Sprache sowie mit Hilfe von Dialogentwürfen und Anforderungsmodellen, beispielsweise in UML, definiert.



**Abb. 2** Mikroprozessmodell

Ein wichtiger Erfolgsfaktor für die Softwareentwicklung sind die regelmäßige und objektivierbare Überprüfung des Erreichten und eine starke Ergebnisorientierung. Der zeitliche Abstand vom Erkennen einer Anforderung bis zum Nachweis ihrer akzeptablen Umsetzung soll möglichst kurz sein.

## 1.6 Daily Build and Smoke

**Buids.** Der Aufwand, eine neue lauffähige Version eines Systems zu erstellen, kann projektspezifisch sehr unterschiedlich sein. Es gibt Projekte, die können täglich bzw. quasi permanent neue Systeme bauen, und es gibt andere, die benötigen Wochen oder Monate dafür. Ein wesentlicher Einflussfaktor hierfür ist die Softwarearchitektur. Wenn Testautomatisierung und ein effizienter Build-Prozess nicht von Anfang an geplant und berücksichtigt werden, also insbesondere auch durch die Softwarearchitektur unterstützt werden, kann es sehr schwierig werden, dies nachträglich umzusetzen.

- Ein **Build** ist eine gewöhnlich unvollständige und vorübergehende, aber ausführbare Version eines in Entwicklung befindlichen Systems.
- Ein **Release** ist ein Build, das an den Auftraggeber (bzw. Produktempfänger) ausgeliefert werden kann.
- Ein **Smoke-Test** überprüft, ob ein Build so weit stabil läuft, dass es überhaupt getestet werden kann.

**Daily Build and Smoke-Test.** Im Idealfall läuft der Build-Prozess nach jeder größeren Änderung oder einmal täglich, beispielsweise als so genannte Daily Builds oder Nightly Builds.

Alle Änderungen und Entwicklungen eines Tages werden beim Daily Build and Smoke-Test einmal täglich zu einem neuen Build zusammengeführt und anschlie-

ßend dahingehend getestet, dass es überhaupt stabil läuft. Ein Smoke-Test gibt keine Antworten darauf, ob Tests erfolgreich laufen, sondern darauf, ob das System überhaupt testbar ist und beispielsweise nicht gleich beim Starten wieder abstürzt.

Mit diesem Verfahren wird sichergestellt, dass es jeden Tag eine neue läuffähige Version des Systems gibt. Selbstverständlich enthält dieses noch Fehler, und viele fachliche Tests werden mit negativem Ergebnis beendet. Wesentlich ist jedoch, dass es überhaupt einen objektiv messbaren Stand gibt, d.h. der Projektfortschritt täglich objektiv überprüfbar wird.

Mit den Daily Build and Smoke-Tests wird sichergestellt, dass das, was erstellt wurde, auch nachweisbar und messbar wird. Das Projekt erhält in sehr kurzen Zeitabständen Rückmeldungen zum inhaltlichen Erfolg und Fortschritt. Damit können grundsätzlich auch Meilensteine und die Einhaltung des aktuellen Projektplans überprüft werden. Bezogen auf die Arbeitsaufträge sind damit jedoch keine Aussagen möglich.

## 1.7 Timeboxing

Liegen die Arbeitsaufträge im Plan, d.h., werden sie wie geplant umgesetzt werden können? Wird die Iteration die geplanten Ergebnisse bringen? Diese Fragen sind mit dem bisher Beschriebenen noch nicht beantwortbar. Spätestens am Ende der Iteration lassen sich diese Fragen beantworten (siehe weiter unten im Text). Um jedoch bereits in einer laufenden Iteration Hinweise hierzu zu erhalten, sind weitere Mechanismen notwendig, beispielsweise wöchentliche Wahrscheinlichkeitsabfragen.

Typischerweise werden die geplanten Aufwände vieler Arbeitsaufträge überschritten bzw. die geplanten Termine nicht eingehalten. Manchmal liegt das an einer mangelhaften Aufwandsschätzung, manchmal an der Qualifikation oder Motivation der beteiligten Projektmitarbeiterinnen. In den meisten Fällen werden Termin- und Aufwandsüberschreitungen jedoch durch unvorhersehbare Erkenntnisse oder Situationen verursacht. Es entstehen beispielsweise spontan technische Probleme, es werden auf einmal bisher unentdeckte Abhängigkeiten oder zusätzliche Voraussetzungen identifiziert, Zulieferleistungen oder Entscheidungen der Fachabteilungen verzögern sich etc.

Unabhängig von den Ursachen und Auslösern sind solche Verzögerungen geeignet, das iterative Vorgehen in Frage zu stellen.

Angenommen ein Arbeitsauftrag für ein Team von drei Personen dauert planungsgemäß die gesamte Iteration, also beispielsweise sechs Wochen. Nach ca. drei Wochen stellen die Beteiligten fest, dass der geplante Aufwand nicht ausreicht, der Termin nicht zu halten ist, und melden dies der Projektleitung. Wie soll die Projektleitung damit umgehen?

Die Projektleitung wird über diesen Sachverhalt ebenso wenig erfreut sein wie das Team, andererseits jedoch erleichtert sein, diese Meldung frühzeitig zu erfahren. Leider gibt es immer wieder Projektmitarbeiter, die erst einen Tag vor dem geplanten Endtermin ankommen und Aufwand und Zeit nachfordern. Auch wenn dies beim iterativen Vorgehen tatsächlich seltener vorkommt, wäre eine solche Situation selten noch zu retten.

Sofern die Verzugsmeldung aber frühzeitig eintrifft, fragt die Projektleitung typischerweise: »*Okay, wie viel zusätzlichen Aufwand schätzt ihr, und wann ist der Auftrag voraussichtlich abzuschließen?*«

Es liegt auf der Hand, dass die Festlegung von Iterationsterminen sinnlos ist, wenn ein gewisser Teil der Arbeitsaufträge solche Terminverlängerungen notwendig macht.

- Ein *Meilenstein* definiert einen Termin, zu dem eine Menge von Ergebnissen in einer bestimmten Detaillierung und Vollständigkeit nachprüfbar und formal dokumentiert vorliegen *muss*. Liegen die Ergebnisse zum geplanten Termin nicht vor, wird der Meilenstein verschoben. Ein Meilenstein ist ein Hilfsmittel zur Planung und Überwachung eines Entwicklungsprozesses.
- Eine *Timebox* definiert einen unverrückbaren Zeitrahmen, in dem eine Menge von Ergebnissen in einer bestimmten Detaillierung und Vollständigkeit nachprüfbar und formal dokumentiert vorliegen *soll*. Liegen die Ergebnisse nicht wie geplant vor, werden die offenen Teile in eine nachfolgende Timebox verschoben. Hierzu werden zum geplanten Endtermin die tatsächlich erreichten Ergebnisse bestimmt. Eine Timebox ist ein Hilfsmittel zur Planung und Überwachung eines Entwicklungsprozesses.

Iteratives Vorgehen kann daher nur in Verbindung mit Timeboxing sinnvoll funktionieren. Beim Timeboxing hat der Endtermin eine höhere Priorität als das gelieferte Ergebnis. Das heißt, an dem vereinbarten Termin wird geliefert – egal was!

Sofern die Projektleitung also erfährt, dass ein Arbeitsauftrag nicht wie geplant abgeschlossen werden kann, fragt sie: »*Okay, was könnt ihr dann bis zum vereinbarten Termin liefern?*« Projektleitung und Team vereinbaren also einen reduzierten Lieferumfang.

Dies ist fast immer möglich, es gibt kaum Aufgaben, die es nicht erlauben, innerhalb einer Frist von 4 – 6 Wochen ein (objektiv messbares) Zwischenergebnis zu erzeugen.

Während der ca. 6-wöchigen Entwicklungszeit innerhalb einer Iteration gilt es also rechtzeitig herauszufinden, ob und wie weit die geplanten Ergebnisse termingerecht erreicht werden können.

Meilenstein und Timebox sind zwei sehr ähnliche, aber komplementäre Verfahren, Projekte zu planen, zu steuern und zu überwachen. Auf die plangerechte Umsetzung der Arbeitsaufträge haben beide Mechanismen keinen direkten Einfluss. Die entscheidenden Effekte entstehen indirekt. Beim Meilenstein erfährt

man, wann eine gegebene Menge von Ergebnissen erreicht wurde. Bei der Timebox erfährt man, welche Ergebnisse zu einem gegebenen Zeitpunkt erreicht wurden. In beiden Fällen entstehen Erkenntnisse, die es ermöglichen, bei entsprechenden Planabweichungen die möglichen Ursachen zu prüfen und im Sinne einer besseren Zielerreichung zu intervenieren und zu unterstützen.

Der Vorgang des Planens ist sehr viel wichtiger als der Plan (das Ergebnis selbst).

## 1.8 Wöchentliches Micro-Controlling

Während der Entwicklungszeit melden die Projektmitarbeiter auftragsbezogen ihre Aufwände, d.h. den Stundenverbrauch. Zusätzlich dazu Restaufwandschätzungen einzuholen hat sich in der Praxis als mehr oder weniger sinnlos herausgestellt. Meistens wird der rechnerisch verbleibende Restaufwand als »Schätzwert« genannt, so lange bis ca. 80% des geplanten Aufwandes verbraucht sind. Danach werden die verbleibenden 20% immer wieder als »Rest« gemeldet.

**Wahrscheinlichkeitsabfragen.** Deutlich vorteilhafter ist es stattdessen, wöchentlich bei den Ergebnisverantwortlichen nachzufragen, wie hoch sie die Wahrscheinlichkeit einschätzen, das Ergebnis wie geplant zum vereinbarten Termin zu erreichen. Solange dieser Wert zwischen 75 und 95 % liegt, ist gewöhnlich alles in Ordnung. Wenn der Wert von Woche zu Woche stark schwankt oder plötzlich absackt, kann die Projektleitung dies als Indikator sehen und gezielt nachhaken. Gegebenenfalls muss dann ein neues (Teil-)Ergebnis vereinbart werden.

Mit dieser Vorgehensweise erreichen Sie, dass ausreichend viele (mehr als 90%) Arbeitsaufträge zum vereinbarten Stichtag mit einem definierten Ergebnis abschließen. Das Ergebnis ist vielleicht geringer als ursprünglich geplant, aber es hat (immerhin bzw. stattdessen) einen definierten und überprüfbaren Zwischenstand.

Ein Statusbericht beschreibt den aktuellen Zustand einer in Entwicklung befindlichen Sache, beispielsweise was erreicht wurde, was offen ist, welche Probleme vorliegen und wie es weitergeht. Eine effiziente und vereinfachte Form der Wahrscheinlichkeitsanfrage besteht darin, die Abfrage auf drei verschiedene Status zu reduzieren und diesen die Ampelfarben rot, gelb und grün zuzuordnen. Ein Ampelstatus reduziert die Statusaussage auf die drei möglichen Aussagen:

- *rot, mangelhaft* (unter 50 % Wahrscheinlichkeit)  
Die Entwicklung ist ungenügend, wesentliche Teile wurden nicht erreicht. Der Plan kann nicht mehr eingehalten werden.
- *gelb, unbefriedigend* (50 – 75 % Wahrscheinlichkeit)  
Einige Punkte laufen nicht so wie geplant. Der aktuelle Zustand ist unbefriedigend.

■ *grün, in Ordnung* (über 75 % Wahrscheinlichkeit)

Alles läuft im Wesentlichen wie geplant. Der aktuelle Zustand ist befriedigend oder besser.

Der Vorteil bei der Ampelstatusabfrage besteht darin, dass die Projektleitung in sehr einfacher Form Interventionssignale erhält: Alle Arbeitsaufträge, die im grünen Bereich sind, können einfach weiterlaufen. Es besteht für die Projektleitung kein Handlungsbedarf. Bei den beiden anderen Farben ist dies jeweils ein Signal an die Projektleitung, dass hier interveniert werden sollte. Die Projektleitung erhält also einmal wöchentlich Hinweise darauf, um welche Arbeitsaufträge sie sich kümmern muss und um welche nicht. Bei den gelben Aufträgen geht es, abstrakt formuliert, darum herauszufinden, was die Projektleitung tun kann, um den Auftrag wieder in den grünen Bereich zu bringen. In vielen Fällen gibt es unmittelbare Handlungsmöglichkeiten für die Projektleitung. Wenn beispielsweise eine Zulieferleistung eines Dritten nicht eintrifft, kann die Projektleitung hier den Druck erhöhen oder eskalieren. Wenn ein Projektmitarbeiter bspw. wegen Krankheit ausgefallen ist, kann die Projektleitung ggf. andere Mitarbeiter diesem Auftrag zuordnen.

Diese Maßnahmen können wiederum negative Folge für andere Arbeitsaufträge haben: Wenn also zur Stabilisierung und Unterstützung eines Arbeitsauftrages Mitarbeiter von anderen Arbeitsaufträgen abgezogen werden, ist es sehr wahrscheinlich, dass in der nächsten Woche die anderen Arbeitsaufträge auf gelb oder rot springen. Die Projektleitung wird dies weitgehend berücksichtigen und entsprechend ihrer Risikobewertung und Prioritäten handeln.

Wenn ein Auftrag länger als eine Woche auf Rot steht oder wenn keine unterstützenden Maßnahmen möglich oder sinnvoll erscheinen, es also keine realistische Möglichkeit gibt, einen roten Auftrag wieder in den grünen Bereich zu bringen, dann gibt es eine letzte Möglichkeit: den Zustand zu akzeptieren und mit ihm zu rechnen, d. h. die Realität zumindest angemessen zu berücksichtigen. Die Planung ist anzupassen. Konkret bedeutet dies, dass der betroffene Arbeitsauftrag korrigiert wird. Er wird so weit vereinfacht, dass er ein unter den gegebenen Umständen wieder realistisches Iterationsziel beschreibt. Oder anders ausgedrückt: Der Arbeitsauftrag wird inhaltlich so weit reduziert, dass er in der nächsten Woche wieder im grünen Bereich sein wird. Wie auch immer, die Projektleitung sorgt also dafür, dass alle Arbeitsaufträge innerhalb von 1 – 2 Iterationen wieder in den grünen Bereich kommen.

Obwohl in dem hier beschriebenen Iterationsmodell unsere Iterationen also 6 – 8 Wochen dauern, sind in jedem Fall wöchentliche Rückmeldungszyklen möglich! Durch die Dauer von 6 – 8 Wochen pro Iteration reduzieren wir den Planungsaufwand auf ein (für Großprojekte so wichtiges) handhabbares Maß. Durch die wöchentlichen Wahrscheinlichkeitsanfragen erhalten wir jedoch einen Rückmeldungs- und Regelkreis, wie er beim Extreme Programming auch nicht

kürzer sein könnte. Die wöchentlichen Rückmeldungen führen zu einer Planungsanpassung und Realitätsberücksichtigung auf der Mikroebene, macht dort zwar auch viel Arbeit, ist aber letztendlich noch gut handhabbar. Die Iterationsplanung insgesamt belassen wir im 6 – 8 Wochenrhythmus, weil hier ein deutlich höherer Planungsaufwand herrscht!

Insofern existieren bezüglich des Erkenntnisgewinns von der untersten detailliertesten Ebene bis zur höchsten und abstraktesten Planungsebene bewusste zeitliche Verzögerungen. Nicht jede Erkenntnis über ein Problem oder möglichen Terminverzug (chaotisches Phänomen) wird sofort in der Gesamtplanung berücksichtigt. Stattdessen werden diese Detailerkenntnisse pro Iteration gesammelt und dann einmal pro Iteration in der Gesamtplanung berücksichtigt. Dieses Verfahren ermöglicht es, die Vorteile und Gesetzmäßigkeiten kleiner Teams bezüglich Planung, Steuerung und Kontrolle auch in größeren und sehr großen Projekten operativ anwendbar zu machen.

### **Optionales Instrument: Der Läufer**

Eine Möglichkeit, noch mehr Nutzen mit diesem Iterationsverfahren zu erzielen, ist die Kombination des wöchentlichen Ampelstatus mit dem Läufer-Verfahren. Hierbei wird auf Rotationsbasis im Projekt wöchentlich ein Läufer bestimmt, der alle Teammitglieder an ihrem Arbeitsplatz besucht und den Status ihrer Arbeitsaufträge abfragt. Die Teammitglieder geben eine kurze Rückmeldung, zum Beispiel mit Hilfe des Ampelstatus. Da sie nicht direkt mit dem Projektmanagement kommunizieren, fällt es leichter, auch nicht-technische Probleme anzusprechen, beispielsweise zur Stimmung, Motivation und allgemein zu sozialen Aspekten. Der Läufer agiert daher als Mediator, der Stimmungen aufgreift und die explizite Aufgabe hat, diese vertrauensvoll, d.h. ggf. anonym, an das Projektmanagement weiterzuleiten.

Es hat sich bewährt, den Läufer einmal wöchentlich, beispielsweise Montag vormittags, herumzuschicken und im Anschluss daran, beispielsweise Montags nach dem Mittagessen, den Bericht in einem Projektmanagement-Jour-fixe entgegenzunehmen. Der Zyklus kann aber auch kürzer, länger oder zu anderen Zeiten sein. Es sollte jedoch ein einfacher, leicht einprägsamer wiederkehrender Termin sein. Statt im Rotationsverfahren alle Mitarbeiter mal zum Läufer zu machen, kann auch ein fester Läufer definiert werden. Das Rotationsverfahren ermöglicht jedoch jedem Mitarbeiter einmal, die Gesamtheit des Projektes wahrzunehmen.

## **1.9 Ergebnislieferung**

Bereits zu Beginn der Iteration wurde der Termin für die Ergebnislieferung festgelegt. Zu diesem Termin sind alle Arbeitsaufträge mit einem Ergebnis zu beenden, damit anschließend die Ergebnisüberprüfung in Form von Reviews stattfinden

kann. Dies ist ein entscheidender Zeitpunkt, der offenbart, wie gut die iterative Vorgehensweise und das Timeboxing funktionieren.

Voraussetzung für ein gutes Funktionieren des Timebox-Verfahrens sind Daily Builds and Smoke-Tests und die Wahrscheinlichkeitsabfragen mit bedarfsweiser Arbeitsauftragskorrektur wie zuvor beschrieben. Wenn sowieso täglich ein neues Build entsteht, ist es kein besonderer Aufwand, jetzt einfach das letzte Build aufzugreifen und einer Bewertung (Review) zu unterziehen. Der wichtigste Unterschied zu den täglichen Builds besteht in der Verwendung für eine objektivierbare Bewertung des Projektfortschritts. Neben den Smoke-Tests werden gegen dieses Build also möglichst viele weitere funktionale und nichtfunktionale Tests gefahren.

### 1.10 Arbeitsauftrags-Reviews

Für jeden Arbeitsauftrag wird nach der Ergebnisablieferung ein kurzes Review durchgeführt. Die Review-Beteiligten wurden bereits mit der Formulierung des Arbeitsauftrages festgelegt. Gewöhnlich wird das Review von einer am Arbeitsauftrag unbeteiligten Person durchgeführt, die jedoch das Ergebnis zu beurteilen vermag. Beim Review zeigt sich auch, ob im Arbeitsauftrag präzise genug beschrieben wurde, welche Form, welchen Inhalt und welche Qualität das Ergebnis haben soll und anhand welcher Kriterien das Ergebnis beurteilt werden soll. Resultate der Arbeitsauftrags-Reviews:

- Stichwortartige Beschreibung des tatsächlich erreichten Ergebnisses
- Stichwortartige Beschreibung der offen gebliebenen Punkte
- Ggf. Anmerkungen und Hinweise
- Soweit möglich eine Schätzung des notwendigen Aufwandes, um die offen gebliebenen Punkte zu erledigen

Für die Reviews müssen Sie in Projekten mit über 20 Mitarbeitern insgesamt ca. 2 – 3 Tage vorsehen, bis 20 Mitarbeiter ca. 1 – 2 Tage. Die Reviews sollten möglichst kurz sein und jeweils 10 – 20 Minuten dauern. Sofern dieser Zeitraum nicht ausreicht, wird das Ergebnis zwar entsprechend ungenauer festgehalten, dies jedoch angemerkt.

Arbeitsaufträge und Ergebnisse von besonderer Wichtigkeit oder mit ungewöhnlichen Abweichungen können bedarfsweise später genauer untersucht werden. Hierzu wird jeweils ein separater Arbeitsauftrag vorgesehen, der dann in der folgenden Iteration (als ganz normaler Arbeitsauftrag) bearbeitet wird. Auf diese Weise werden auch alle über diese Bestandsaufnahme hinausgehenden Qualitätssicherungsmaßnahmen eingeplant und bearbeitet. Die Reviews zum Iterationsende dienen primär nicht der Qualitätssicherung, sondern dienen wirklich nur dem Zweck einer Bestandsaufnahme.

Optional können an den Arbeitsauftrags-Reviews auch zukünftige Benutzer oder Vertreterinnen des Auftraggebers teilnehmen. Dadurch verändert sich jedoch in der Regel der Charakter der Reviews, weil die qualitativ-inhaltlich Bewertung meistens die reine Fortschritts- und Bestandsaufnahme überlagert. Auch der Zeitbedarf sowie der Organisationsaufwand (Einladung, Protokoll etc.) sind gewöhnlich erheblich höher.

Zu Abstimmung und Abnahme von Teilergebnissen, wie sie durch einzelne Arbeitsaufträge möglicherweise entstehen, durch die Fachabteilung oder den Auftraggeber, sollten daher vielmehr separate Arbeitsaufträge eingeplant werden, die ganz normal wie andere Arbeitsaufträge auch innerhalb der Iteration abgearbeitet werden. Unproblematisch hingegen ist die Teilnahme der Projektleitung, eines Architekturverantwortlichen oder eines externen technischen Auditors.

### 1.11 Planungskorrektur

Innerhalb der laufenden Iteration hat die Projektleitung bereits die Arbeitsaufträge der nächsten Iteration geplant und vorbereitet. Die gerade abgeschlossenen Reviews führen jedoch zu zahlreichen Erkenntnissen, die eine Anpassung der Planung notwendig oder sinnvoll werden lassen. Die geplanten Arbeitsaufträge sind daher noch mal zu korrigieren. Um den Korrekturaufwand zu minimieren, ist es zweckmäßig, wenn bei der Vorbereitung und Planung der Arbeitsaufträge nicht die *geplanten* Ergebnisse der laufenden Iteration als Basis genommen werden, sondern die zu *erwartenden* Ergebnisse.

Bei der Vorbereitung der Arbeitsaufträge für die Folgeiteration sollten also soweit möglich und vorhersehbar die tatsächlichen Ergebnisse vorweg genommen werden, also erwartet werden, dass nicht alle Arbeitsaufträge vollständig erreicht werden. Dazu ist es hilfreich, wenn die Projektleitung die betroffenen (auftragsausführenden) Mitarbeiterinnen angemessen in die Planung einbezieht bzw. die Planung gemeinsam stattfindet, was hier aber aus Platzgründen nicht näher ausgeführt wird. Trotz aller Voraussicht wird jedoch ein gewisser Anpassungsbedarf nach Abschluss der Reviews übrig bleiben. Dieser ist nun zu leisten. Die aus den Reviews resultierenden Erkenntnisse müssen nun in kürzester Zeit (ca. 2 – 3 Tage) in die Planung einfließen, d. h., die Arbeitsaufträge und Iterationsziele sind anzupassen. Anschließend kann die nächste Iteration beginnen.

Die Projektleitung steht hier unter besonderem Druck. Wenn sie diese Planungskorrektur nicht in der vorgesehenen Zeit leisten kann und dann dem Projektteam mitteilen muss »*Sorry, wir haben die Arbeitsaufträge nicht fertig bekommen, die Iteration fängt deswegen 2 Tage später an*«, wird das Timeboxing nicht funktionieren. Die Glaubwürdigkeit der Projektleitung ist hier von besonderer Bedeutung. Wenn die Projektleitung signalisiert, dass sie sich nicht an die Termine einer Timebox gebunden fühlt, warum sollte sich dann irgendein anderes Projektmitglied daran halten? Planen Sie als Projektleitung also ausrei-

chend Zeit ein, besonders wenn Sie mit dem iterativen Vorgehen noch nicht so geübt sind. Welche Aufwände und Zeitbedarfe welche Beteiligten haben – dies sollte unbedingt im Detail berücksichtigt und geplant werden. Aus Platzgründen kann dies im Rahmen dieses Beitrages nicht weiter ausgeführt werden.

### **1.12 Restrukturierung und Weiterentwicklung**

Während die Reviews laufen und die Projektleitung die Planung anpasst, sollen die Entwickler natürlich nicht die Däumchen drehen. Sie haben in dieser (kurzen) Zeit die Möglichkeit, die entstandenen Ergebnisse zu restrukturieren (Refactoring). Ist zuvor in der Entwicklung vor allem neue Funktionalität entstanden und am Ende möglicherweise unter dem Druck des Abschlusstermines stellenweise mit der heißen Nadel gestrickt worden, so können die Ergebnisse nun konsolidiert und wieder etwas aufgeräumt werden.

Sofern Arbeitsaufträge nicht wie geplant fertig wurden, haben die Beteiligten außerdem die Möglichkeit, an den Aufträgen einfach weiterzuarbeiten. Sollte im Einzelfall weder eine Restrukturierung noch eine Weiterentwicklung sinnvoll sein, haben die Projektmitarbeiter die Möglichkeit, einfach schon mal mit den geplanten Arbeitsaufträgen für die nächste Iteration anzufangen.

### **1.13 Releaseentwicklung**

Täglich und spätestens am Ende der Iteration entsteht ein aktuelles Build von dem in Entwicklung befindlichen System. Zusätzlich hierzu stellt sich die Frage, wie Releases entstehen.

Ein Release ist die Weiterentwicklung eines Build. Im Gegensatz zum Build muss ein Release definierte fachliche Anforderungen erfüllen, d.h. entsprechend getestet werden, in die Zielumgebung überführt werden oder zumindest dafür bereitstehen, zuvor möglicherweise in einer Testumgebung betrieben werden, Abnahme- und FreigabeprozEDUREN durchlaufen und einiges mehr. Manchmal sind Testläufe notwendig, die nur nachts oder am Wochenende möglich sind. Die dafür notwendigen Tätigkeiten können wiederum mit Arbeitsaufträgen geplant und umgesetzt werden, müssen dabei jedoch nicht mehr dem Iterationszyklus folgen. Welche Vorgehensweise hier konkret sinnvoll ist, hängt von dem für die Releaseentwicklung notwendigen Aufwand ab.

Sofern Releases ohne größere besondere Aufwände erstellt werden können, ist es denkbar, dass das letzte Build innerhalb einer Iteration auch gleichzeitig ein Release wird. In den meisten Fällen sind für die Releaseentwicklung aber einige Tage oder Wochen Arbeit notwendig. In diesem Fall wird ein Build aus der iterativen Entwicklung herausgegriffen und Basis für die Releaseentwicklung. Die Entwicklung verzweigt dann in zwei Richtungen. Mit dem einen Entwicklungspfad geht die iterative Entwicklung ganz normal weiter, d.h., es werden schritt-

weise, Build für Build, weitere Funktionalitäten und Features hinzugefügt. In dem anderen Entwicklungspfad werden keine weiteren Funktionalitäten hinzugefügt, außer sie sind zur Stabilisierung oder Freigabe notwendig. Stattdessen werden alle notwendigen Voraussetzungen geschaffen, um die Software einführen bzw. ausliefern zu können.

Die damit verbundenen Arbeitsaufträge und Schritte nach dem gleichen iterativen Schema abzuwickeln, sie also auch Reviews zu unterziehen etc., ist in den meisten Fällen nicht sinnvoll, da der Prozess der Releaseentwicklung bereits eine Bewertung beinhaltet und eine Rückmeldung darüber beinhaltet, wie brauchbar das Build war. Nach der Releasefreigabe gibt es dann weitere Rückmeldungen.

Ausnahmen sind möglicherweise Projekte, die mehrere Monate Zeit benötigen, um aus einem Build ein Release zu machen. Dann sind dies schon eigene kleine Projekte, die wiederum iterativ abgewickelt werden können.

## 2 Abschluss

Das hier beschriebene Timeboxing-Verfahren ist in seiner Grundstruktur seit vielen Jahren weitgehend unverändert (vgl. [Oes 2000]) und wurde aufbauend auf zahlreichen Praxiserfahrungen seitdem vor allem in seinen Details stetig weiterentwickelt und verfeinert. Eingeführt wurde es vor allem in mittleren und großen Projekten mit 10 – 200 Mitarbeitern in zahlreichen namhaften deutschen Unternehmen. Um die praktische Einführung zu unterstützen, wurden verschiedene Dienstleistungen und Produkte entwickelt, unter anderem auch ein Poster und eine Kurzreferenz (Prinzipbild aus Abb. 1) auf einem Becher (siehe [APM-Kurzreferenz]).

### Referenzen

- [APM-Kurzreferenz] Timeboxing-Kurzreferenz als Poster und als Becher:  
<http://www.oose.de/apmbecher.htm>, <http://www.oose.de/umlposter.htm>
- [APM-Seminar] B. Oestereich: Agiles Software-Projektmanagement: Iteratives Vorgehen und Timeboxing in der Praxis. Seminar/Workshop, oose Innovative Informatik GmbH, [www.oose.de/apm](http://www.oose.de/apm)
- [Oes 2000] B. Oestereich: Die Macht des Rhythmus: das OEP-Timepacing-Verfahren. In: Objekt-Spektrum 2/2000.
- [Oes 2005] B. Oestereich: Beweglich bleiben: Möglichkeiten und Grenzen des iterativen Vorgehens. In: Objekt-Spektrum 01/2005.
- [Oes 2006a] B. Oestereich: Der agile Festpreis und andere Preis- und Vertragsmodelle. In: Objekt-Spektrum, 01/2006.
- [Oes 2006b] B. Oestereich: Agiles Projektmanagement. Iterativ vorgehen: Die Kunst des Timeboxing. dpunkt.verlag, Heidelberg, 2006.