

Projektmanagement

Udo Kelter

31.03.2003

Zusammenfassung dieses Lehrmoduls

Bei der Durchführung von Software-Entwicklungsprojekten fallen viele Managementaufgaben an. Dieses Lehrmodul gibt eine Übersicht. Zunächst wird das Projektmanagement von allgemeineren Managementaufgaben abgegrenzt und eine Übersicht über die wichtigsten Tätigkeiten im Projektmanagement gegeben. Am Anfang eines Projekts steht eine Kosten-/Nutzen-Analyse. Mit Hilfe von Netzplänen können dann Termine geplant und Ressourcen reserviert werden. Wir untersuchen ferner, wie die organisatorischen Strukturen, die im Projektmanagement eingesetzt werden, mit Strukturen zusammenhängen, die grobgranular durch Vorgehensmodelle und feingranular durch sogenannte Prozeßprogramme definiert werden.

Vorausgesetzte Lehrmodule:

- obligatorisch: – Vorgehensmodelle
- empfohlen: – Systemanalyse und Systemmodellierung

Stoffumfang in Vorlesungsdoppelstunden: 1.0

Inhaltsverzeichnis

1 Einordnung und Abgrenzung	3
2 Projektmanagementtätigkeiten	4
2.1 Projektdefinition	5
2.2 Projektorganisation	6
2.3 Vorgehensmodell festlegen	6
2.4 Termin- und Ressourcenplanung	7
2.5 Projektüberwachung	7
2.6 Fachliche Klassifizierung der Tätigkeiten	8
3 Kosten-/Nutzenanalyse	9
3.1 Kosten-/Nutzenanalyse aus Sicht des Anwenders	9
3.2 Kosten-/Nutzenanalyse aus Sicht des Softwareherstellers	11
4 Termin- und Ressourcenplanung	12
5 Vorgehensmodelle und Projektplanung	15
Literatur	17
Glossar	17
Index	18

1 Einordnung und Abgrenzung

Größere Softwaresysteme können nur arbeitsteilig von vielen Entwicklern in einem längeren Zeitraum entwickelt bzw. weiterentwickelt werden. Den kompletten Entwicklungsvorgang bezeichnet man oft als **Projekt**. Charakterisiert wird ein Projekt dadurch, daß bestimmte Personen in einem bestimmten Zeitraum bestimmte Ziele erreichen sollen bzw. rückblickend erreicht haben. Bei einem Software-Haus entspricht ein Projekt typischerweise einem Vertrag mit einem Kunden, es ist also explizit dokumentiert. Bei internen Entwicklungsvorhaben bei einem größeren Anwender fehlt manchmal ein derartiger formeller Rahmen oder er ist wesentlich weniger ausgeprägt.

In einem Projekt fallen vielfältige planerische und organisatorische Aufgaben an. Unter **Projektmanagement** versteht man die Gesamtheit der Aufgaben, organisatorischen Strukturen, Techniken und Mittel, die bei der Führung und Abwicklung eines Projekts anfallen bzw. eingesetzt werden.

Wesensmerkmal eines Projekts ist die zeitliche Abgeschlossenheit und Orientierung an einem bestimmten Entwicklungsziel. Manche Entwicklungsaufgaben, z.B. bei langlebigen Systemen die Entgegennahme von Fehlermeldungen, laufende Durchführung von kleineren Änderungen und Anpassungen, Distribution von Zwischenversionen und Patches, Anwenderberatung u.ä., weichen von diesem Wesensmerkmal deutlich ab, so daß man hier i.d.R. nicht von Projekten spricht und auch die Methoden zum Management von Projekten weniger geeignet sind, sondern stattdessen eigene Managementmethoden eingesetzt werden.

Eine Entwicklergruppe wird im Laufe der Zeit oder zeitlich parallel viele Projekte durchführen. Viele Managementaufgaben sind unabhängig von einzelnen Projekten, bspw. die laufende Bereitstellung von Büroraum, Arbeitsplätzen, Rechnernetzen und anderen Infrastrukturen, die Personalfortbildung u.ä.; die Abgrenzung ist allerdings recht unscharf. Die projektunabhängigen Managementaufgaben werden meist nicht zum Projektmanagement gezählt. Oft werden diese Aufgaben auch von anderen Personen bearbeitet als dem Projekt

unmittelbar zugeordneten Mitarbeitern.

Vorgehensmodelle. Vorgehensmodelle definieren die methodisch-/fachliche Struktur bzw. Systematik eines Entwicklungsvorgangs. Insofern stellen sie die fachliche Grundlage für viele Projektmanagementaufgaben dar. Anders gesehen abstrahieren die Vorgehensmodelle weitgehend von Projektmanagementproblemen und sind insofern offen für sehr unterschiedliche Managementstile.

Projektmanagementtätigkeiten fallen in allen Phasen bzw. Entwicklungsstadien an, teilweise sind sie phasenspezifisch. Meist werden die Managementtätigkeiten in Vorgehensmodellen nicht explizit benannt. Eine Ausnahme hiervon bildet die Analyse- und Definitionsphase (im Phasenmodell bzw. entsprechende Phasen in anderen Vorgehensmodellen). Hier wird der Teilbereich Projektplanung oft durch eine eigene Unterphase repräsentiert; Ergebnis dieser Phase sind diverse Planungsdokumente (z.B. ein Netzplan), genauer gesagt deren Erstversion; in den späteren Phasen werden viele Planungsdokumente weiterentwickelt, ohne daß dies durch entsprechende Unterphasen dargestellt würde. Rechtfertigen kann man dies dadurch, daß der Anteil der Projektmanagementtätigkeiten in der Analyse- und Definitionsphase deutlich höher als in den übrigen Phasen ist – wenn man will, kann man fast alle Tätigkeiten in der Analyse- und Definitionsphase als Managementtätigkeiten auffassen, denn letztlich wird die Entscheidung vorbereitet, ob das Projekt wirklich durchgeführt werden soll.

2 Projektmanagementtätigkeiten

Die häufigsten Arten von Tätigkeiten bzw. Tätigkeitsgruppen, die im Rahmen von Projektmanagementaufgaben anfallen, sind:

- Projektdefinition erarbeiten
- Projektorganisation festlegen
- Vorgehensmodell festlegen
- Projektplanung
- Projektüberwachung

- Projektsteuerung
- Projektkontrolle

Man kann diese Tätigkeiten auch als Rollen auffassen. Unterschiedliche Rollen werden manchmal von verschiedenen Personen, manchmal von ein und derselben Person geführt. Im universitären Kontext muß z.B. ein Diplomand in dem Projekt “Diplomarbeit” alle Rollen spielen, während in einer studentischen Projektgruppe einzelne Rollen an einzelne Gruppenmitglieder delegiert sein können.

2.1 Projektdefinition

Ausgangspunkt dieser Tätigkeiten sind generelle Planungsziele des Auftraggebers (gewünscht ist ein System mit bestimmten Funktionen, das ein gegebenes Problem löst, das zu einem bestimmten Zeitpunkt einsatzbereit ist und das bestimmte Maximalkosten nicht überschreitet), ferner ggf. Restriktionen, wonach z.B. vorhandene Hard- und Software weitergenutzt und vorhandenes Personal einbezogen werden muß. Typische Ergebnisse dieser Tätigkeiten sind:

- genauere Beschreibung der Planungsziele und Auflistung von Detailzielen, i.d.R. in Form eines Lastenhefts
- Vorstudie zur Realisierbarkeit
- Risikoanalyse
- Marktanalyse (bei Anbietern von Standardsoftware)
- Aufwands- bzw. Kostenschätzungen
- Kosten-/Nutzenanalyse

Vielfach werden die Ergebnisse dieser Tätigkeiten in Form eines Dokuments fixiert; wichtige und arbeitsintensive Ergebnisse sollten generell schriftlich fixiert werden.

Große Teile der Analyse- und Definitionsphase kann man als Projektdefinitionstätigkeit ansehen, denn letztlich wird die Managemententscheidung, ob das Projekt wirklich stattfinden soll, vorbereitet. Viele Details, die im Pflichtenheft fixiert werden (z.B. die exakte Liste der Entitätstypen und deren Attribute, die durch ein Informationssystem abgedeckt werden sollen), sind irrelevant für die Managemententscheidung. Daher betrachtet man die detaillierte Systemanalyse

und Ausarbeitung des Pflichtenhefts meist nicht mehr als Managementtätigkeit.

2.2 Projektorganisation

Bei der Bildung der Projektorganisation geht es darum, die Personen, die in das Projekt involviert sind, und deren Kompetenzen und Verantwortungen festzulegen.

Die projektspezifischen Organisationsstrukturen müssen oft eingebettet werden in die Firmenstruktur des Auftraggebers bzw. des Auftragnehmers, da diese Strukturen i.a. länger als das Projekt existieren. Es sind viele unterschiedliche Strukturen möglich und sinnvoll, die Auswahl ist abhängig von der Größe des Projekts und anderen Merkmalen. Beispiele von Organisationsstrukturen sind ein Arbeitskreis, eine Matrix-Projektorganisation, eine unabhängige Projektorganisation und eine Projektorganisation mit Stabsstellen.

2.3 Vorgehensmodell festlegen

Die grundlegenden Vorgehensmodelle (Phasenmodell, evolutionäres Modell usw.) sind sehr allgemein und müssen projektspezifisch konkretisiert und an die gegebenen Umstände angepaßt werden. U.a. sind die einzusetzenden Modellierungs- und Programmiersprachen, einzusetzende Werkzeuge usw. festzulegen.

Größere Entwicklergruppen haben meist Methodenhandbücher, in denen die durchzuführenden Entwicklungsschritte weitaus detaillierter als bei den grundlegenden Vorgehensmodellen festgelegt werden. Projektspezifisch sind dann nur noch offene Details zu ergänzen bzw. bei Wahlmöglichkeiten ist eine Auswahl zu treffen.

Die detaillierten Vorgehensmodelle befassen sich in erheblichem Umfang mit Qualitätssicherungsmaßnahmen. Insb. wird festgelegt, welche Teil- und Zwischenergebnisse wann und wie zu prüfen sind¹.

¹Die durchzuführenden Qualitätssicherungsmaßnahmen werden ggf. durch ein firmenweites, also projektübergreifendes Qualitätsmanagement vorgegeben. Ein derartiges Qualitätsmanagement ist für eine ISO9000-Zertifizierung erforderlich.

2.4 Termin- und Ressourcenplanung

Vorgehensmodelle legen für jeden Typ von Arbeitsvorgang fest, wie Arbeitsvorgänge dieses Typs durchzuführen sind; Vorgehensmodelle sind sozusagen eine Planung auf Typebene. Im Gegensatz dazu werden bei der Termin- und Ressourcenplanung einzelne, konkrete Arbeitsvorgänge geplant, d.h. es wird hier sozusagen auf Instanzenebene geplant. Ziel dieser Planung ist es,

- die Entwicklungsdauer bzw. den Fertigstellungszeitpunkt des Systems zu prognostizieren,
- den Bedarf an Personal und anderen Ressourcen zu bestimmten Zeitpunkten zu prognostizieren und mit den verfügbaren Ressourcen abzugleichen.
- die Mitarbeiter in konkreten Zeiträumen konkreten Aufgaben zuzuordnen

Hierzu muß das Gesamtprojekt in einzelne, identifizierbare Arbeitsvorgänge zerlegt werden; für jeden einzelnen Arbeitsvorgang müssen weiter die Bearbeitungsdauer und die erforderlichen Ressourcen geschätzt werden. Reihenfolgebeziehungen zwischen den Arbeitsvorgängen stellt man in Netzplänen dar.

2.5 Projektüberwachung

Kritisch bei den vorstehenden Planungstätigkeiten ist offensichtlich die Präzision, mit der Zeitdauern und Aufwände der einzelnen Arbeitsvorgänge vorausgesagt werden. I.w. können hier nur Erfahrungen mit früheren, vergleichbaren Tätigkeiten herangezogen werden. Selbst bei einer "vernünftigen" Schätzung können durch unvorhergesehene Ereignisse, z.B. die Erkrankung eines Mitarbeiters, Abweichungen von der Planung eintreten. Derartige Abweichungen führen zu folgenden Managementaufgaben:

1. Erkennen von Abweichungen von Termin-, Qualitäts- oder Kostenzielen

2. Gegensteuerung

Die Gegensteuerung entspricht einer inkrementellen Neuplanung und soll hier nicht weiter betrachtet werden. Um Abweichungen erkennen zu können, sind folgende Maßnahmen zu ergreifen:

- Für alle relevanten Zielgrößen sind geeignete Beobachtungs- bzw. Datenerfassungsmechanismen zu installieren. Ideal sind objektive, automatisch durchführbare Messungen, z.B. der automatische Test eines Programms anhand von Testfällen. Teilweise können die interessierenden Daten aber nur in Handarbeit erstellt werden, z.B. bei der Aufschreibung von Arbeitsstunden, ferner kann es hier zu erheblichen Verzögerungen kommen, bis alle relevanten Daten vorliegen.

Sofern Projektüberwachung und Projektdurchführung organisatorisch getrennt sind (z.B. bei einem externen Auftrag), kann das zusätzliche Problem auftreten, daß die Projektüberwachung keinen direkten Zugriff auf die aktuellen Zwischenzustände von Dokumenten bzw. keinen direkten Zugang zu sonstigen Informationsquellen hat.

- Die Zustandsdaten sind regelmäßig zu erfassen.
- Der Projekt-Istzustand ist mit der Planung zu vergleichen.
- Bei Abweichungen ist den entsprechenden Instanzen zu berichten.

Problematisch sind erstens unzutreffende Daten zum Projektzustand, infolge derer die Projektleitung Abweichungen nicht erkennt oder falsch einschätzt, und zweitens zu lange Verzögerungen bei der Erfassung und Auswertung der Daten, durch die die Projektleitung Abweichungen zu spät erkennt. Diesen Gefahren muß durch ein sorgfältig zu planendes Projektberichtswesen entgegengewirkt werden.

2.6 Fachliche Klassifizierung der Tätigkeiten

Die vorstehenden Gruppen von Tätigkeiten erheben keinen Anspruch auf Vollständigkeit; je nach den besonderen Merkmalen eines Projekts kann es diverse weitere Tätigkeiten geben. Es existiert eine große Vielfalt an Management-Problemen und -Techniken.

In den vorstehenden Abschnitten waren die Tätigkeiten anhand der Leitungsfunktionen gruppiert. Eine Alternative besteht darin, nach der erforderlichen fachlichen Qualifikation bzw. den eingesetzten Technologien zu gruppieren.

Viele Tätigkeiten erfordern zwar überwiegend organisatorische Fähigkeiten, gleichzeitig aber auch informatisches Fachwissen. Zu erwähnen sind hier die Systemmodellierung und der Grobentwurf - ohne eine ungefähre Festlegung des Funktionsumfangs eines Systems und seiner Architektur sowie der eingesetzten Implementierungstechnologien kann man keine diesbezüglichen Alternativen auf ihre Kosten hin analysieren und vergleichen. Auf die Modellierung und den Entwurf gehen wir hier aber nicht weiter ein, weil diese Themen ohnehin zum Standardlehrstoff einer Softwaretechnik-Vorlesung gehören.

Fast alle Tätigkeiten erfordern auch die Fähigkeit, mit anderen Personen erfolgreich zu kommunizieren, sowohl mit dem Kunden, i.a. einem Nicht-Informatiker, wie auch mit den Mitgliedern der Entwicklergruppe.

3 Kosten-/Nutzenanalyse

In Abschnitt 2 war wiederholt von Kosten die Rede (u.a. Planung und Überwachung). In diesem Abschnitt gehen wir vertieft auf diesen Aspekt ein, weil er wichtig und in der Informatik-Ausbildung ansonsten eher unterrepräsentiert ist.

Der Sinn einer Software-Entwicklung besteht letztlich darin, einen Nutzen zu erzielen, der die Kosten übersteigt. Allerdings ist schon diese plausible Behauptung interpretationsbedürftig: vom Standpunkt des Anwender bzw. Entwicklers ergeben sich ganz unterschiedliche Bedeutungen.

3.1 Kosten-/Nutzenanalyse aus Sicht des Anwenders

Als Beispiel betrachten wir den typischen Fall, daß ein Anwender ein Informationssystem entwickeln läßt, das in einem Betrieb ein bisher vorhandenes Verfahren ablösen soll. Ein wesentlicher Kostenblock sind

natürlich die reinen Entwicklungskosten. Daneben treten aber noch ganz erhebliche andere Kosten auf:

- es müssen ggf. Rechner (Arbeitsplatzrechner und Server) und Netzwerke beschafft und installiert werden,
- die Mitarbeiter müssen geschult werden,
- ggf. müssen Altdaten konvertiert und nachbearbeitet werden

usw. Nach der Installation und Inbetriebnahme des Systems entstehen während der Nutzungsdauer, die bei größeren Systemen meist 10 oder mehr Jahre beträgt, weitere erhebliche Aufwände. Hierbei können im Prinzip Kosten in allen vorgenannten Kostenarten auftreten. Die direkt auf das Softwaresystem bezogenen Wartungskosten können grob eingeteilt werden in Fehlerbehebungen, Portierungen und Weiterentwicklungen. Diese Folgekosten können ohne weiteres über die Jahre hinweg das Zehnfache der initialen Entwicklungskosten ausmachen.

Weiter treten laufend Kosten auf, die nicht zur Wartung des Softwaresystems gezählt werden können. Beispiele sind die laufende Betreuung der Benutzer, Einarbeitung neuer Mitarbeiter, Administration des Systems (z.B. Datensicherung, Abwehr von Einbruchversuchen), Wartung und Reparatur von Hardware, Datenübertragungskosten und vieles andere mehr. Diese Kosten können fallweise wesentlich größer sein als die direkt dem Softwaresystem zuzuordnenden Kosten und werden häufig völlig unterschätzt. Unter dem Schlagwort *“total cost of ownership”* sind solche Vollkostenberechnungen aber zunehmend ins Bewußtsein gerückt. Sinnvoll sind sie natürlich nur, wenn die Lebensdauer des Systems und die weitere Entwicklung des Unternehmens oder des sonstigen Kontextes halbwegs präzise eingeschätzt werden können.

Der Nutzen eines Softwaresystems kann materiell (finanziell, z.B. Lizenzentnahmen, Einsparungen) oder immateriell (Image, Kundenbindung, -service) sein.

Das vorstehende Beispiel ging von einem betrieblichen Informationssystem aus. Bei anderen Klassen von Software, z.B. eingebetteten Systemen oder Standardprodukten, ergeben sich ähnliche Überlegun-

gen. Festzuhalten ist jedenfalls, daß ein Anwender die Kosten-/Nutzenanalyse wesentlich breiter anlegen muß als ein reiner Entwickler. Aus Sicht eines Anwenders ist Software ein Investitionsgut wie andere (z.B. Gebäude, Maschinen) und unterliegt daher den generellen betriebswirtschaftlichen Kosten-/Nutzenabwägungen.

Weiter verkompliziert wird die Kosten-/Nutzenbetrachtung dadurch, daß die Kosten oft lange vor dem Eintritt des Nutzens anfallen. In solchen Fällen muß zusätzlich die Kapitalverzinsung berücksichtigt werden. Wir kommen hier zu Themenkomplexen, die eher der Betriebswirtschaftslehre zuzuordnen sind und die wir hier aus Zeitgründen nicht behandeln², zumal im universitären Kontext allenfalls "Trockenübungen" denkbar sind.

Ergebnis einer Kosten-/Nutzenanalyse kann sein, daß ein angedachtes Projekt fallengelassen wird, weil die Kosten nicht finanzierbar sind oder die Kosten höher als der Nutzen sind, das Projekt also unwirtschaftlich wäre.

3.2 Kosten-/Nutzenanalyse aus Sicht des Softwareherstellers

Für einen Softwarehersteller ist die Kosten-/Nutzenanalyse meist einfacher als für den Anwender. Ein Hersteller, der ein Softwaresystem für einen Kunden entwickelt, zieht dann einen Nutzen, wenn das vereinbarte Honorar größer als die (internen) Kosten ist. Wir konzentrieren uns i.f. auf die reinen Entwicklungskosten. Die Entwicklung eines Softwaresystems verursacht unterschiedliche Arten von Kosten:

- Personalkosten; hierin enthalten seien neben den Gehalts- und Gehaltsnebenkosten auch die Kosten der Grundausrüstung wie Büromiete, allgemeine Infrastrukturen (z.B. Internet-Anbindung), Sekretariatskosten, Verwaltungskosten usw.; diese Kosten können i.a. nur summarisch erfaßt und anhand von Pauschalsätzen auf Projekte umgelegt werden.

²Wir schlußfolgern hier allerdings, daß eigentlich jeder Informatiker betriebswirtschaftliche Grundkenntnisse haben sollte.

- zusätzliche Kosten für die Entwicklungsumgebung (Hardware und/oder Software)
- projektbezogene Reisekosten, Datenübertragungskosten usw.
- Unteraufträge und Fremdleistungen

Mit Abstand dominierend sind dabei normalerweise die Personalkosten. Von daher konzentrieren sich die Kostenschätzungsverfahren auf die Schätzung dieser Personalkosten. Geschätzt wird der Arbeitsaufwand in Personentagen oder Personenmonaten; die Kosten ergeben sich dann durch Multiplikation mit den Tages- bzw. Monatssätzen.

Kostenschätzungen werden auch benötigt, um zwischen alternativen Realisierungsmöglichkeiten entscheiden zu können, z.B. komplette Neuentwicklung vs. Kauf und Anpassung eines Standardprodukts.

4 Termin- und Ressourcenplanung

Die Kostenschätzungen zielen zunächst nur auf die Finanzierbarkeit bzw. Wirtschaftlichkeit einer Entwicklung. Dabei macht es keinen Unterschied, ob eine Person 10 Monate arbeitet oder 2 Personen jeweils 5 Monate. Wenn das gewünschte System allerdings schon nach 6 Monaten benötigt wird, ist aus Termingründen nur die zweite Alternative gangbar. Daher ist nicht nur der Aufwand, sondern auch die (minimale) Kalenderzeit, die für die Entwicklung benötigt wird, wichtig.

Bei einem gegebenen Aufwand von P Personenmonaten und einer gewünschten Entwicklungsdauer von K Kalendermonaten müssen offensichtlich im Durchschnitt

P/K Personen

parallel arbeiten. Da die meisten Projekte schnell durchgeführt werden müssen, liegt es nahe, die Anzahl der Entwickler, also die Parallelität, zu erhöhen. Dies ist aber nicht beliebig möglich. Hierzu eine Testfrage: Ein Koch braucht 5 Minuten, um 1 Ei zu kochen. Wie lange brauchen 5 Köche für die gleiche Aufgabe?

Viele Entwicklungsaufgaben lassen sich gar nicht oder nicht kostenneutral parallelisieren. Ein wesentlicher Grund hierfür ist, daß

Tätigkeiten aufeinander aufbauen. Hier sei daran erinnert, daß man das Phasenmodell auch so interpretieren kann, daß zunächst vage und abstrakte Systembeschreibungen sukzessive in konkretere Systembeschreibungen und schließlich in ausführbaren Code transformiert werden.

Will man eine Aufgabe dahingehend analysieren, ob und wie weit sie parallel bearbeitet werden kann, muß sie zunächst einmal in Teilaufgaben zerlegt werden. Für die Teilaufgaben ist anzugeben, welche Reihenfolgerestriktionen vorliegen und wie lange sie dauern. Auf dieser Basis kann u.a. die *minimale Gesamtdauer des Projekts* berechnet werden. Ein Modell, in dem sich derartige Strukturierungen von Aufgaben darstellen und analysieren lassen, sind Netzpläne; sie werden u.a. in Lehrmodul [NPT] vorgestellt.

Mit Hilfe eines Netzplans können wir z.B. herausfinden, daß in einem Projekt mit einer minimalen Gesamtdauer von 10 Monaten im 6. bis 8. Monat jeweils 3 Programmierer eingesetzt werden müssen. Nun kann es aber sein, daß, wenn das Projekt am 1.7.2000 startet, in dem fraglichen Zeitraum (1.12.2000 bis 28.2.2001) in unserer konkreten Entwicklergruppe nur 2 Personen zur Verfügung stehen. Wir müssen das Projekt also entweder zeitlich strecken oder zusätzliche Ressourcen beschaffen, z.B. durch Vergabe von Teilaufträgen an externe Entwickler.

Insgesamt können wir also zwei Schritte bei der Termin- und Ressourcenplanung unterscheiden:

1. Zerlegung in Teilaufgaben; Feststellung von Reihenfolgerestriktionen zwischen den Teilaufgaben, Schätzung von Realisierungsaufwand und Minimaldauer der einzelnen Teilaufgaben;

Hieraus ableitbar sind Mindestzeitdauer des ganzen Projekts und Zeiten relativ zum Projektbeginn, zu denen die einzelnen Teilaufgaben beginnen bzw. enden können.

Analysiert wird hier nur die "innere Logik" des Entwicklungsvorgangs.

2. Ressourcenplanung: zusätzlich wird jetzt berücksichtigt, ob ausreichende Ressourcen verfügbar sind. Hierzu muß zunächst ein kon-

kreter Startdatum gewählt werden; damit lassen sich die relativen Kalenderzeitangaben in absolute Datumsangaben umwandeln.

Als zusätzliche Eingabe geht in die Betrachtungen ein, welche Personen (und sonstigen Ressourcen) an den konkreten Terminen verfügbar sind.

Ergebnis dieses Schrittes ist eine Einsatzplanung, in der für konkrete Kalendermonate (oder andere Planungseinheiten) mehr oder minder exakt festgelegt ist, welche Person an welcher Aufgabe arbeitet. Hierbei können recht komplexe Entscheidungssituationen auftreten.

Ziele sind hier einerseits die Einhaltung der gesetzten Termine, andererseits eine gleichmäßige Auslastung der vorhandenen Ressourcen.

Die Granularität der Aufgabenzerlegung kann nicht beliebig klein gewählt werden. Typischerweise wird man Arbeitstage als Einheiten verwenden, eine elementare Teilaufgabe hat dann einen Umfang von mindestens einem Arbeitstag. Kleinere Einheiten lassen sich nicht mehr hinreichend präzise planen und führen zu einem unangemessenen Aufwand für die Planung. Bei sehr großen Projekten kann man aus ebendiesen Gründen eine größere Einheit verwenden, z.B. einen Arbeitsmonat.

Das vorstehende Denkmodell, wonach die Ressourcenplanung nach der Aufwandsschätzung kommt, ist leider nicht ganz realistisch. Alternative Ressourcenzuordnungen sind nämlich nicht notwendigerweise kostenäquivalent. Beispielsweise kann der Aufwand für eine Teilaufgabe erheblich von der Qualifikation und Produktivität des Mitarbeiters abhängen, dem diese Aufgabe zugewiesen wird.

Ferner können bei größeren Projekten solche Teilaufgaben, die weiter in der Zukunft liegen, nicht in alle Details zerlegt werden. Diese Zerlegungsschritte sind später rechtzeitig nachzuholen. Dies bedeutet, daß die Ressourcenplanung beginnt, bevor die Aufgabenzerlegung vollständig ist. Insofern ist die strikte Reihenfolge, die oben unterstellt wurde, ebenfalls unrealistisch.

5 Vorgehensmodelle und Projektplanung

In den Abschnitten 2.3 und 2.4 ist die Relation zwischen Vorgehensmodellen und Projektplanung schon skizziert worden; sie soll hier detaillierter betrachtet werden. Rein äußerlich sieht das Phasenmodell, wenn man es als Sequenz von Teilaufgaben auffaßt, wie ein Netzplan aus. Ferner wird gerade das Phasenmodell als managementorientiert bezeichnet, weil es durch die Meilensteine am Ende der Phasen eine Fortschrittskontrolle ermöglicht.

Trotz dieser Ähnlichkeiten ist das Phasenmodell kein Netzplan, und genereller gesagt sind Vorgehensmodelle nicht unmittelbar Instrumente zur Terminplanung. Netzpläne sind zyklusfrei³ (s. Lehrmodul [NPT]). Nun enthält aber das Phasenmodell in der Praxis bereits potentiell beliebig viele Rücksprünge in frühere Phasen, andere Vorgehensmodelle (insb. das evolutionäre Modell und das Spiralmodell) bestehen sogar auf der obersten Ebene aus einem Zyklus.

Die abstrakten Vorgehensmodelle legen nur sehr grobe Tätigkeitsbereiche fest⁴, lassen die Feinstruktur in diesen Bereichen aber offen. Betrachten wir hierzu die Analysephase, und nehmen wir an, daß im Rahmen der Anforderungsermittlung ein ER-Diagramm entwickelt werden soll. Der Entwicklungsprozeß für dieses ER-Diagramm ist ein Zyklus (s. Abschnitt 5 in [SASM]), in dem in etwa die Schritte

1. Anforderungen ermitteln
2. ER-Diagramm erweitern / korrigieren
3. ER-Diagramm prüfen / besprechen

iteriert werden. Dieser Zyklus ist sozusagen das interne Vorgehensmodell innerhalb der Entwicklungsmethode “ER-Modellierung”.

An dieser Stelle sei an den Begriff Entwicklungsmethode erinnert (vgl. Abschnitt 4 in [SASM]). Diese legt Regeln fest, welche Tätigkeiten bzw. Arbeitsschritte mit dem entstehenden Dokument in welcher

³Ohne diese Annahme ist keine Berechnung der minimalen Durchlaufzeit und anderer interessierender Eigenschaften möglich.

⁴Dies ist untypisch für kommerziell eingesetzte Vorgehensmodelle; diese sind oft sehr detailliert auf Hunderten von Seiten beschrieben.

Reihenfolge auszuführen sind. Hier können zwei Beobachtungen gemacht werden:

- Typisch für fast alle Methoden sind Zyklen der Form: denke nach ... editiere Dokument ... prüfe Dokument ... falls Fehler, beginne von vorne. Die Zahl der Durchläufe durch diesen Zyklus kann unangenehm groß werden, wie vermutlich jeder bestätigen kann, der schon einmal programmiert hat. Von daher sind solche Detail-Vorgehensmodelle nicht als Basis für Terminplanungen geeignet. Sie sind es auch deshalb nicht, weil ein Arbeitsschritt (z.B. das Editieren eines Dokuments) nur wenige Minuten dauern kann, was nach früheren Überlegungen viel zu feingranular wäre.
- Die Regeln beinhalten oft einen erheblichen Ermessensspielraum. In einer bestimmten Situation können mehrere Regeln anwendbar sein, und die Bedingung, wann eine Regel anwendbar ist, kann vage formuliert sein. (Beispiel aus der ER-Modellierung: wenn zwei Entitätstypen viele gemeinsame Attribute haben, sollte für diese ein Generalisierungstyp gebildet werden, sofern dieser aus Anwendersicht sinnvoll ist. Was bedeutet hier “viele gemeinsame Attribute”? “aus Anwendersicht sinnvoll sein”?)

In Netzplänen kann man Alternativen, zwischen denen man in einem Arbeitsschritt auswählen kann, überhaupt nicht ausdrücken. Selbst wenn man eine ausdrucksstärkere Variante von Netzplänen hätte, in der dies ginge, wären immer noch die Grundlagen, auf deren Basis die vagen Bedingungen ausgewertet werden, nicht vorhersehbar und damit nicht planbar. Die eigentliche Logik des Vorgehens würde auch nicht richtig dargestellt.

Zu den allgemeinen Vorgehensmodellen ist noch eine weitere Anmerkung zu machen: sie beschränken sich thematisch weitgehend auf die Entwicklung des eigentlichen Softwareprodukts; Themenkomplexe wie Qualitätssicherung, Anwenderschulung, begleitende Beschaffungen und Installationen werden weitgehend ausgeblendet, müssen aber natürlich dennoch geplant werden. Insofern sind die Vorgehensmodelle ziemlich unvollständig, man kann sie als “Modelle” (!) der ge-

samt den Planungsaktivitäten ansehen, die nur einen ganz bestimmten Ausschnitt der Vorgänge (die fachliche Sicht) vereinfacht darstellen.

Planung feingranularer Entwicklungsprozesse. Feingranulare Entwicklungsprozesse sind nicht im Sinne von Aufwänden oder Terminen planbar. Dennoch liegt es nahe, sie in dem Sinne zu planen, daß kein notwendiger Schritt (versehentlich) ausgelassen wird, weil sonst irgendein Defekt im System oder sonstiger Schaden auftritt. Der Sinn solcher Planungen ist die *Qualitätssicherung*: letztlich bedroht das Auslassen eines wichtigen Schrittes die Qualität des entstehenden Systems.

Will man den Entwicklungsprozeß rechnergestützt überwachen, muß man das feingranulare Vorgehensmodell in einer Sprache bzw. einem Formalismus notieren, der eine exakte Bedeutung hat. Ein derartiger Formalismus muß wesentlich ausdrucksstärker sein als Netzpläne. Vorgeschlagen worden sind zu diesem Zweck u.a. Formalismen auf Basis erweiterter Petri-Netze und KI-Sprachen (insb. Prolog). Statt von Vorgehensmodellen spricht man, weil die Formalismen in etwa Programmiersprachen entsprechen, von *Prozeßprogrammen* oder auch Prozeßskripten und nennt den ganzen Bereich *Prozeßmodellierung*; Prozeß steht hier für Entwicklungsprozeß.

Literatur

[NPT] Kelter, U.: Lehrmodul "Netzplantechnik"; 2001/12

[SASM] Kelter, U.: Lehrmodul "Systemanalyse und Systemmodellierung"; 2002/01

[ZM] Kelter, U.: Lehrmodul "Zeitmanagement"; 2002/04

Glossar

Projekt: Entwicklungsvorgang, bei dem bestimmte Personen in einem definierten Zeitraum bestimmte Ziele erreichen sollen

Projektmanagement: Gesamtheit der Aufgaben, organisatorischen Strukturen, Techniken und Mittel, die bei der Führung und Abwicklung eines Projekts anfallen bzw. eingesetzt werden

Projektorganisation: definiert die Personen, die in ein Projekt involviert sind, und deren Kompetenzen und Verantwortungen

Projektüberwachung: Überwachung von Termin-, Qualitäts- oder Kostenzielen, d.h. Erkennen von Abweichungen und anschließendes Gegensteuern

Index

- Analysephase, 4, 5
- Aufgabe
 - Zerlegung, 13
 - Granularität, 14
- Aufwandsschätzung, 5
- Einsatzplanung, 14
- Kommunikation, 9
- Kosten
 - ~/Nutzenanalyse, 5, 9, 11
 - ~schätzung, 12
 - Entwicklungs~, 9, 11
 - Personal~, 11
- Management
 - Tätigkeiten, 4
 - projektunabhängiges, 3
- Methodenhandbuch, 6
- Netzplan, 13
- Parallelisierung, 12
- Pflichtenheft, 5
- Phasenmodell, 15
 - Analysephase, *siehe Analysephase*
- Projekt, 3, 17
 - Berichtswesen, 8
 - Definition, 5
 - Istzustand, 8
 - Mindestzeitdauer, 13
 - Organisation, 6
 - Steuerung, 7
 - Überwachung, 7, 8
- Projektmanagement, 3, 17
- Projektorganisation, 18
- Projektplanung, 15
 - inkrementelle, 14
- Projektüberwachung, 18
- Prozeßmodellierung, 17
- Prozeßprogramm, 17
- Qualitätssicherung, 6, 17
- Ressourcenplanung, 7, 12, 13
- Systemmodellierung, 9
- Terminplanung, 7, 12
- Vorgehensmodell, 4, 6
 - abstraktes, 15
 - feingranulares, 6, 16, 17
- Vorstudie, 5
- Wartung, 10