

September 2003

Vorgehensmodell zur Durchführung von IT-Projekten unter expliziter Einbeziehung der Kundensicht

Kai Marquardt

BEIT Systemhaus GmbH, KMarquardt@beit.de

Follow this and additional works at: <http://aisel.aisnet.org/wi2003>

Recommended Citation

Marquardt, Kai, "Vorgehensmodell zur Durchführung von IT-Projekten unter expliziter Einbeziehung der Kundensicht" (2003).
Wirtschaftsinformatik Proceedings 2003. 99.
<http://aisel.aisnet.org/wi2003/99>

This material is brought to you by the Wirtschaftsinformatik at AIS Electronic Library (AISeL). It has been accepted for inclusion in Wirtschaftsinformatik Proceedings 2003 by an authorized administrator of AIS Electronic Library (AISeL). For more information, please contact elibrary@aisnet.org.

In: Uhr, Wolfgang, Esswein, Werner & Schoop, Eric (Hg.) 2003. *Wirtschaftsinformatik 2003: Medien - Märkte - Mobilität*, 2 Bde. Heidelberg: Physica-Verlag

ISBN: 3-7908-0111-9 (Band 1)

ISBN: 3-7908-0116-X (Band 2)

© Physica-Verlag Heidelberg 2003

Vorgehensmodell zur Durchführung von IT-Projekten unter expliziter Einbeziehung der Kundensicht

Kai Marquardt

BEIT Systemhaus GmbH

Zusammenfassung: Die Durchführung von IT-Projekten beschränkt sich nicht darauf, Anforderungen aufzunehmen und dann das fertig entwickelte System abzuliefern (Black-Box-Syndrom). IT hat in Unternehmen einen Stellenwert eingenommen, der sogar eine Anpassung von Unternehmensprozessen während der Durchführung von IT-Projekten beinhaltet. Dementsprechend muss während dieser IT-Projekte die Zusammenarbeit zwischen der Auftraggeber- und Auftragnehmerseite intensiviert werden. Einerseits will bzw. muss die Auftraggeberseite aktiv in das Projektvorgehen integriert werden, um Entscheidungen treffen bzw. bewerten und den Projektfortschritt unter Qualitäts-, Kosten- und Zeitaspekten beurteilen zu können. Wichtig ist dabei, dass Zwischen- und Endergebnisse zum richtigen Zeitpunkt verifiziert und validiert werden können. Andererseits bilden IT-Projekte Teilprojekte von im Auftraggeber-Unternehmen durchzuführenden Projekten, die es mit dem Vorgehen im Auftragnehmer-Unternehmen abzustimmen gilt. Zur Unterstützung dieser Aspekte soll das Vorgehensmodell unter expliziter Einbeziehung der Kundensicht beitragen.

Schlüsselworte: IT-Projekte, Vorgehensmodelle, Kundenorientierung

1 Einführung

Zur Durchführung von IT-Projekten sollte ein Vorgehensmodell eingesetzt werden. Es bildet die Beschreibung der koordinierten Vorgehensweise bei der Abwicklung des Projekts und definiert sowohl den Input, der zur Abwicklung einer Aktivität notwendig ist, als auch den Output, der als Ergebnis einer Aktivität produziert wird. Weiterhin wird eine feste Zuordnung von Rollen, die die jeweiligen Aktivitäten ausüben, vorgenommen. [vgl. Vers02]

Viele Vorgehensmodelle aus der einschlägigen Literatur setzen ihren Schwerpunkt in der Ausgestaltung der entwicklungsseitigen Phasen und dementsprechend bei der Berücksichtigung der auftragnehmerseitigen Rollen. Eine explizite Ausgestaltung der auftraggeberseitigen Tätigkeiten findet nicht statt. Im

Folgenden sollen deshalb zunächst einige Vorgehensmodelle vorgestellt und in Bezug auf ihre Berücksichtigung der Auftraggeberseite beschrieben werden, bevor das Vorgehensmodell unter expliziter Einbeziehung der Kundensicht vorgestellt wird.¹ Um dies vornehmen zu können, muss zunächst definiert werden, welche Rollen von Auftragnehmer- und Auftraggeberseite an IT-Projekten teilnehmen können.

Auftragnehmerseite

Von Auftragnehmerseite können die Rollen Auftragnehmer, Auftragnehmer-Projektleiter, Phasenverantwortlicher und Arbeitspaketverantwortlicher unterschieden werden.

Auftragnehmer

Der Auftragnehmer ist für die Erfüllung des Entwicklungsauftrags rechtlich verantwortlich. Er hat in der Regel keine ausführenden Tätigkeiten in einem Projekt.

Auftragnehmer-Projektleiter

Der Auftragnehmer-Projektleiter ist für die Durchführung des Projekts verantwortlich. Er führt damit das Projektmanagement aus. Hierzu zählt die Überwachung der Qualität, des Kosten- und Zeitrahmens.

Phasenverantwortlicher

Für die Phasen des Vorgehensmodells können zur Unterstützung des Projektleiters Verantwortliche eingesetzt werden. Sie sind in der Regel Spezialisten für die Durchführung der jeweiligen Phase und bringen damit ihre Kompetenz ein.

Arbeitspaketverantwortlicher

Der Arbeitspaketverantwortliche führt die ihm zugewiesenen Aufgaben aus. Das können Analyse- oder auch Entwicklungstätigkeiten sein. Er ist dem Projektleiter und einem eventuell vorhandenen Phasenverantwortlichen organisatorisch unterstellt.

¹ Im Folgenden wird von Auftraggeber- und Auftragnehmerseite gesprochen. Um den Bezug zur Kundensicht zu bekommen, ist zu definieren, dass der Kunde des Auftragnehmers der Auftraggeber ist. Jedoch kann der Auftragnehmer, wenn er Tätigkeiten aus dem Projekt an andere Unternehmen weitergibt, auch wiederum Kunde dieses Unternehmens sein.

Auftraggeberseite

Von der Auftraggeberseite können die Rollen Auftraggeber, Auftraggeber-Projektleiter, Initiator und Endbenutzer definiert werden.

Auftraggeber

Der Auftraggeber initiiert das Entwicklungsvorhaben und stellt die finanziellen Mittel zur Verfügung. Auf welcher Ebene er ein Projektcontrolling durchführt, entscheidet er selbst. Der Rahmen bewegt sich von einem reinen Überblick bis zu einzelnen detaillierten Betrachtungen.

Der Auftraggeber muss nicht Initiator (siehe unten) sein. Entscheidend ist, dass der Auftraggeber die Entscheidungen (vor allem finanzieller Art) treffen kann.

Auftraggeber-Projektleiter

Der Auftraggeber-Projektleiter treibt das Projekt von Auftraggeberseite voran. Er ist in der Lage, Entscheidungen für die Weiterführung des Projekts zu treffen. Ab einer gewissen „Schwere“ der Entscheidung wird er sich mit dem Auftraggeber abstimmen müssen.

Initiator

Bei größeren Firmen kommt der Anstoß für ein Projekt üblicherweise von höheren Angestellten, evtl. von der Firmenleitung. Sie beabsichtigen damit meist eine Kosteneinsparung, Beschleunigung bestimmter Arbeitsabläufe oder eine Flexibilisierung der Unternehmensprozesse. [Suhl+02] Ihre Einbindung in das Projekt ist für die Akzeptanz des einzuführenden Systems von entscheidender Bedeutung.

Endbenutzer

Der Endbenutzer wird später mit dem System arbeiten. Der Großteil der Anforderungen wird von ihm gestellt. Seine Akzeptanz ist ebenfalls von großer Bedeutung.

2 Vorgehensmodelle der Systementwicklung

Im Laufe der Zeit wurden einige Vorgehensmodelle entwickelt. Im Folgenden sollen

- das strenge Phasenmodell,
- das Wasserfall-Modell,

- das V-Modell,
- das Prototypen-Modell,
- das evolutionäre/inkrementelle Modell,
- das objektorientierte Modell,
- das nebenläufige Modell,
- das Spiralmodell und
- das überlappende Phasenmodell

kurz vorgestellt werden. Dabei beziehen sich die Darstellungen vor allem auf [Balz98] und [Suhl+02]. Neben den Modell-Eigenschaften soll die Möglichkeit der Einbindung der Auftraggeberseite im Projektablauf beschrieben werden.

Strenges Phasenmodell

In Anlehnung an allgemeine Problemlösungsprozesse (Problemdurchdringung, Lösungsplan, Lösung) wurden sequentielle Lösungsmodelle entwickelt. Sie enthalten im Wesentlichen die Phasen Analyse, Design, Implementierung und Einführung.

Der Start einer Phase setzt die Beendigung der vorangehenden Phase und damit das vollständige Vorliegen aller dort erstellten Entwicklungsdokumente voraus. Der Benutzer wird zu Projektbeginn über seine Anforderungen befragt und am Ende mit dem fertigen Produkt konfrontiert. [Suhl+02] Es erfolgt in der Regel keine Aussage darüber, wer welche Aktivitäten ausführt, wie die Ergebnisse zusammenhängen und wie phasenübergreifende Maßnahmen durchzuführen sind.

Wasserfallmodell

Das Wasserfallmodell ist eine Weiterentwicklung des strengen Phasenmodells. Es soll die Nachteile ausgleichen, indem explizit Rückkopplungen zu vorhergehenden Phasen eingeführt werden. Es legt trotzdem fest, dass Software in sukzessiven Stufen entwickelt wird. Dabei ist eine Phase in der richtigen Reihenfolge und vollständig durchzuführen, bevor in die nächste eingestiegen werden darf.

Eine Benutzerbeteiligung ist, wie beim Wasserfallmodell, nur in der Definitionsphase vorgesehen, anschließend erfolgen der Entwurf und die Implementierung ohne Beteiligung der Auftraggeberseite. [vgl. Balz98]

Das Wasserfallmodell ist in irgendeiner Form in vielen Modellen wiederzufinden (z.B. im V-Modell und Rational Unified Process²).

² Siehe Abschnitt „Objektorientiertes Modell“.

V-Modell

Das V-Modell ist wiederum eine Weiterentwicklung des Wasserfallmodells. Es legt die Aktivitäten und Produkte³ des Entwicklungs- und Pflegeprozesses fest. Außerdem werden die Produktzustände und die logischen Abhängigkeiten zwischen Aktivitäten und Produkten dargestellt. Aktivitäten und Produkte werden auf verschiedenen Abstraktionsebenen beschrieben.

Das V-Modell ist in vier Submodelle gegliedert: Systemerstellung (SE), Qualitätssicherung (QS), Konfigurationsmanagement (KM) und Projektmanagement (PM).

Das V-Modell ist von Auftraggeberseite entwickelt worden, hat aber das Problem, dass Projekte zu großen Teilen an die Auftragnehmerseite abgegeben werden.

In den Rollen des V-Modells⁴ wird von Auftraggeberseite nur der Anwender aufgeführt. Eine Beschreibung der Einbindung bzw. der Aufgaben des Auftraggebers und eines eventuellen Initiators im Entwicklungsprozess erfolgt nicht.

In der Systementwicklung ist die Einbindung der Auftraggeberseite (durch den Endanwender) nur im Anfangs- und Endstadium vorhanden. [vgl. V-Mod97] Das bedeutet, dass der Endanwender nur bei der Anforderungsdefinition und der Systemeinführung einbezogen wird, was zur Folge haben kann, dass eventuelle Fehler erst sehr spät im Entwicklungsprozess erkannt werden. [vgl. Wall01]

Prototypen-Modell

Ein weit verbreiteter Lösungsansatz für das Dilemma der Sequentialität ist die Erstellung von Prototypen⁵.

Das Prototypen-Modell unterstützt auf systematische Weise die frühzeitige Erstellung ablauffähiger Modelle (Prototypen) des zukünftigen Software-Produkts, um die Umsetzung von Anforderungen und Entwürfen in Software zu demonstrieren und mit ihnen zu experimentieren. [Balz98]

Das Konzept kann bzw. sollte in andere Modelle integriert werden. Es findet eine starke Rückkopplung mit der Auftraggeberseite statt. Dabei sind die Zeitpunkte und die Ausgestaltung der Abstimmung zwischen der Auftragnehmer- und Auftraggeberseite immer nur bis zur nächsten Vorstellung eines Prototyps definiert oder werden vom übergeordneten Modell vorgegeben.

Evolutionäres/inkrementelles Modell

Ausgangspunkt beim evolutionären Modell sind die Kern- oder Mussanforderungen des Auftraggebers. Diese Anforderungen definieren einen

³ Ein Produkt ist das Ergebnis einer Aktivität. [Balz98]

⁴ Es sind im V-Modell insgesamt 25 verschiedene Rollen definiert. [vgl. V-Mod97]

⁵ Dabei können verschiedene Prototypen-Arten unterschieden werden. [vgl. Balz98]

Produktkern. Nur dieser Produktkern wird anschließend entworfen und implementiert. Das Kernsystem, auch Nullversion genannt, wird an die Auftraggeberseite ausgeliefert. Die Auftraggeberseite sammelt Erfahrungen mit dieser Nullversion und ermittelt daraus ihre Produkthanforderungen für eine erweiterte Version. Die Nullversion wird anschließend um die neuen Anforderungen ergänzt. Die neue Produktversion wird eingesetzt. Anhand der gewonnenen Erfahrungen werden wieder neue Anforderungen aufgestellt.

Das Software-Produkt wird allmählich und stufenweise entwickelt, gesteuert durch die Erfahrungen, die die Auftraggeberseite (vor allem der Endbenutzer) mit dem Produkt macht. Die Auftraggeberseite erhält somit in kurzen Zeitabständen einsatzfähige Produkte.

Bei der inkrementellen Entwicklung werden die Anforderungen an das zu entwickelnde Produkt möglichst vollständig erfasst und modelliert. Analog zur evolutionären Entwicklung wird jedoch nur ein Teil der Anforderungen entworfen und implementiert. Die Auftraggeberseite erhält in kurzer Zeit ein einsatzfähiges System. Anschließend wird die nächste Ausbaustufe realisiert, wobei Erfahrungen der Auftraggeberseite mit der laufenden Version berücksichtigt werden.

Da bei der inkrementellen Entwicklung bereits ein vollständiges Analysemodell entwickelt wird, ist sichergestellt, dass die inkrementellen Erweiterungen zum bisherigen System passen. Das muss bei der evolutionären Entwicklung nicht der Fall sein. [Balz98]

Objektorientiertes Modell

Die Methoden der objektorientierten Softwareentwicklung gehen in erster Linie auf Booch, Rumbaugh und Jacobson zurück und bilden mit der Unified Modeling Language (UML) einen Quasi-Standard.

Die UML ist in erster Linie die Beschreibung einer einheitlichen Notation und Semantik sowie die Definition eines Metamodells. Die Beschreibung einer Entwicklungsmethode gehört nicht direkt dazu, sie wurde erst Anfang 1999 mit der Publikation des Buches [Jaco+99], dem sogenannten Unified Process (UP), nachgeliefert.

Der Object Engineering Process (OEP) ist eine spezielle, praxisbewährte und konkrete Ausprägung des Unified Process. Eine andere konkrete Ausprägung des Unified Process ist der Rational Unified Process (RUP), der ein maßgeschneidertes Vorgehen speziell für die Werkzeuge der Firma Rational darstellt (Rose, Requisite Pro etc.).

Das Vorgehen des OEP unterscheidet die Analyse und das Design. Mit (objektorientierter) Analyse werden alle Aktivitäten im Rahmen des Softwareentwicklungsprozesses bezeichnet, die der Ermittlung, Klärung und Beschreibung der Anforderungen an das System dienen (d.h. die Klärung, was das System leisten soll). Mit (objektorientiertem) Design werden alle Aktivitäten im

Rahmen des Softwareentwicklungsprozesses bezeichnet, mit denen ein Modell logisch und physisch strukturiert wird. Zusätzlich dient es der Beschreibung, wie das System die in der Analyse beschriebenen Anforderungen erfüllt. [Oest01] Damit bezeichnet die Analyse mehr die Phase Sollkonzept und das Design mehr die Phase Systementwurf des später beschriebenen Vorgehensmodells. Zwischen Ist- und Soll-Zustand wird nicht explizit unterschieden.

Der Rational Unified Process (RUP) ist ein iterativer Ansatz, bei dem die Phasen Konzeptionalisierung, Entwurf, Konstruktion und Übergang durchgeführt werden. Die Aufwände, die in jeder Phase in den einzelnen Iterationen anfallen sowie auch die Anzahl der Iterationen, variieren je nach Notwendigkeit.

Beim RUP werden die Rollen von Entwicklerseite definiert. Die Aufnahme der Anforderungen wird somit nur von der Auftragnehmerseite beleuchtet. [vgl. Vers02]

Ein wesentlicher Vorteil einer objektorientierten Software-Entwicklung besteht darin, dass die Wiederverwendung von Klassen und Klassenhierarchien durch Komposition, Vererbung und Polymorphie unterstützt wird und sich so der Fokus weg von der Eigenentwicklung und hin zur Wiederverwendung verschiebt.

Das Modell ist gut kombinierbar mit dem evolutionären und inkrementellen Modell sowie dem Prototypen-Modell, da Erweiterungen und Änderungen mit der objektorientierten Technik gut durchführbar sind. [Balz98] Das Modell selber beinhaltet keine Berücksichtigung der auftraggeberseitigen Tätigkeiten, sondern fokussiert auf die Wiederverwendung von Systemteilen und ist damit entwicklerlastig. Erst durch die Kombination mit den oben genannten Modellen wird eine Auftraggeberseite berücksichtigt.

Nebenläufiges Modell

Im nebenläufigen Modell sind möglichst viele Aktivitäten zu parallelisieren oder stark zu überlappen. Dadurch wird das auf Problemlösung gerichtete Zusammenarbeiten der am Entwicklungsprozess beteiligten Personen gefördert. Wie [Balz98] feststellt, besteht das Risiko, dass grundlegende und kritische Entscheidungen zu spät oder nicht abgestimmt getroffen werden und dadurch Iterationen notwendig werden.

Spiralmodell

Beim Spiralmodell handelt es sich um einen Ansatz, der evolutionäres Prototyping sowie die Integration von Simulation, Benchmarking und anderen Risiko reduzierenden Techniken explizit unterstützt. Der Projektfortschritt wird hier durch eine Spirale dargestellt. Der Abstand zum Mittelpunkt ist ein Maß für die kumulierten Kosten, die jeweils bis zum dargestellten Zeitpunkt entstanden sind. Der gesamte Projektablauf ist in mehrere Spiralumläufe gegliedert, wobei jeweils

vier grundsätzliche Schritte, symbolisiert durch vier Quadrante, durchlaufen werden.

Im ersten Quadrant der Spirale werden die Ziele des (Teil-)Produkts festgelegt (Performance, Funktionalität etc.), Randbedingungen festgestellt und Implementierungsalternativen ermittelt. Anschließend werden die Alternativen im zweiten Quadrant bewertet, insbesondere im Hinblick auf mögliche Risiken. Ein Prototyp der gewählten Lösungsalternative wird implementiert. Im dritten Quadrant werden die verbleibenden Risikoquellen durch Simulation, Benchmarking etc. eingegrenzt. Genügt der Prototyp insgesamt den Anforderungen nicht, wird die Erstellung eines weiteren Prototypen vorbereitet, ansonsten kann das Produkt selbst weiterentwickelt werden. Im vierten Quadrant werden die nächsten Phasen geplant. [vgl. Boeh88]

Nach [Balz98] besteht der größte Nachteil im hohen Managementaufwand, da oft neue Entscheidungen über den weiteren Prozessverlauf getroffen werden müssen. Eine explizite Einbindung der Auftraggeberseite in das Modell wird nicht beschrieben.

Überlappendes Phasenmodell

Bei der Durchführung eines Projekts sind Sprünge notwendig, die weiter als zur vorangegangenen Phase zurückgehen (vgl. Wasserfallmodell) bzw. vorwärts gerichtet sind. Daher werden im überlappenden Phasenmodell die Phasen explizit parallel modelliert, jedoch mit wechselnder Intensität während des Projektablaufs.

Jede Phase ist durch bestimmte typische Tätigkeiten gekennzeichnet. Beginn und Ende der Phasen stimmen im Allgemeinen nicht mit den vom Projektmanagement zu planenden Meilensteinen überein. Meilensteine liegen (unter anderem) an den Punkten, an denen konsistente Ergebnisse der jeweiligen Phase vorliegen und der Auftraggeberseite vorgestellt werden (z.B. Sollkonzept, Systementwurf). Üblicherweise folgen darauf noch mehrere Änderungszyklen, deren Tätigkeiten teilweise wiederum den vorhergehenden Phasen zuzuordnen sind und die durch eigene Meilensteine gekennzeichnet sein können. Dementsprechend ist es möglich, dass sich zu einem bestimmten Zeitpunkt verschiedene Teile des Systems in unterschiedlichen Phasen befinden (Abbildung 1 und 2).

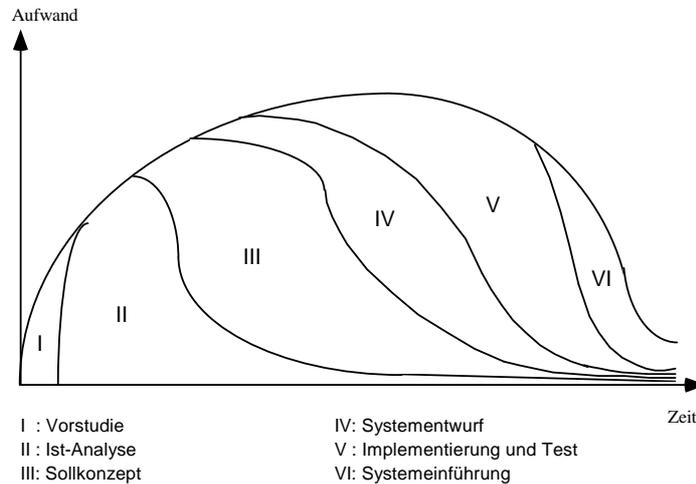


Abbildung 1: Überlappendes Phasenmodell bei Individualentwicklung

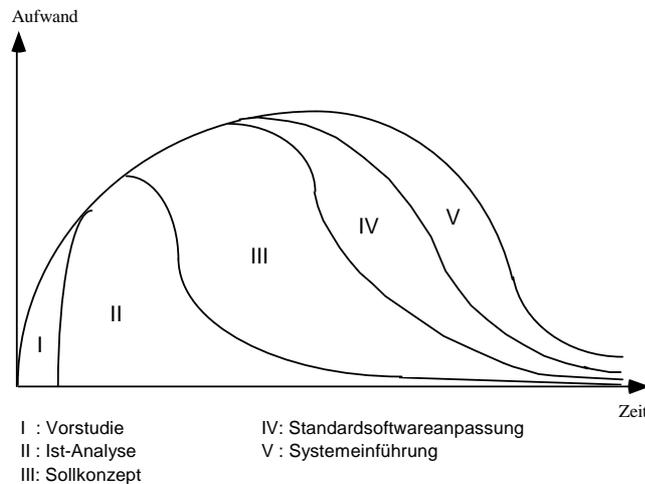


Abbildung 2: Überlappendes Phasenmodell bei Entscheidung für Standardsoftware

Nach einer Vorstudie wird in der Ist-Analyse der aktuelle Stand des Systems erfasst und analysiert, wobei die gefundenen Schwachstellen als Verbesserungspotentiale zu verstehen sind. Die dritte Phase legt ein Konzept fest, das die Schwachstellen beseitigt. Wichtig ist, dass die unternehmerischen Ziele im Vordergrund stehen. Daher sind die Phasen II und III eher organisations- als IT-orientiert. Betriebliche Nutzenpotentiale der Lösung müssen der Auftraggeberseite im Rahmen des Sollkonzepts eindeutig vermittelt werden.

Die Vorgehensweise nach dem Sollkonzept wird unterschiedlich gestaltet, je nach dem, ob Individual- oder Standardsoftware eingesetzt werden soll. Im Falle der Individualentwicklung wird das vollständige Phasenschema angewandt. In der Phase Systementwurf wird die Architektur des zu entwickelnden Softwaresystems festgelegt, dessen Realisierung in der Phase Implementierung und Test erfolgt. Wenn Standardsoftware eingesetzt wird, entfallen der Entwurf und die Implementierung des Gesamtsystems. Dennoch beinhaltet die Standardsoftware-Anpassung oft auch einen erheblichen Implementierungsaufwand. Die letzte Phase beider Varianten des Phasenschemas ist die Systemeinführung, nach der das neue IT-System in der Organisation im tatsächlichen Einsatz läuft.

Nach der Systemeinführung fängt die Systembetriebsphase an, die im eigentlichen Sinn nicht zur originären Systementwicklung gehört. In der Praxis ist die Abgrenzung zwischen Erstentwicklung und Wartung jedoch oft fließend, das Gesamtsystem ist mehreren Änderungszyklen (vor und nach der Auslieferung an den Auftraggeber) unterworfen.

Das überlappende Phasenmodell unterscheidet sich von den reinen softwareorientierten Vorgehensweisen vor allem in folgenden Punkten:

- Es sind nicht nur Rücksprünge zugelassen, sondern auch Vorgriffe auf spätere Phasen. Beispielsweise ist bereits parallel zu den frühen Phasen Prototyping möglich, das der Implementierungsphase zuzuordnen ist.
- Es wird großer Wert auf die organisatorische Einbindung des Systems gelegt. Organisatorische Prozesse sollen nicht automatisiert und damit zementiert, sondern zunächst kritisch überdacht und optimiert oder sogar radikal umgestellt werden.
- Zwischen dem organisatorischen Ist- und Soll-Zustand wird explizit unterschieden. [Suhl+02]

Das Problem des nebenläufigen Modells, dass Entscheidungen zu spät oder nicht abgestimmt getroffen werden, liegt auch hier vor. Durch die Überlappung können wichtige Zeitpunkte der Entscheidungsfindung verpasst werden.

3 Vorgehensmodell unter expliziter Einbeziehung der Kundensicht

Bei dem hier beschriebenen Vorgehensmodell handelt es sich um eine Anpassung und Erweiterung des zuvor vorgestellten überlappenden Phasenmodells. Das ursprüngliche Modell wurde neben der Anpassung einiger Phaseninhalte durch so genannte Abstimmungsprozesse erweitert. Damit besteht das neue Vorgehensmodell aus Phasen und Abstimmungsprozessen.

Eine Phase ist ein zeitlicher Abschnitt des Projektablaufs, der gegenüber anderen Phasen sachlich getrennt ist. In ihr sind vor allem das Projektteam sowie die verschiedenen Typen von Benutzern (Initiatoren und Endbenutzer) beteiligt. Abstimmungsprozesse ermöglichen fest definierte Zeitpunkte zur Kommunikation zwischen der Auftraggeber- und Auftragnehmerseite. In den Abstimmungsprozessen sollte die Auftraggeberseite in der Lage sein, Ergebnisse (qualitativ) bewerten und auf dieser Basis Entscheidungen treffen zu können. Bei der Erläuterung der einzelnen Abstimmungsprozesse werden Beispiele für Möglichkeiten zur Bewertung angeführt.

Ein Aspekt, der in allen Abstimmungsprozessen Berücksichtigung finden sollte und deshalb an dieser Stelle angeführt wird, ist die Einhaltung anvisierter Endtermine. Ein unter Zeitdruck erstelltes Softwaresystem führt zur Vernachlässigung von Aktivitäten, Ergebnissen und Tests. Die Softwarequalität wird das gewünschte Niveau nicht erreichen. Indizien hierfür sollte die Auftraggeberseite zum Anlass nehmen, die Ergebnisse in Frage zu stellen.

Der Autor hält die Abtrennung der in den Abstimmungsprozessen durchzuführenden Tätigkeiten von den Phasen für sinnvoll, um die Auftraggeberseite während der Projektlaufzeit besser einbinden zu können. Die Phasen sind auftragnehmer-/IT-orientiert, wohingegen die Abstimmungsprozesse über die Fortführung des Projekts entscheiden und vornehmlich auftraggeberorientiert sind.

In Abbildung 3 ist dargestellt, wie Auftragnehmer- und Auftraggebersicht zusammengeführt werden können. In der Regel wird die Auftraggeberseite selbst ein Projekt initiiert haben. So ist eventuell ein eigenes Vorgehen definiert (hier mit Vorstudie, Ist-Analyse, Sollkonzept und nicht weiter erläuterten Teilprojekten). Aus diesem Projekt heraus kann zum Beispiel ein (Teil)projekt mit einem IT-Dienstleister gestartet werden. (Abbildung 3, rechter Kasten). Es ist dann notwendig, das Auftraggeber- und IT-Dienstleisterprojekt bzw. -vorgehen miteinander abzustimmen (Abbildung 3, mittlerer Kasten).

Aus Sicht des IT-Dienstleisters ist zusätzlich zu berücksichtigen, dass eventuell einzelne Phasen des Entwicklungsprozesses mit Partnern durchgeführt werden (Abbildung 3, linker Kasten).

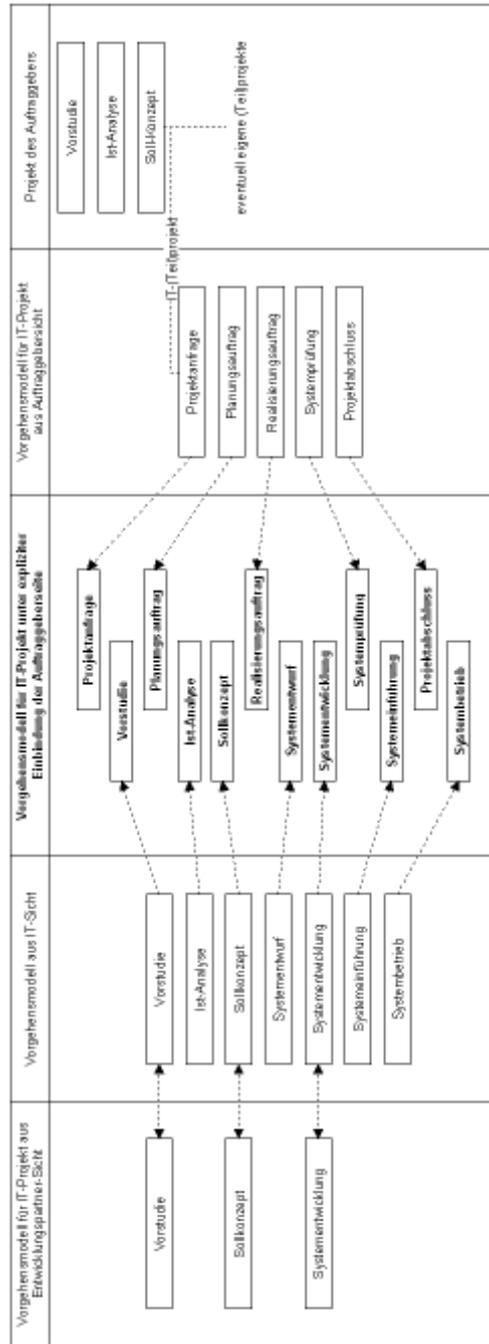


Abbildung 3: Explizite Einbindung der Auftraggeberseite

Die Vorgehensmodelle der Auftraggeber- und Auftragnehmerseite müssen weder dieselben Phasen beinhalten noch das hier beschriebene Vorgehensmodell einsetzen. Eventuell initiiert die Auftraggeberseite gar kein Projekt und steuert den Projektablauf nur über die Abstimmungsprozesse. Wichtig ist die Ausrichtung im Hinblick auf die fünf Abstimmungsprozesse. Diesen Aspekt soll Abbildung 4 verdeutlichen. Dabei wird das relativ einfache Vorgehensmodell eines Auftraggebers (Phasen Ist-Analyse, Sollkonzept, Umsetzung und Einführung) mit dem eines Auftragnehmers (Nutzung des hier beschriebenen Vorgehensmodells) verknüpft.

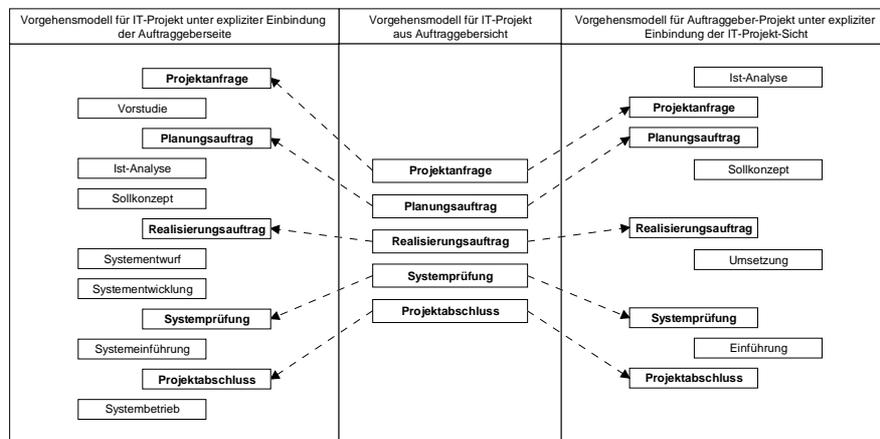


Abbildung 4: Schnittstellen zwischen den Vorgehensmodellen des Auftraggebers und des Auftragnehmers

Die Annahmen des überlappenden Phasenmodells, dass Phasen explizit parallel, jedoch mit wechselnder Intensität während des Projektablaufs modelliert werden, sollen übernommen werden (vgl. Abschnitt „Überlappendes Phasenmodell“ bzw. Abbildung 1 und 2). Damit sind auch in diesem Modell nicht nur Rücksprünge zugelassen, sondern auch Vorgriffe auf spätere Phasen. Bei Vorgriffen sollten die dazwischenliegenden Abstimmungsprozesse immer Berücksichtigung finden.

Die explizite Parallelität der Phasen ist in den Abbildungen 3 bis 8 nicht dargestellt. Im Endeffekt müssten die Abbildungen 1 und 2 vom sachlichen Inhalt in die genannten Abbildungen integriert sein.

Die Erweiterung des Vorgehensmodells um die Abstimmungsprozesse hat im Wesentlichen zwei Vorteile: (1) Das Vorgehensmodell ist nicht nur IT-entwicklungslastig, sondern integriert die Auftraggeberseite. (2) Die Nutzung von Projektlinien in Kombination mit den Abstimmungsprozessen ermöglicht die einfache Verfolgung mehrerer Alternativen bzw. die Aufsplittung in Teilprojekte (Integration von Entwicklungspartnern).

(1) *Das Vorgehensmodell ist nicht nur IT-entwicklungslastig, sondern integriert die Auftraggeberseite.*

Mit der expliziten Berücksichtigung der Auftraggeberseite wird ein Schwerpunkt des Vorgehensmodells auf die Auftraggeber-Auftragnehmer-Beziehung gelegt. Die vorher beschriebenen Vorgehensmodelle vernachlässigen diese durchaus wichtige Beziehung für ein Projekt. Bei [Stah+99] weist der Autor z.B. darauf hin, dass Aufträge zum einen für Ist-Analyse/Sollkonzept und zum anderen für die Realisierung gegeben werden können. So kann mit dem ersten Auftrag festgestellt werden, ob die Einführung, Umstellung oder Reorganisation sinnvoll bzw. wirtschaftlich ist, bevor mit dem zweiten die Entscheidung zur Systementwicklung gegeben wird. Oft werden diese Entscheidungsfindungs-Prozesse den Phasen zugeordnet und gehen in diesen unter. Bei [Suhl+02] wird z.B. erwähnt, dass es Meilensteine während des Projektablaufs geben kann, an denen konsistente Ergebnisse der jeweiligen Phasen vorliegen, die dem Auftraggeber vorgestellt werden müssen. Als Beispiele werden das Sollkonzept und der Systementwurf angeführt. Eine Ausgestaltung dieser Prozesse im Modell unterbleibt.

(2) *Die Nutzung von Projektklinien in Kombination mit den Abstimmungsprozessen ermöglicht die einfache Verfolgung mehrerer Alternativen bzw. die Aufsplittung in Teilprojekte.*

Durch die Betrachtung mehrerer Alternativen in den Phasen soll die optimale Problemlösung gefunden werden. Jede Alternativenverfolgung wird durch eine Alternativen-Projektklinie dargestellt. Die Koordination liegt beim Auftragnehmer-Projektleiter. Eine alternative oder parallele Phasenbearbeitung durch verschiedene Anbieter ist somit möglich. In diesem Fall kann die Projektanfrage an externe Lösungsanbieter oder an eine in-House-Abteilung weitergegeben werden. In den von den Phasen abgetrennten Abstimmungsprozessen kann damit von der Auftraggeberseite entschieden werden, mit welcher Alternative bzw. welchen Alternativen das Projekt weitergeführt werden soll. So nimmt die Anzahl der Projektklinien von Phase zu Phase ab, bis spätestens im Systembetrieb nur noch die Projektklinie des endgültig zu installierenden Systems übrig bleibt (Abbildung 5).

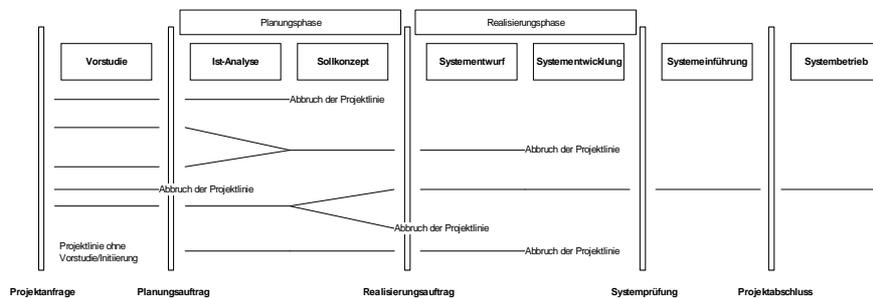


Abbildung 5: Einsatz verschiedener Projektklinien

Außerdem kann das Konzept der Projektlinien dazu genutzt werden, das Gesamtprojekt in Teilprojekte aufzusplitten. Eine Aufteilung in Arbeitspakete für einzelne Projektlinien kann zur Arbeitsteilung und damit zur Parallelbearbeitung eingesetzt werden. Zum Beispiel können eine Gesamtanalyse und die Phasen der Konzepterstellung, des Systementwurfs und der Systementwicklung in Parallelarbeit durchgeführt werden. In der Systemeinführung würde dann die Integration der Teilkomponenten stattfinden (Abbildung 6).

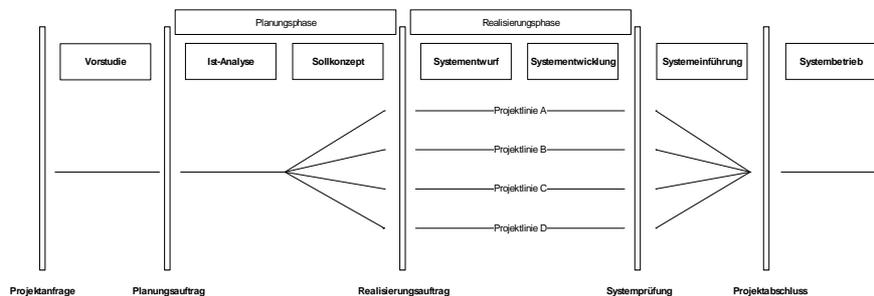


Abbildung 6: Nutzung von Projektlinien zur Aufsplittung in Teilprojekte

Eine Kombination der genannten Einsatzmöglichkeiten für Projektlinien (Betrachtung mehrerer Alternativen und Aufsplittung in Teilprojekte) ist möglich.

- Abstimmungsprozess "**Projektanfrage**"
1. Phase "**Vorstudie**"
- Abstimmungsprozess "**Planungsauftrag**"
2. Phase "**Ist-Analyse**"
 3. Phase "**Sollkonzept**"
- Abstimmungsprozess "**Realisierungsauftrag**"
4. Phase "**Systementwurf**"
 5. Phase "**Systementwicklung**"
- Abstimmungsprozess "**Systemprüfung**"
6. Phase "**Systemeinführung**"
- Abstimmungsprozess "**Projektabschluss**"
7. Phase "**Systembetrieb**"

Abbildung 7: Vorgehensmodell für Individualentwicklung

- Abstimmungsprozess "**Projektanfrage**"
1. Phase "**Vorstudie**"
- Abstimmungsprozess "**Planungsauftrag**"
2. Phase "**Ist-Analyse**"
 3. Phase "**Sollkonzept**"
- Abstimmungsprozess "**Realisierungsauftrag**"
- 4./5. Phase "**Standardsoftwareanpassung**"
- Abstimmungsprozess "**Systemprüfung**"
6. Phase "**Systemeinführung**"
- Abstimmungsprozess "**Projektabschluss**"
7. Phase "**Systembetrieb**"

Abbildung 8: Vorgehensmodell für Standardsoftware-Einführung

Einen Überblick über das Vorgehensmodell bei Individualentwicklung und Standardsoftware-Einführung geben die Abbildungen 7 und 8. Die Phasen Ist-Analyse und Sollkonzept werden auch zur Planungsphase und die Phasen Systementwurf und Systementwicklung zur Realisierungsphase zusammengefasst.

Die einzige Phase, die deutlich vom bei [Suhl+02] definierten Phasen-Ziel abweicht, ist die Vorstudie. Auf sie und die Abstimmungsprozesse wird deshalb im Folgenden mehr Gewicht gelegt.

Ein weiterer Schwerpunkt liegt auf der Überprüfung bzw. Bewertung der Phasen-Ergebnisse in den Abstimmungsprozessen durch die Auftraggeberseite. Sie sollte erkennen können, ob in den Phasen brauchbare Ergebnisse erarbeitet worden sind und welche Entscheidungen von ihr in den Abstimmungsprozessen getroffen werden müssen.

Bei dem hier vorgestellten Ansatz handelt es sich um eine Maximalausprägung des Modells. Je nach vorliegenden Rahmenbedingungen, Auftraggeberwünschen und Komplexität der Aufgabenstellung kann es sinnvoll sein, Tätigkeiten oder sogar ganze Phasen und Abstimmungsprozesse nicht oder nur verkürzt durchzuführen. Die Anpassung an das jeweilige Projekt kann, genauso wie beim V-Modell, Tailoring („Maßschneidern“) genannt werden. Zum Beispiel gibt es Projekte, in denen keine Vorstudie oder in denen eine Vorstudie als verkürzte/ günstigere Planungsphase (Phasen Ist-Analyse und Sollkonzept) durchgeführt wird.

3.1 Abstimmungsprozess Projektanfrage

Eine Projektanfrage sollte an eine definierte Anlaufstelle gerichtet werden können. Diese positioniert die Anfrage und bestimmt das weitere Vorgehen. Dazu sollte die Auftraggeberseite einen Projekttitel, Projektziele, eine Grobbeschreibung ihrer Anforderungen, durch die die durchzuführende Aufgabe abgegrenzt und kurz beschrieben wird, sowie eventuelle Rand- und Rahmenbedingungen formulieren. Diese Angaben sind zur Projekt-Charakterisierung notwendig, können aber in den nachfolgenden Phasen noch angepasst werden. Oft wird ein Projekt-Kick-Off⁶ durchgeführt. Der Abstimmungsprozess Projektanfrage ist in erster Linie mit der Phase Projektbegründung bei [Kral96] und [Stah99] zu vergleichen. Die dort vorgestellten Vorgehensmodelle fahren danach jedoch direkt mit der Ist-Analyse

⁶ Kick-Offs werden in anderen Modellen häufig zu Beginn einer Phase angesetzt. Hier werden Kick-Offs in den Abstimmungsprozessen durchgeführt und leiten so inhaltlich die nächste Phase ein. Dies erscheint dem Autor sinnvoll, weil nach der Entscheidungsfindung in den Abstimmungsprozessen ein fließender Übergang in die Phase stattfindet, die der Auftraggeber oft noch mitbestimmt. Den Prozess der Phasenverabschiedung und der Phasenvorbereitung zu trennen (Typische Frage: ... und wie geht es jetzt weiter?) ist in der Praxis nicht sinnvoll.

fort, sodass die im Folgenden vorgestellten Tätigkeiten der Vorstudie nicht explizit ausgeführt werden.

Die Prozess-Ergebnisse können z.B. durch technische, organisatorische und menschliche Bedingungen beurteilt werden. Auf der technischen Seite ist beispielsweise die Verwendung von Prinzipien, Methoden und Werkzeugen zu betrachten.

Organisatorisch ist die Bewertung des Vorgehensmodells möglich. Dabei können die folgenden Punkte berücksichtigt werden: [vgl. Wall01]

- *Vollständigkeit*: Das Modell überspannt alle Phasen/Iterationen der Entwicklung.
- *Modularität*: Der Projektablauf ist in überschaubare, separat zu behandelnde Abschnitte gegliedert, die wohldefinierte Vorgaben und Ergebnisse haben.
- *Systematik*: Alle Begriffe sind nach einer einheitlichen Systematik gebildet und sind treffend und aussagekräftig gewählt.
- *Anpassbarkeit*: Das Modell lässt sich organisatorischen, firmen- und projektspezifischen Gegebenheiten sowie softwaretechnischen Neuerungen leicht anpassen.
- *Maschinelle Unterstützung*: Es ist eine Rechnerunterstützung für das Vorgehensmodell vorhanden.

Der menschliche Aspekt kann z.B. durch die Kompetenz des Projektleiters bzw. Phasenverantwortlichen überprüft werden.

Des Weiteren sollte analysiert werden, ob die Anforderungen und das Projektziel richtig definiert bzw. richtig aufgenommen wurden.

3.2 Phase Vorstudie

Gegenüber dem ursprünglichen überlappenden Phasenmodell hat die Vorstudie einen anderen Charakter. Das Ziel einer Vorstudie ist nicht mehr die Entscheidung über die Projektdurchführung. [vgl. Suhl+02] Stattdessen soll eine Vorstudie einen Lösungsvorschlag unterbreiten, der der Auftraggeberseite im nachfolgenden Abstimmungsprozess zur Entscheidung vorgestellt wird. Durch die Trennung der Entscheidung von der Phase können mehrere Vorstudien bzw. Lösungsvorschläge erarbeitet werden. In der Praxis wird diese Möglichkeit genutzt, sich von verschiedenen Anbietern einen Lösungsvorschlag erarbeiten zu lassen. Das Ergebnis der Vorstudie bildet damit die Entscheidungsgrundlage zur Vergabe des Planungsauftrags. In vielen Projekten wird ein Teilauftrag für die Planungsphase und dann für die Realisierungsphase vergeben. So wird der Auftraggeber vornehmlich in den Abstimmungsprozessen Planungsauftrag und Realisierungsauftrag über die Weiterführung des Projekts entscheiden.

Die Vorstudie ist selbst ein kleines Projekt und hat den Charakter einer abgespeckten Planungsphase (Phasen Ist-Analyse und Sollkonzept). Sie kann auch ein günstigerer Ersatz der Planungsphase sein. Dann würde nach der Vorstudie direkt der Realisierungsauftrag vergeben.

Im Abstimmungsprozess Projektanfrage wird in der Regel schon entschieden, ob oder wie die erste Phase Vorstudie ausgeführt werden soll. Abhängig von der Tragweite des Projekts und den Risiken, die für den Auftraggeber und den Auftragnehmer damit verbunden sind, können verschiedene Ausprägungen von Vorstudien unterschieden werden. Im Extremfall wird die gesamte Planungsphase in der Vorstudie abgehandelt. Dann steht als Ergebnis eine ausführliche Analyse mit einem konkreten Pflichtenheft und einer detaillierten Machbarkeits- und Durchführbarkeitsstudie fest. Der Projektleiter entscheidet, welche Tätigkeiten aus den Phasen Ist-Analyse und Sollkonzept in die Vorstudie aufgenommen werden sollen.

Im Folgenden werden vier typische Vorstudien mit ihren verschiedenen Ausprägungen vorgestellt.

Grundtyp 1: Ist-Analyse und Lösungsvorschlag

Eine häufig vorkommende Ausprägung nimmt eine Ist-Analyse des Betrachtungsgegenstands vor, an die sich die Erarbeitung eines Lösungsvorschlags anschließt.

Grundtyp 2: Lösungsvorschlag

Gibt es keinen Ist-Zustand oder lohnt es sich nicht den Ist-Zustand zu analysieren, wird nur ein Lösungsvorschlag erarbeitet.

Grundtyp 3: Grundtyp 1 oder 2 mit Durchführbarkeitsstudie

Der ausführliche Vorstudien-Typ erstellt auch eine Durchführbarkeitsstudie. Werden die Ergebnisse sehr detailliert erarbeitet, kann die Vorstudie die Planungsphase ersetzen.

Grundtyp 4: Grundtyp 1, 2 oder 3 mit Prototyp

Um einen besseren Eindruck vom Lösungsvorschlag zu bekommen, kann zusätzlich ein Prototyp erstellt werden.

Die Ergebnisse der einzelnen Phasen-Tätigkeiten sollten im Vorstudien-Dokument schriftlich verfasst werden.

3.3 Abstimmungsprozess Planungsauftrag

Mit Abschluss der Phase Vorstudie bzw. der verschiedenen Vorstudien der einzelnen Projektlinien wird der Abstimmungsprozess Planungsauftrag eingeleitet. Die Ergebnisse der Vorstudien werden hier entweder durch den Auftragnehmer-Projektleiter (in Zusammenarbeit mit dem Auftraggeber und/oder Auftraggeber-

Projektleiter) bewertet und verglichen oder der Auftraggeberseite in einer Präsentation vorgestellt. Im Rahmen einer Zusammenkunft wird von einem Planungs-Kick-Off gesprochen. Danach wird entschieden, welchem Vorgehen ein Planungsauftrag, der die ausführliche Ist-Analyse und die Erarbeitung eines Detail-Konzepts in Auftrag gibt, erteilt werden soll.

Im Extremfall kann das gesamte Projekt abgebrochen werden (keine Projektlinie bekommt einen Planungsauftrag), wenn zu diesem Zeitpunkt kein Lösungsansatz als umsetzungswürdig eingestuft wird.

Zur Bewertung des Prozess-Ergebnisses durch die Auftraggeberseite sollten die in der Vorstudie gemachten Lösungsvorschläge auf ihre Relevanz, ihre Machbarkeits- und Durchführbarkeitsstudien getestet werden. Hierzu können verschiedene Kosten-/ Nutzenvergleiche eingesetzt werden.

3.4 Planungsphase

Die Phasen Ist-Analyse und Sollkonzept werden auch zur Planungsphase zusammengefasst.

3.4.1 Phase Ist-Analyse

Zur Feststellung und Bewertung des organisatorischen und technischen Ist-Zustandes dient die Phase Ist-Analyse. Als Phasen-Tätigkeiten werden dabei die Systemabgrenzung, die Systemerhebung (mit Ablauf-, Aufbauorganisation und IT-Infrastruktur), die Systembeschreibung und die Systemanalyse (mit Potential-/ Schwachstellenanalyse) durchgeführt. Hierfür stehen verschiedene Techniken und Methoden bereit. [siehe Suhl+02, S. 354-359]

Die Phase kann entsprechend dem Betrachtungsgegenstand nur rudimentär oder auch gar nicht durchgeführt werden. Abschlussdokumente sind die schriftlich verfasste Systembeschreibung und Systemanalyse.

3.4.2 Phase Sollkonzept

Ziel der Phase Sollkonzept ist die Gestaltung eines (neuen) organisatorischen und informationstechnischen Konzeptes. Bevor das Detail-Konzept erstellt werden kann, sind verschiedene Tätigkeiten durchzuführen:

1. Die detaillierten Beschreibungen der Systemanforderungen aus Benutzersicht müssen (aus der Vorstudie) überarbeitet oder erstellt werden. Das Ergebnis wird auch Fachkonzept oder Lastenheft genannt.
2. Auf Basis der Systemanforderungen sollte die Machbarkeitsstudie (aus der Vorstudie) überarbeitet oder erstellt werden.

3. Aus dem Fachkonzept bzw. Lastenheft wird das Pflichtenheft verfasst. In ihm wird festgeschrieben, was später wirklich umgesetzt werden soll.

Nachdem dann das Detail-Konzept erstellt worden ist, kann eine Empfehlung für Standard- oder Individualsoftware gegeben werden.

Bei Vergabe mehrerer Planungsaufträge werden auch mehrere Konzepte auf Basis verschiedener Lösungsansätze entwickelt. Im nachfolgenden Abstimmungsprozess wird entschieden, welche Konzepte in der Realisierungsphase bzw. den Phasen Systementwurf und Systementwicklung weiterverfolgt bzw. umgesetzt werden. Als Entscheidungsgrundlage muss eine Durchführbarkeitsstudie für das Detail-Konzept erstellt werden. Oft wird die Durchführbarkeitsstudie aus der Vorstudie überarbeitet und detailliert.

3.5 Abstimmungsprozess Realisierungsauftrag

In Gesprächen und Diskussionen werden die Lösungsansätze präsentiert. Sämtliche Alternativen sind hinsichtlich ihrer Machbarkeits- und Durchführbarkeitsstudien gegenüberzustellen (Wirtschaftlichkeitsvergleiche). Dabei ist neben der Durchführung von Kosten-/Nutzenvergleichen vor allem festzustellen, ob es überhaupt möglich bzw. sinnvoll ist, eine der vorgeschlagenen Alternativen unter den gegebenen personellen, technischen und organisatorischen Voraussetzungen zum gegenwärtigen Zeitpunkt zu realisieren.

Eventuell ist die Auftraggeberseite mit keinem Lösungsvorschlag zufrieden. Dann muss ganz neu in die Konzeption mit anderen Ideen eingestiegen oder das Projekt an dieser Stelle auf Grund der ungenügenden Lösungsvorschläge abgebrochen werden.

Von Auftraggeberseite können auch weitere Anforderungen definiert werden, die zur Änderung des jeweiligen Konzepts führen können. Das würde den Wiedereinstieg in die Phase Sollkonzept bedeuten. Mit der Wahl eines Konzepts wird der Realisierungsauftrag vergeben. Damit wird die Realisierungsphase (bei Individualentwicklung die Phasen Systementwurf und Systementwicklung und bei Wahl von Standardsoftware die Phase Standardsoftwareanpassung) auf der Basis des gewählten Konzepts in Auftrag gegeben.

Mit dem Realisierungs-Kick-Off, das ein Teil der oben genannten Besprechungen sein kann, wird auf die Realisierungsphase eingestimmt.

Die durchgeführten Tätigkeiten bzw. die Prozesse der Phasen Ist-Analyse und Sollkonzept können durch die Phasen-Ergebnisse bewertet werden. Dabei sollten die betrachteten Prozesse messbar sein.⁷

⁷ Wie Prozesse gemessen werden können, wird z.B. bei [Wall01] beschrieben.

3.6 Realisierungsphase

Bei dem hier beschriebenen Vorgehensmodell werden Projekte mit Individualentwicklung und Projekte zur Einführung von Standardsoftware unterschieden. Der Unterschied liegt in erster Linie in der Durchführung der Realisierungsphase: Bei der Individualentwicklung sind die Phasen Systementwurf und Systementwicklung und bei der Nutzung von Standardsoftware die Phase Standardsoftwareanpassung durchzuführen.

3.6.1 Individualentwicklung

3.6.1.1 Phase Systementwurf

Die Phase Systementwurf dient der Beschreibung der Systemarchitektur. Dabei werden die Komponenten des Systems mit ihren Schnittstellen (maschinennah) modelliert, ohne auf konkrete Produkte der Informationstechnik ausgerichtet zu sein.

Der Phasenverantwortliche muss die Aufteilung des Problems in Teilprobleme bzw. Systemkomponenten vornehmen. Hat er diese identifiziert, sind Beziehungen zwischen ihnen zu erarbeiten und der Funktions- und Leistungsumfang sowie das Verhalten eventueller Systemkomponenten zu beschreiben.

Es sollte eine auf relativ abstraktem Niveau durchgeführte Spezifikation im so genannten Architekturentwurf, der sich durch eine starke Orientierung an Implementierungsdetails auszeichnet, verfeinert werden. So nähert sich der Phasenverantwortliche langsam einem in der Systementwicklung umsetzbaren Modell. Der Systementwurf wird häufig von Prototyping-Aktivitäten begleitet.

3.6.1.2 Phase Systementwicklung

Ziel der Phase Systementwicklung ist die Erstellung eines lauffähigen Programms mit der zugehörigen Dokumentation auf Basis des Systementwurfs. Als Phasentätigkeiten können (1) die Programmierung, (2) Modul-, Integrations-, Programm- und Systemtests [vgl. Suhl+02, S. 397+398] und (3) die Erstellung der Produktdokumentation aufgeführt werden.

3.6.2 Standardsoftwareanpassung

Bei Projekten mit Einführung von Standardsoftware findet die Realisierungsphase nur bedingt statt. Die Phasen Systementwurf und Systementwicklung werden nur für einzelne Systemkomponenten durchgeführt, für die Anpassungen bzw. Erweiterungen des Systems durch Implementierungstätigkeiten notwendig werden. Bei der Individualentwicklung wird das System vornehmlich selbst

implementiert, wohingegen bei der Einführung eines Standardprodukts in erster Linie das so genannte Customizing anfällt.

An dieser Stelle wird angemerkt, dass die nachfolgenden Abstimmungsprozesse und Phasen in vielen Projekten vernachlässigt werden. Das heißt, dass nach der Systementwicklung eine schnelle Systemeinführung durchgeführt wird. Oft wird der Systembetrieb gar nicht geplant. Deshalb sollte die Auftraggeberseite besonders darauf achten, dass die nachfolgenden Abstimmungsprozesse und Phasen gewissenhaft ausgeführt werden.

3.7 Abstimmungsprozess Systemprüfung

Nachdem die Realisierungsphase abgeschlossen ist, wird der Abstimmungsprozess Systemprüfung eingeleitet. Er sollte auf Grund der Qualitätssicherung auf zwei Ebenen ablaufen. Zunächst überprüft der Auftragnehmer-Projektleiter das System, bevor er es zum Test an die Auftraggeberseite freigibt. So wird eine Prüfung vorgenommen, die eine gewisse Qualität, Funktionalität und Fehlerfreiheit des Systems erreicht. Danach erfolgt der eigentliche Abnahmetest unter Mitwirkung der Auftraggeberseite in der realen Einsatzumgebung und mit realen Daten.

Beim Abnahmetest wird das System gegen das Pflichtenheft geprüft.⁸ Die Auftraggeberseite kann dabei Testfälle vorgeben und soll sich von der Einsatzfähigkeit des Systems überzeugen. Der Abnahmetest kann am Ende der Phase Systementwicklung oder im Abstimmungsprozess Systemprüfung durchgeführt werden.

Bei Nicht-Erfüllung der Anforderungen wird erneut in die Entwicklung eingestiegen bis die Abnahme positiv verläuft. Der Vorgang wird oft mit einer förmlichen Übergabe abgeschlossen. Die Auftraggeberseite ist dann mit dem Projektergebnis einverstanden und gibt die Systemeinführung in Auftrag.

Während der Tests oder schon während der Phase Systementwicklung kann erkannt werden, dass Modifikationen oder Erweiterungen der neu erstellten Anwendung erforderlich sind, die vom ursprünglichen Konzept (noch) nicht abgedeckt werden. Diese Anforderungen müssen dokumentiert und die zusätzlichen Entwicklungsaufwände geschätzt werden. Nachdem entschieden wurde, ob und welche zusätzlichen Anforderungen zu realisieren sind, werden die Aufgaben aktualisiert und an die verantwortlichen Personen übergeben. Der Projektplan muss entsprechend überprüft und überarbeitet werden. So kann für zusätzliche Entwicklungsaufwände ein Rückschritt im Vorgehensmodell bis maximal an den Anfang der Phase Ist-Analyse durchgeführt werden. Von dort wird eine weitere Projektlinie gestartet, die im Abstimmungsprozess

⁸ Für weitere Prüfziele siehe [Wall01].

Systemprüfung wieder in die ursprünglichen Projektklinie mündet (Abbildung 9). Wichtig ist, dass bei Rücksprüngen die Abstimmungsprozesse beachtet werden und so die Auftraggeberseite eingebunden ist.

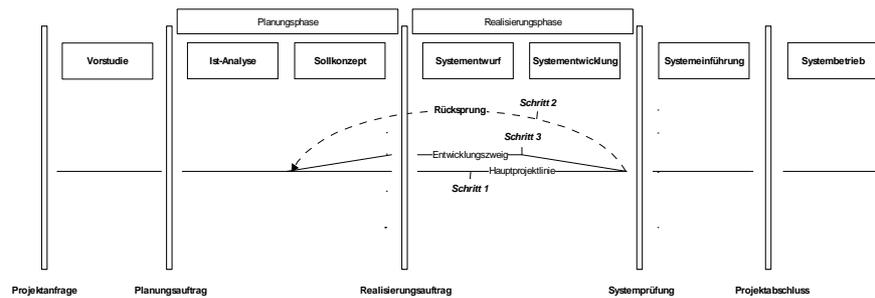


Abbildung 9: Rücksprung im Vorgehensmodell durch zusätzliche Entwicklungsaufwände

Die Überprüfung der Entwicklungsergebnisse durch die Auftraggeberseite kann z.B. anhand der Qualität des Software-Produkts vorgenommen werden. Hierfür können verschiedene Merkmale herangezogen werden: [vgl. DIN 66272 und Wall01]

- *Wartbarkeit*: Wartungsfreundlichkeit, Pflegbarkeit, Testbarkeit, Änderbarkeit, Anpassbarkeit, Erweiterbarkeit
- *Benutzerfreundlichkeit*: Benutzbarkeit, Handhabbarkeit, Bedienerfreundlichkeit, Anwenderfreundlichkeit, Dokumentation, Transparenz
- *Zuverlässigkeit*: Verfügbarkeit, Robustheit, Sicherheit, Stabilität, Instandsetzbarkeit, Reaktionssicherheit im Fehlerfall, Reproduzierbarkeit
- *Korrektheit*: Funktionserfüllung, Fehlerfreiheit, Richtigkeit, Gebrauchstauglichkeit, Vollständigkeit
- *Effizienz*: Performance, Verbrauchsverhalten, angemessener Ressourcenverbrauch, Zeitverhalten
- *Portabilität*: Übertragbarkeit, Kompatibilität
- *Sonstige*: Modularität, Strukturierung, Systemkonformität, Verwendung von Standards
- *Funktionalität*

Weiterhin kann die Qualität des Entwicklungsprozesses analysiert werden. Dazu sollte ein vorhandenes Software-Konfigurationsmanagement⁹ betrachtet werden.

Den Abschluss des Abstimmungsprozesses bildet das Systemeinführungs-Kick-Off. Mit ihm wird die Phase Systemeinführung gestartet.

3.8 Phase Systemeinführung

Die Einführung umfasst alle Tätigkeiten, die mit dem Wechsel zum neuen System verbunden sind. Am Ende dieser Einführung muss das System in die neu gestalteten Geschäftsprozesse eingebunden sein.

Für die Phase sollte ein detaillierter Einführungsplan erarbeitet werden.

Vor der eigentlichen Inbetriebnahme des Systems sollten folgende Aktivitäten durchgeführt werden: Beschaffung und Installation der erforderlichen Hardware und Basissoftware sowie zusätzlicher technischer Infrastruktur, Umstellung alter Datenbestände, Installation neuer Software und Schulung der Anwender.

Diese Tätigkeiten können auch parallel zu der Phase Systementwicklung durchgeführt werden (überlappendes Phasenmodell). Mit den Schulungsmaßnahmen kann zum Beispiel auf einem weit entwickelten Prototyp begonnen werden.

Eine Einführung kann mit dem gesamten Anwendungssystem einschließlich sämtlicher Daten zu einem Stichtag (schlagartige Einführung), stufenweise (inkrementell) mit nur einem Teil des Anwendungssystems, mit nur einem Teil der Daten (Pilotanwender) oder als Parallellauf unter gleichzeitiger, zeitlich begrenzter Fortführung des alten Systems stattfinden.

3.9 Abstimmungsprozess Projektabschluss

Dieser Abstimmungsprozess definiert das Ende des Projekts mit dem Auftraggeber. Das System ist zu diesem Zeitpunkt vollständig installiert und integriert. Es wird offiziell an die Anwender übergeben. Das Projektteam wird aufgelöst und das System geht in den Systembetrieb über.

Das Systembetriebs-Kick-Off bildet den Übergang in die Phase Systembetrieb.

Projektabschlusskontrollen dienen der Bewertung des Projektverlaufs und der Dokumentation der gewonnenen Erfahrungen.

⁹ Eine Konfiguration ist eine benannte und formal freigegebene Menge von Entwicklungsergebnissen, die in ihrer Wirkungsweise und ihren Schnittstellen aufeinander abgestimmt sind und gemeinsam eine vorgegebene Aufgabe erfüllen sollen. [Wall01]

Die Produktbeschreibung und die Benutzerdokumentation können von der Auftraggeberseite herangezogen werden, um das Gesamtprodukt zu bewerten. Die Anforderungen an eine qualitativ gute Dokumentation sind z.B. Änderbarkeit, Aktualität, Eindeutigkeit, Identifizierbarkeit, Normkonformität, Verständlichkeit und Widerspruchsfreiheit. [vgl. Wall01]

3.10 Phase Systembetrieb

Ziel der Phase, die die Nutzung des Systems und so hoffentlich den größten zeitlichen Anteil am Software-Lebenszyklus beinhaltet, ist die Aufrechterhaltung der Systemfunktionen in ihrem betriebswirtschaftlichen Umfeld. Dazu muss das System nach der Einführung bis zur Stilllegung gewartet, gepflegt, manchmal umgebaut bzw. erweitert werden. So beinhaltet die Phase neben der reinen Fehlerbeseitigung auch die Anpassung an geänderte Systemanforderungen und ist durch eine Reihe von Systemänderungszyklen gekennzeichnet.

Die Mitarbeiter, die sich um den Systembetrieb kümmern, müssen mit ihren Aufgaben vertraut gemacht werden.

Der Systembetrieb gehört nicht zur originären Entwicklung. In der Praxis ist die Abgrenzung zwischen Erstentwicklung und Wartung oft fließend.

Eine zweite Art von – fachlichen – Projektabschlusskontrollen kann erst in der Nutzungszeit erfolgen. Sie beziehen sich auf die ersten Erfahrungen bei der Nutzung des Systems, auf dessen tatsächliche Verwendung durch die Endbenutzer und schließlich auf Untersuchungen zum Eintritt der geplanten Wirtschaftlichkeit.

4 Zusammenfassung

Das hier vorgestellte Vorgehensmodell hilft in erster Linie, die Schnittstelle zwischen der Auftragnehmer- und Auftraggeberseite für ein Projekt mit IT-Entwicklung zu verbessern. Die fünf Abstimmungsprozesse ermöglichen es, die in den beiden Unternehmen geplanten Projektvorgehen miteinander zu verknüpfen. Ein wesentlicher Vorteil des Vorgehensmodells besteht darin, dass beide Seiten zu expliziten Entscheidungspunkten eine Abstimmung erzielen können.

Literatur

[Abts+98] Abts, D.; Müller, W.: Grundkurs Wirtschaftsinformatik – Eine kompakte und praxisorientierte Einführung, Verlag Vieweg, Göttingen, 2., überarbeitete Auflage, 1998.

- [Balz00] Balzert, H.: Lehrbuch der Software-Technik. Band I: Software-Entwicklung. Spektrum Akademischer Verlag, Heidelberg, Berlin, 2. Auflage, 2000.
- [Balz98] Balzert, H.: Lehrbuch der Software-Technik. Band II: Software-Management, Software-Qualitätssicherung, Unternehmensmodellierung. Spektrum Akademischer Verlag, Heidelberg, Berlin, 1998.
- [Boeh88] Boehm, B. W.: Verifying and Validating Software Development and Enhancement. IEEE Computer, May 1988, pp. 61-71.
- [Jaco+99] Jacobson, I.; Booch, G.; Rumbaugh, J.: The Unified Software Development Process, Addison Wesley Longman, 1999.
- [Kral96] Krallmann, H.: Systemanalyse im Unternehmen: Geschäftsprozessoptimierung, Partizipative Vorgehensmodelle, Objektorientierte Analyse, R. Oldenbourg Verlag, München, 2. Auflage, 1996.
- [Oest01] Oestereich, B.: Objektorientierte Softwareentwicklung: Analyse und Design mit der Unified modeling language, Oldenbourg Wissenschaftsverlag, München, 5., völlig überarbeitete Auflage, 2001.
- [Sche97] Scheer, A..W.: Referenzmodelle für industrielle Geschäftsprozesse, Springer Verlag, Berlin, 2., durchgesehene Auflage, 1997.
- [Schw97] Schwarze, J.: Einführung in die Wirtschaftsinformatik, Verlag Neue Wirtschafts-Briefe, Herne, 4., völlig überarbeitete und erweiterte Auflage, 1997.
- [Suhl+02] Suhl, L.; Blumstengel, A.: Systementwicklung, Aus: J. Fischer u.a.: Bausteine der Wirtschaftsinformatik: Grundlagen, Anwendungen, PC-Praxis, Erich Schmidt Verlag, Berlin, 3., überarbeitete Auflage, 2002.
- [Stah+99] Stahlknecht, P.; Hasenkamp, U.: Einführung in die Wirtschaftsinformatik, Springer Verlag, Berlin, 9. Auflage, 1999.
- [V-Mod97] V-Modell 97: Entwicklungsstandard für IT-Systeme des Bundes, Vorgehensmodell, Teil 1: Regelungsteil, Teil 3: Handbuchsammlung, Allgemeiner Umdruck Nr. 250/1, Juni 1997, BWB IT 15, Koblenz 1997.
- [Vers02] Versteegen, G. (Hrsg.): Software Management, Springer Verlag, Berlin, 2002.
- [Wall01] Wallmüller, E.: Software-Qualitätsmanagement, Carl Hanser Verlag, München, 2., völlig überarbeitete Auflage, 2001.