

Teil I

Methodisches Vorgehen

In diesem Teil:

- **Kapitel 1**
Einleitung 13
- **Kapitel 2**
Die Implementierungsmethode im Überblick 23
- **Kapitel 3**
Das Projekt 43

Einleitung

Ein neues **Enterprise-Resource-Planning-System** (ERP-System) soll in Ihrer Firma eingeführt werden – und Sie sind dabei! Bei diesem System handelt es sich nicht um irgendeines, sondern es ist »DAS« IT-System.

Diese sind in der Regel:

- Das Finanz System mit
 - dem Hauptbuch,
 - den Nebenbüchern Debitoren, Kreditoren- und Anlagebuchhaltung
- das Materialwirtschaftssystem
- das Kunden-Management-System/CRM
- das Beschaffungssystem
- das Auftragsverwaltungssystem
- das Produktionsplanungs- und Steuerungssystem
- Lohn- und Gehalt-Software
- Personal-Management-Systeme

Nicht alle diese Systeme, wie Lohn und Gehalt, sind klassisch im Begriff ERP zusammengefasst, doch ist die hier vorgestellte Methode auch für die Implementierung solcher Systeme geeignet.

Wir sprechen hier von dem System, das die betriebswirtschaftlichen Geschäftsprozesse und Wertschöpfungen Ihres Unternehmens

- abbildet,
- dokumentiert und
- unterstützt.

Abgebildet und dokumentiert

Die Abbildung und die Dokumentation der betriebswirtschaftlichen und wertschöpfenden Prozesse ist eine Kernverpflichtung eines Unternehmens gegenüber den Eigentümern und der Gesellschaft (Staat, Finanzamt ...), die mit der Nutzung eines ERP-Systems erfüllt wird. Zwar erfolgt die Wertschöpfung im Unternehmen z.B. in der Produktion, indem neue Produkte hergestellt werden, doch die Manifestierung/Abbildung dieses Wertes erfolgt erst mit einer (Lager-)Buchung im

ERP-System. Diese und alle anderen Buchungen werden periodisch summiert und z.B. im Monatsabschluss dokumentiert. Dies ist der wichtigste Aspekt eines ERP-Systems.

Unterstützt

Darüber hinaus wird vom ERP-System erwartet, dass das System bei den betriebswirtschaftlichen und wertschöpfenden Prozessen unterstützt, sei es durch Planungs- oder Kalkulationsfunktionen oder durch die Beistellung von Informationen, um die Prozesse des Unternehmens zu steuern. Hier wären die Verkaufsstatistiken oder die Materialbedarfsplanung als Beispiel zu nennen.

1.1 Ein neues System

Beginnen wir mit dem Warum.

Warum soll ein neues System eingeführt werden? Die Antworten sind hier so vielfältig wie das Geschäftsleben selbst und dennoch lassen sich die Beweggründe gruppieren.

1.1.1 Unternehmerische Notwendigkeit: Auf zu neuen Ufern

Junge und alte Organisationen haben häufig das gleiche Problem:

Das bisher genutzte System entspricht nicht mehr den unternehmerischen Anforderungen. Dies bei jungen Unternehmen aufgrund schnellen Wachstums und veränderten Prozessen bzw. bei älteren, reiferen Organisationen, weil die Anwendung zu unflexibel ist oder durch alte Hardware und bzw. oder durch umfangreiche Anpassungen zu hohe Betriebskosten hat.

Herausforderung hier: Eine neue Software ist notwendig, und die soll natürlich alles besser können!

1.1.2 Release-»Upgrades«

Es beginnt mit der scheinbar harmlosen Information Ihres Systemlieferanten: »Ihre Version ist aus dem Support, Sie müssen auf eine aktuellere Version upgraden.«

Nun gibt es häufig das Argument: »Die Software läuft seit Jahren problemlos und den Support haben wir sowieso nur selten gebraucht, wir können damit noch lange arbeiten.« Dies ist eine Fehleinschätzung!

Betrachtet man die starke Verquickung von Hardware, Betriebssystemen und Anwendungssoftware, so ist dieses Aussitzen und Abwarten nicht ratsam! Ein banaler Ausfall und Austausch einer einfachen Netzwerkkomponente kann zu einem

notwendigen Betriebssystem-Upgrade führen, die dann nicht mehr mit der Anwendungssoftware kompatibel ist, was wiederum zum Systemstillstand führt.

Dieses Risiko sollte man bei einem unternehmenskritischen System nicht eingehen!

Mit der Einsicht in die Notwendigkeiten des Upgrades ist leider auch häufig die Erkenntnis verbunden, dass die im Betrieb eingesetzte Software-Version so stark angepasst ist, dass ein automatisches Upgrade mit den Tools des Systemlieferanten nur sehr selten, eingeschränkt, bis gar nicht möglich ist. Es ist also kein »Upgrade«, sondern eine Neuinstallation notwendig. Dies besitzt den Vorteil, dass man hier auch die Chance hat, alte Zöpfe (Funktionen, die man eigentlich nicht mehr braucht, aber das EDV-System verlangt) abzuschneiden und zusätzliche, in der neuen Version vorhandene Funktionen wie zum Beispiel das Kundenmanagementsystem zu nutzen.

Herausforderung hier: Die neue Version soll natürlich besser sein, aber die gewohnten Abläufe müssen wie bisher funktionieren.

1.1.3 Drop-Ins

In der globalisierten Welt werden Unternehmen permanent gekauft, konsolidiert und eingegliedert – oder weiterverkauft. Das Eingliedern führt bei betroffenen Unternehmenseinheiten häufig dazu, dass auf Druck der neuen »Muttergesellschaft« Prozesse vereinheitlicht und verschlankt werden müssen. Dabei wird auch häufig vorgegeben, mit welchem Business-System zukünftig gearbeitet wird, und dieses ist dann nicht immer das vorhandene.

Herausforderung hier: Das neue System wird in die Unternehmenseinheit hineingedrückt. Eine optimale Unterstützung für die Unternehmenseinheit ist hinter der Vereinheitlichung der Prozesse nachrangig.

1.2 Business-Software einführen, was ist zu tun?

Die Tätigkeiten, die im Rahmen einer ERP-Implementierung zu erledigen sind, lassen sich in zwei Bereiche einordnen.

1. Änderung an und in der Software
2. Änderungen im Unternehmen

Beide Tätigkeiten erfordern Ressourcen und haben somit Auswirkung auf die Kosten der Implementierung und auf die sich ergebenden Betriebskosten.

Dabei ist es wichtig zu verstehen, dass das Implementierungsprojekt immer beide Bereiche umfasst und ein erfolgreiches Projekt sich dadurch auszeichnet, hier die richtige Balance gefunden zu haben.

1.2.1 Änderung an der Software

Beginnen wir mit dem weniger Komplexem, der Software. Die Aktivitäten an der Software lassen sich in zwei Gruppen grob zusammenfassen,

1. der Konfiguration, also dem Einstellen der vorhandenen Funktionalität und
2. den Anpassungen (Berichte, Belege, Schnittstellen, Funktionen).

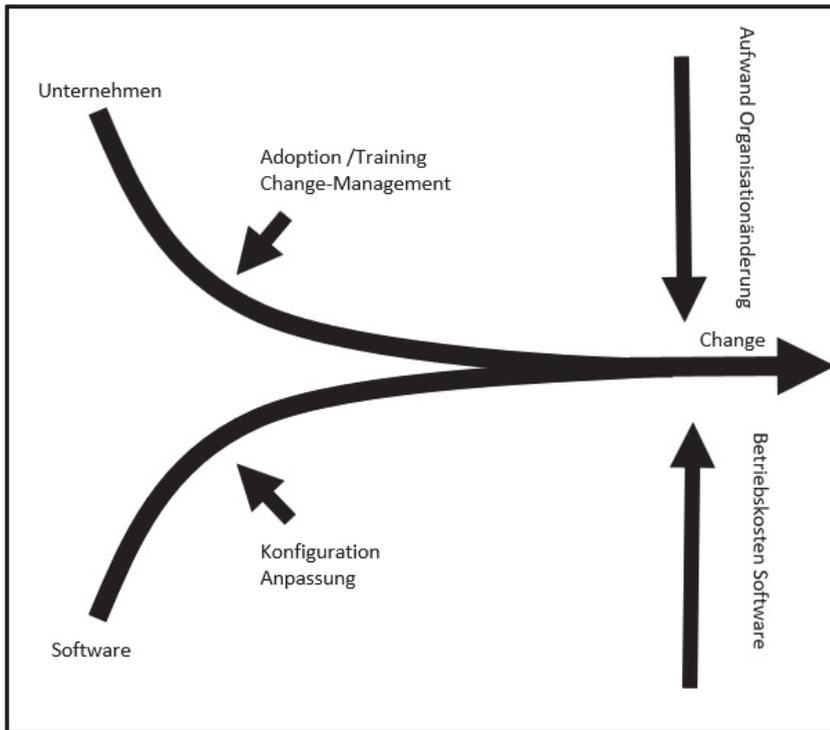


Abb. 1.1: Änderungsprozess

Diese Aktivitäten sind bei korrekter Aufnahme der Anforderungen durch erfahrende Berater in der Regel risikoarm durchzuführen. Falls es hier zu einer Kostenexplosion kommt oder Termine nicht gehalten werden können, ist dies in der Regel ein Zeichen dafür, dass nicht die richtige Balance zwischen den Änderungen an der Software und am Unternehmen gefunden wurde. Frühzeitiger Indikator ist die Gap-Liste (siehe Abschnitt 2.3.4 »Die Gap-Liste«).

1.2.2 Änderungen im Unternehmen

Die Aktivitäten im und am Unternehmen sind wesentlich komplexer. Eine neue Software verändert mit ihren neuen, anderen und ungewohnten Funktionen und

Prozessen die Arbeitsabläufe von fast allen Mitarbeitern. Das heißt, alle Mitarbeiter im Betrieb müssen sich umstellen. Sie werden teilweise ihre gewohnten Abläufe und somit Know-how verlieren und sie müssen die neuen Abläufe lernen. Dies führt insbesondere bei den erfahrenen Mitarbeitern mit viel Know-how nicht immer zu Begeisterungstürmen, denn wer fängt schon gern wieder bei null an und sei es nur in der Nutzung des IT-Systems. Hier können auch leicht Ängste oder eine Verweigerungshaltung entstehen.

Dies zu erkennen ist eine wichtige Voraussetzung, um die Implementierung erfolgreich durchzuführen, denn eine neue Software lässt sich nicht gegen die Mitarbeiter einführen.

Im Einzelnen lassen sich die Änderungsaktivitäten in

- Verstehen und Annehmen der neuen Geschäftsabläufe
- Training der Projektkernteams
- Training der End-User

gliedern.

Bei großen Veränderungen z.B. dem Wechseln von einer dezentralen zu einer zentralen Anwendung ist es empfehlenswert, diesen Prozess mit erfahrenen Beratern aus dem Themengebiet Change-Management begleiten zu lassen.

1.3 Das methodische Vorgehen

Obwohl in der Einleitung steht, dass dieses Buch keine wissenschaftliche Abhandlung über Vorgehensmodell und Methoden darstellt, so ist es doch sinnvoll, die hier beschriebene Vorgehensweise methodisch abzugrenzen. Mit methodischer Vorgehensweise beschreibe ich die Art und Weise (Methode), wie man zu einem angestrebten Ziel gelangen kann (Vorgehen).

Wie im Titel des Buches angekündigt, möchte ich hier eine Methode für eine »agile« Vorgehensweise vorstellen und diese von dem üblichen »Wasserfall-Modell« abgrenzen.

1.3.1 Warum kein Wasserfall?

In der Implementierungsphase wurde in der Vergangenheit und wird auch jetzt noch von vielen Systemhäusern und von vielen Kunden bei Eigenimplementierung eine klassische Wasserfallmethode mit den Einzelschritten

- Projektdefinition und Anforderungsanalyse,
- Entwurf mit dem Systemdesign,
- Implementierung mit Programmierung und Systemaufbau,

- Systemtest und
- Betrieb und Wartung
angewandt.

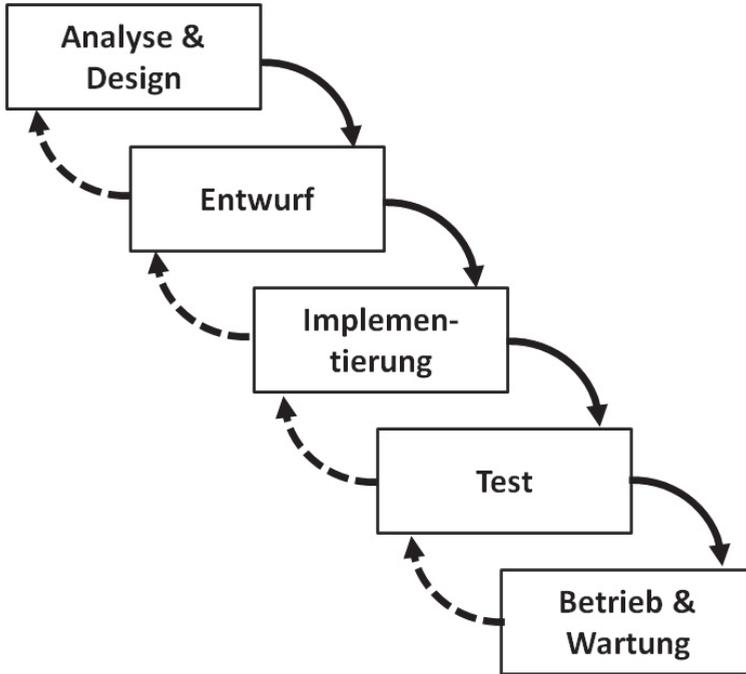


Abb. 1.2: Wasserfallmethode

In diesen Phasen wird ein System Schritt für Schritt erstellt bzw. implementiert, das den funktionalen Anforderungen, beschrieben in einem Lasten- oder Pflichtenheft, entspricht.

Diese Vorgehensweise hat allerdings einige Nachteile:

- Es gelingt selten, bereits zu Projektbeginn alle Anforderungen endgültig und theoretisch im Detail festzulegen. Daher besteht das Risiko, dass die fertiggestellte Software bzw. das implementierte System nicht den tatsächlichen realen Anforderungen entspricht. Um dem zu begegnen, wird oftmals ein unverhältnismäßig hoher Aufwand in der Analyse- und Entwurfsphase betrieben, ohne letztlich das oben genannte Risiko vollständig zu eliminieren.
- Die Wasserfall-Methode erlaubt es zwar, im laufenden Projekt Änderungen vorzunehmen (Change Requests), doch dies hat Grenzen, insbesondere dadurch, dass erst am Ende des Projekts das Ergebnis »vorzeigbar ist« und von dem Anwender beurteilt werden kann. Die fertiggestellte Software bildet folg-

lich nicht den aktuellen, sondern in der Regel den (ggf. fehlerhaften) Anforderungsstand zu Projektbeginn wieder. Da größere Softwareprojekte meist auch eine sehr lange Laufzeit haben, kann es vorkommen, dass eine neue Software bereits zum Zeitpunkt ihrer Einführung inhaltlich veraltet ist.

Diese Methode war und ist weiterhin sinnvoll, wenn ein hohes Maß an spezifischen Anforderungen bzw. Zusatzprogrammierungen zu erwarten ist, also der Gedanke der Standardsoftware in den Hintergrund tritt.

Die Alternative

Durch den erheblichen Funktionszuwachs der Standardsoftwaresysteme und den Möglichkeiten, durch Konfiguration diese Funktionen an die Anforderungen des Unternehmens anzupassen, ist es in der heutigen Zeit möglich, den Anteil der individuell erstellten Anpassungen wesentlich zu reduzieren. Dies ist deshalb wichtig, da eine Anpassung nicht nur hohe Investitionen bedeutet, sondern auch, weil sich dadurch die Wartungs- und Updatekosten des neuen Systems deutlich erhöhen.

Um zu gewährleisten bzw. zu unterstützen, dass die individuell erstellten Anpassungen so gering wie möglich gehalten werden, wird in diesem Buch eine Methode vorgestellt und beschrieben, die die funktionalen Möglichkeiten der Software in den Mittelpunkt stellt, die vom Rapid Prototype Ansatz abgeleitet ist und sich somit von der klassischen Wasserfall-Methode deutlich absetzt.

Abgeleitet vom Spiralmodell Barry W. Boehms¹ durchläuft die Methode drei Zyklen, in denen jeweils Prototypen mit unterschiedlichem Reifegrad erstellt werden.

Dabei wird der Prototyp in Test-Workshops (VWS) am Ende jedes Zyklus verifiziert. Die Workshops sind nummeriert (von 1 bis 3) und stellen den jeweiligen Reifegrad da.

Was ist daran agil?

»Agil« ist in aller Munde. Kaum eine IT-Abteilung existiert, die nicht vorgibt, »agil« zu arbeiten, so wie es sich auch kein IT-Experte leisten kann, nicht schon »agil« gearbeitet zu haben.

Hinterfragt man aber, was genau »agil« an ihrem Vorgehen ist, so gibt es häufig eine große Vielzahl von Statements.

Von »Scrum !!!«, »Pflichtenhefte brauchen wir nicht mehr« bis zu »wir brauchen keinen Plan, wir machen Sprints« reichen die Antworten. Dabei ist weder Scrum

¹ Barry W. Boehm: A Spiral Model of Software Development and Enhancement. In: IEEE Computer. Vol. 21, Ausg. 5, Mai 1988, S. 61–72.

Diese Leseprobe haben Sie beim
 **edv-buchversand.de** heruntergeladen.
Das Buch können Sie online in unserem
Shop bestellen.

[Hier zum Shop](#)