



Xpert.press

Wolfgang Osterhage

# Abnahme komplexer Software-Systeme

 Springer

Xpert.press

Die Reihe **Xpert.press** vermittelt Professionals in den Bereichen Softwareentwicklung, Internettechnologie und IT-Management aktuell und kompetent relevantes Fachwissen über Technologien und Produkte zur Entwicklung und Anwendung moderner Informationstechnologien.

Wolfgang Osterhage

# Abnahme komplexer Software-Systeme

Das Praxishandbuch



Springer

Dr. Wolfgang Osterhage  
Finkenweg 5  
53343 Wachtberg-Niederbachem  
Deutschland  
wwost@web.de

ISBN 978-3-540-68223-3

e-ISBN 978-3-540-68230-1

DOI 10.1007/978-3-540-68230-1

Xpert.press ISSN 1439-5428

Bibliografische Information der Deutschen Nationalbibliothek  
Die Deutsche Nationalbibliothek verzeichnet diese Publikation in der Deutschen Nationalbibliografie;  
detaillierte bibliografische Daten sind im Internet über <http://dnb.d-nb.de> abrufbar.

© 2009 Springer-Verlag Berlin Heidelberg

Dieses Werk ist urheberrechtlich geschützt. Die dadurch begründeten Rechte, insbesondere die der Übersetzung, des Nachdrucks, des Vortrags, der Entnahme von Abbildungen und Tabellen, der Funksendung, der Mikroverfilmung oder der Vervielfältigung auf anderen Wegen und der Speicherung in Datenverarbeitungsanlagen, bleiben, auch bei nur auszugsweiser Verwertung, vorbehalten. Eine Vervielfältigung dieses Werkes oder von Teilen dieses Werkes ist auch im Einzelfall nur in den Grenzen der gesetzlichen Bestimmungen des Urheberrechtsgesetzes der Bundesrepublik Deutschland vom 9. September 1965 in der jeweils geltenden Fassung zulässig. Sie ist grundsätzlich vergütungspflichtig. Zuwiderhandlungen unterliegen den Strafbestimmungen des Urheberrechtsgesetzes.

Die Wiedergabe von Gebrauchsnamen, Handelsnamen, Warenbezeichnungen usw. in diesem Werk berechtigt auch ohne besondere Kennzeichnung nicht zu der Annahme, dass solche Namen im Sinne der Warenzeichen- und Markenschutz-Gesetzgebung als frei zu betrachten wären und daher von jedermann benutzt werden dürften.

*Einbandgestaltung:* KuenkelLopka GmbH, Heidelberg

Gedruckt auf säurefreiem Papier

9 8 7 6 5 4 3 2 1

[springer.de](http://springer.de)

# Inhalt

<b>1 Zielsetzung</b> .....	1
1.1 Einführung .....	1
1.2 Gültigkeitsbereich .....	2
1.3 Normative Verweisungen .....	2
1.4 Methodologien .....	3
1.5 Komplexe Systeme .....	4
<b>2 Projektorganisation</b> .....	5
2.1 Qualitätsmanagement als Projekt .....	5
2.2 Vollständigkeit der Prozesse .....	6
2.3 Teilprojektstrukturen .....	7
2.3.1 Anforderungsmanagement .....	8
2.3.2 Change-Management .....	10
2.3.3 Fehlermanagement .....	11
2.4 Teilprojektautonomie .....	14
2.5 Templates .....	15
2.6 Lösungsweg .....	17
<b>3 Qualitätsmanagement im Rahmen von Abnahmen</b> .....	19
3.1 Qualitätsmanagement .....	19
3.2 Ziele des Qualitätsmanagements .....	19
3.2.1 Vermeidung von langen Pilotbetrieben .....	19
3.2.2 Vermeidung von Produktionsausfällen .....	20
3.2.3 Sofortige Korrektur von Mängeln .....	20
3.2.4 Vermeidung von Dateninkonsistenzen .....	21
3.3 Verantwortlichkeiten des Qualitätsmanagements .....	22
3.3.1 Identifikation von Testanforderungen .....	22
3.3.2 Festlegen personeller Zuständigkeiten für den Abnahmeprozess .....	23
3.3.3 Koordination von Testskripten und Testdaten .....	24
3.3.4 Gesamtplanung der Abnahmen zusammen mit dem Lieferanten .....	26
3.3.5 Durchführung der Abnahmen .....	27
3.3.6 Abschlussbewertung mit Empfehlung .....	27

3.4	Grundsätze des Abnahmeverfahrens . . . . .	28
3.4.1	Bereitstellungstermine . . . . .	28
3.4.2	Einleitung des Abnahmeverfahrens . . . . .	29
3.4.3	Problem- und Ideenspeicher . . . . .	29
3.4.4	Reviewprozess . . . . .	30
3.4.5	Patchzyklus . . . . .	30
3.4.6	Nachabnahme . . . . .	30
3.4.7	Detailaufgaben bei der Abnahme. . . . .	30
3.4.8	Betriebsbereitschaft und Probebetrieb. . . . .	33
3.5	Templates. . . . .	35
	Abnahme-Protokoll . . . . .	36
	Inhaltsverzeichnis . . . . .	37
	1 Ziel des Dokuments . . . . .	37
	2 Abnahmeverfahren Software xyz . . . . .	37
	3 Teilnehmer der Abnahme . . . . .	38
	4 Gegenstand der Abnahme . . . . .	38
	5 Ergebnis der Abnahme. . . . .	39
	6 Abnahme durch den Auftraggeber . . . . .	39
3.6	Lösungsweg. . . . .	41
<b>4</b>	<b>Teilprojektorganisation Qualitätsmanagement. . . . .</b>	<b>43</b>
4.1	Projektumfeld . . . . .	43
4.2	Kommunikation und Dokumentation . . . . .	43
4.2.1	Regelkommunikation. . . . .	43
4.2.2	Dokumentation . . . . .	46
4.3	Strukturen . . . . .	50
4.4	Abbildung der Projektstrukturen . . . . .	50
4.4.1	Budgetplanung. . . . .	51
4.5	Templates. . . . .	51
4.6	Lösungsweg. . . . .	53
<b>5</b>	<b>Abnahme-Richtlinie . . . . .</b>	<b>55</b>
5.1	Sinnfälligkeit . . . . .	55
5.2	Verhältnis zwischen den Vertragspartnern . . . . .	56
5.3	Konsensfindung. . . . .	56
5.4	Die Richtlinie (Beispiel) . . . . .	57
5.4.1	Einführung . . . . .	57
5.4.2	Begriffsdefinitionen . . . . .	57
5.4.3	Mitwirkungsleistungen . . . . .	58
5.4.4	Abnahmebereitschaft und Zeitplan . . . . .	59
5.4.5	Aufbau der Tests . . . . .	60
5.4.6	Abnahmeobjekt: xyz n.1 . . . . .	62
5.4.7	Abnahmegegenstand . . . . .	62
5.4.8	Voraussetzungen . . . . .	63

5.4.9	Durchführung . . . . .	63
5.4.10	Abnahmeprüfungen . . . . .	64
5.4.11	Anhänge/Checklisten . . . . .	65
5.4.12	Bereitstellungsprotokoll. . . . .	66
5.4.13	Abnahmeprotokoll . . . . .	66
5.5	Templates. . . . .	66
5.6	Lösungsweg. . . . .	69
<b>6</b>	<b>Methodik Daten- und Software-Migration . . . . .</b>	<b>71</b>
6.1	Abnahme und Daten . . . . .	71
6.2	Migrationsvorhaben. . . . .	72
6.3	Planung und Durchführung . . . . .	73
6.4	Aktivitätentracking . . . . .	75
6.5	Templates. . . . .	76
6.6	Lösungsweg. . . . .	78
<b>7</b>	<b>Sonderfälle . . . . .</b>	<b>79</b>
7.1	Abnahme von Schnittstellen . . . . .	79
7.1.1	Sonderfall Schnittstellen . . . . .	79
7.1.2	Vollständigkeit der Testlandschaft . . . . .	79
7.1.3	Eigenständiges Verfahren. . . . .	80
7.1.4	Simulationen und mehrstufige Verfahren . . . . .	80
7.1.5	Lösungsweg . . . . .	84
7.2	Customizing von Testsystemen . . . . .	84
7.2.1	Simulation der Produktionsumgebung . . . . .	84
7.2.2	Fachliches Know-how . . . . .	84
7.2.3	Eigenständiger Meilenstein . . . . .	85
7.2.4	Einbindung der Fachbereiche . . . . .	86
7.2.5	Templates. . . . .	89
7.2.6	Lösungsweg . . . . .	89
7.3	Aufbau eines Testmandanten. . . . .	89
7.3.1	Technologische Voraussetzungen . . . . .	89
7.3.2	Geeignete Umgebung . . . . .	90
7.3.3	Eigenständiger Meilenstein . . . . .	90
7.3.4	Standbau . . . . .	92
7.3.5	Templates. . . . .	97
7.3.6	Lösungsweg . . . . .	98
7.4	Integrationstests. . . . .	98
7.4.1	Prozesstests . . . . .	98
7.4.2	Verfügbarkeit von Partnersystemen . . . . .	99
7.4.3	Integriertes Testvorgehen. . . . .	100
7.4.4	Integrierte Testskripte . . . . .	100
7.4.5	Templates. . . . .	101
7.4.6	Lösungsweg . . . . .	102

7.5	Performance	102
7.5.1	Performance als Akzeptanzkriterium	102
7.5.2	Benchmarks	102
7.5.3	Performanceläufe	103
7.5.4	Monitoring	103
7.5.5	Templates	107
7.5.6	Lösungsweg	108
<b>8</b>	<b>Dokumentation</b>	<b>109</b>
8.1	Dokumentationsarten	109
8.2	Vollständigkeit und Relevanz	111
8.2.1	Vollständigkeit	111
8.2.2	Relevanz	111
8.3	Verantwortlichkeiten	112
8.4	Versionierung und Freigaben	112
8.5	Templates	113
<b>9</b>	<b>Sicherheitsaspekte</b>	<b>115</b>
9.1	IT-Sicherheit als Teil des Qualitätsmanagements	115
9.1.1	Gültigkeitsbereich	115
9.2	Normative Verweisungen	116
9.2.1	Gesetzliche Vorschriften	117
9.2.2	Richtlinien und Normen	117
9.3	Strategische Voraussetzungen	118
9.4	Einbindung der Mitarbeiterschaft	120
9.4.1	Nachhaltigkeit	120
9.5	Templates	121
9.6	Lösungsweg	122
<b>10</b>	<b>Projektmanagement</b>	<b>123</b>
10.1	Technische und organisatorische Hilfsmittel	123
10.2	Zielsetzung und Aufgaben	124
10.3	Möglichkeiten und Umfang	124
10.4	PMS-Funktionalitäten	125
10.4.1	Verwaltung von Vorgängen	125
10.4.2	Gantt-Diagramm	127
10.4.3	Meilensteinplan	128
10.4.4	Kapazitäten	129
10.4.5	Netzplan	129
10.4.6	Erweiterte Funktionalitäten	130
10.5	Templates	132
10.6	Lösungsweg	132

<b>11 Projektcontrolling</b> .....	133
11.1 Budgetvorgaben .....	133
11.2 Besonderheiten .....	133
11.3 Objekte .....	134
11.3.1 Direkte IT-Kosten .....	134
11.3.2 Entwicklungskosten .....	134
11.3.3 Kosten, die bei den Fachbereichen anfallen .....	135
11.3.4 Projektmanagement .....	135
11.4 Prozessuale Ansätze .....	135
11.4.1 Strategische Ansätze .....	136
11.4.2 Prozessuale Ansätze .....	137
11.4.3 Berichtswesen .....	139
11.4.4 Steuerung .....	140
11.4.5 Übergabe an die Linie .....	141
11.5 Templates .....	142
11.6 Lösungsweg .....	142
<b>Quellenangaben</b> .....	143
<b>Sachverzeichnis</b> .....	145

# Kapitel 1

## Zielsetzung

### 1.1 Einführung

Dieses Buch ist aus der Praxis gewachsen und soll der Praxis dienen. Dahinter stehen Erfahrungen aus fast zwanzig Jahren Test und Abnahme, während deren mittlere bis große IT-Systeme und komplexe Software in Verwaltung und Industrie in vielen Ländern eingeführt, getestet und letztendlich abgenommen wurden. Bekanntlich stehen sich wandelnde Methodologien unterschiedlicher Ausprägung aus unterschiedlichen Beratungs- oder Lehrstuhlkulturen zur Verfügung, um solche Aufgaben zu unterstützen. Es ist aber ebenso offensichtlich, dass auf Grund unterschiedlicher Organisationsstände in den Unternehmen die Stringenz zu Gunsten pragmatischer Ansätze leiden muss. Im Interesse einer wirtschaftlichen Zielerreichung lässt sich das oftmals nicht vermeiden. Deshalb soll hier keine neuartige oder konsolidierte Methodologie vorgestellt werden, sondern ein praxisnahes Vorgehen, das der Wirklichkeit nahe kommt und Ansätze anbietet, die je für sich genommen Hilfestellung geben.

Nachdem hier die Zielsetzung und eine Eingrenzung vorgegeben werden, erfolgt der Einstieg über mögliche Projektorganisationen, die für solch komplexe Aufgaben geeignet sind. Innerhalb der Gesamtprojektorganisation findet sich auch das IT-Qualitätsmanagement. Die Einführung geschieht hier über die Aufgaben dieser Funktion im Rahmen von Software-Abnahmen. Das IT-Qualitätsmanagement gliedert sich vor diesem Hintergrund in Teilverantwortlichkeiten.

Im Weiteren werden die Grundsätze des Abnahmeverfahrens dargelegt, die in eine beispielhafte Abnahmerichtlinie münden, die als Muster für konkrete Konstellationen zwischen Kunde und Software-Lieferant dienen kann.

Da häufig mit Systemeinführungen Datenmigrationen verbunden sind, werden die wesentlichen Punkte zur Beachtung in einem gesonderten Kapitel dargestellt. Weitere Sonderfälle bilden die Abnahme von Schnittstellen, das Customizing, der Aufbau eines Testmandanten und Performance. Schließlich wird auf das Thema Dokumentation im Rahmen von Software-Lieferungen eingegangen. Darüber hinaus werden der Vollständigkeit halber die klassischen Tools des Projektmanagement und das IT-Projektcontrolling vorgestellt.

Jedes Kapitel mit Ausnahme dieser Einführung und den Anlagen ist gleichartig aufgebaut:

- Hinführung
- Problemstellung
- Organisatorische Umsetzung
- Instrumente
- Templates
- Zusammenfassung Lösungsweg

## 1.2 Gültigkeitsbereich

Der Gültigkeitsbereich des Handbuchs bzw. einiger Vorlagen wird durch zwei Begrenzungen charakterisiert:

- systemisch und
- zeitlich.

Die systemische Gültigkeit bezieht sich auf die Komplexität und den Anpassungsgrad von Software. Natürlich kann man die vorgeschlagenen Methoden auf jede Art von Software anwenden. Aus Gründen der Wirtschaftlichkeit empfiehlt sich es sich jedoch, bei kleineren und Standardanwendungen nur begrenzt Gebrauch davon zu machen.

Bei zeitlich begrenzter Gültigkeit handelt es sich normalerweise um Versionsstände abgestimmter Dokumente, wie z. B. der Abnahme-Richtlinie oder Teile der unter „Dokumentation“ (Kap. 8) beschriebenen Unterlagen. Jedes Dokument hat eine Versionsnummer, die auf das Hauptdokument verweist. Die Gültigkeitsaussage bezieht sich dann auf die aktuelle Version, in Ausnahmefällen auf Abschnitte von Vorgängerversionen. Auf jeden Fall gilt das letzte Update. Hierzu gehören auch Aussagen, wie mit Änderungen zu den einzelnen Dokumenten zu verfahren ist. Die Änderungen sind in einer Versionshistorie bis zur endgültigen Freigabe festzuhalten.

Spätestens nach wichtigen technologischen Entwicklungen ist auch der Inhalt dieses Handbuchs auf seine Aktualität zu überprüfen.

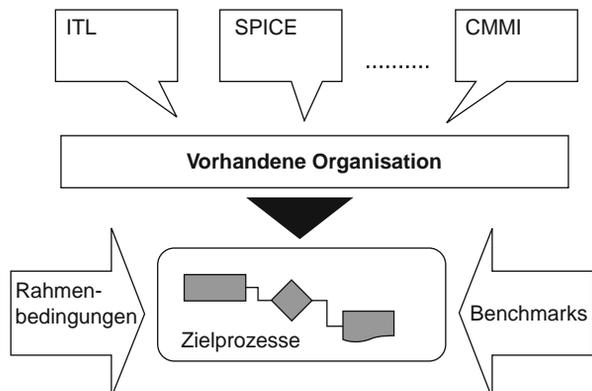
## 1.3 Normative Verweisungen

Neben den allgemeingültigen Qualitätsnormen der DIN-ISO-Familie (9000ff) soll an dieser Stelle hingewiesen sein auf die Standards des Bundesamts für Sicherheit in der Informationstechnik (BSI). Obwohl der Schwerpunkt des IT Grundschutz-Katalogs auf Fragen der IT-Sicherheit liegt, finden sich wertvolle Hinweise unter G 2.26 „Fehlendes oder unzureichendes Test- und Freigabeverfahren“ sowie unter S 2.83 „Testen von Standardsoftware“.

## 1.4 Methodologien

Für die Einführung komplexer Systeme wird normalerweise ein eigenes Projekt aufgesetzt, bzw. die Struktur eines schon vorhandenen Projektes genutzt. Es gibt nun eine Reihe von Methodologien, in die auch die Abnahmeverfahren eingebettet werden können. Dazu gehören z.B. CMMI (Capability Maturity Model Integration), SPICE (Software Process and Capability Determination) und ITIL. CMMI dient zur Beurteilung und Verbesserung der Qualität im Produkt-Entwicklungsprozess, ist also der eigentlichen Qualitätsprüfung aus Kundensicht vorgeschaltet, deckt allerdings auch Teile der Vorlaufstrecke wie z.B. Anforderungsmanagement ab. Bei SPICE handelt es sich um eine echte Norm (ISO/IEC 15504), die zunächst allgemein Unternehmensprozesse bewertet, deren Schwerpunkt aber auf der Software-Entwicklung liegt. Ein Teil davon beschäftigt sich auch mit Kunden-Lieferantenprozessen. Die ITIL Bibliothek schließlich hat sich zu einem Referenzwerk und damit zu Quasi-Standards entwickelt und deckt unter vielem Anderen auch solche Prozesse wie Release- und Change-Management ab. Auf die genannten und andere Methodologien bzw. Normen soll hier nicht im Detail eingegangen werden.

Neben diesen mittlerweile klassischen Methodologien wie ITIL oder SPICE zeigt sich allerdings in der Praxis, dass häufig nur Versatzstücke davon zum Einsatz kommen. Das liegt an den gewachsenen Strukturen und Prozessen in Unternehmen. Eine Komplettumstellung z.B. nach ITIL vor Projektstart erweist sich häufig als schwierig wegen der damit verbundenen Kosten oder aus Zeitdruck. Dieses Buch liefert sozusagen eine neutrale Vorgehensweise, die praxiserprobt ist. Dabei kann es durchaus vorkommen, dass Elemente aus dem einen oder anderen Regelwerk erscheinen. Schließlich gibt es nur eine endliche Palette von Vorgehensmöglichkeiten bei identischer Problemstellung. Die Abb. 1.1 zeigt schematisch die Zusammenhänge zwischen den Standard-Methodologien und dem vorgestellten Ansatz.



**Abb. 1.1** Praxiserprobte Vorgehensweise

## 1.5 Komplexe Systeme

Komplexe IT-Systeme gehen in der Regel über singuläre Standardprodukte hinaus; d. h. sie enthalten Vernetzungen zwischen verschiedenen Systemen, die jedes für sich je eigene Aufgaben im Geschäftsgeschehen von Unternehmen wahrzunehmen haben. Das Funktionieren solcher Vernetzungen muss über spezielle Schnittstellen sichergestellt werden. Außerdem enthalten komplexe IT-Systeme häufig Software-Komponenten, die für die Bedarfe eines Unternehmens angepasst oder gar neu entwickelt wurden. Auf jeden Fall muss man davon ausgehen, dass solche Systeme Gegenstand von Customizing-Vorgaben und Parametrisierungen sind.

Zu diesen Systemen gehören z. B. große betriebswirtschaftliche und technische Anwendungen. Genannt seien hier ERP, CRM, PPS, Lagerwirtschaftssysteme, Qualitätsmanagementsysteme, Instandhaltungsmanagementsysteme, Systeme zur Unterstützung der Geschäftsprozesse in der Elektrizitätsversorgungswirtschaft, Abrechnungssysteme der Telekommunikation aus dem betriebswirtschaftlichen Bereich. Im technischen Feld kann man beispielhaft nennen: CAD, Berechnungen mit der Finite-Elemente-Methode, Steuerungssysteme, Simulationen, Modellrechnungen in der Nuklearphysik.

Wesentlich ist, dass zum einen ein signifikanter Anteil von Customizing vorliegt (bis hin zu Erstentwicklungen), und dass ein Komplexitätsgrad aus funktionalem Volumen, Anzahl Anwender, Schnittstellenvernetzungen und Datenvolumen vorliegt. Sind diese Voraussetzungen nicht oder nur unzureichend gegeben, werden die vorgeschlagenen Maßnahmen wirtschaftlich nur schwer zu rechtfertigen sein.

# Kapitel 2

## Projektorganisation

### 2.1 Qualitätsmanagement als Projekt

Grundsätzlich ist zu unterscheiden zwischen der Linienfunktion „IT-Qualitätsmanagement“ und einem Qualitätsmanagement-Projekt. Abnahmen neuer Software oder neuer Releases oder Teile daraus haben typischen Projektcharakter:

- definierter Anfangszeitpunkt,
- definierter Endzeitpunkt,
- klar umschlossener Inhalt und
- die Beteiligung von dafür speziell abgestellten Fachbereichsressourcen.

Zudem besteht die Möglichkeit, dass Abnahmen als Teilprojekt in einem größeren Projektzusammenhang von z. B. Restrukturierungsmaßnahmen stehen können. Die Leitung eines solchen Projektes bzw. Teilprojektes wird in die Hand der Organisationseinheit „IT-Qualitätsmanagement“ gelegt, sofern das Unternehmen ein solches besitzt. Ansonsten muss für dieses Projekt eine temporäre Struktur geschaffen werden. Gibt es bereits ein ähnliches Projekt, können unter Umständen seine Strukturen genutzt werden.

In den folgenden Ausführungen wird davon ausgegangen, dass die Entwicklungsorganisation des Softwarelieferanten ein eigenes Qualitätswesen unterhält. Dieses Qualitätswesen sorgt dafür, dass nur so genannte qualitätsgesicherte Software-Komponenten zur Auslieferung an den Kunden gelangen. Das bedeutet:

- durch Entwickler getestete und
- eindeutig versionierte, durch den Kunden beauftragte Module.

Als Pendant dazu wird auf der Kundenseite eine Kompetenz geschaffen, die ihrerseits sicherstellt, dass „geliefert wird, wie bestellt worden ist“. Ähnlich wie bei einem Wareneingang werden also Bestellung und Lieferung miteinander abgeglichen und eventuelle Mängel festgehalten bzw. deren Nachbesserung eingefordert. Grundlage der Bestellung sind gemeinsam vereinbarte Lastenhefte. Während die interne Qualitätssicherung des Lieferanten für ihre Belange eine Art „Werksabnahme“ durchführt, nimmt das kundenseitige Qualitätsmanagement die ausge-

lieferte Software ab mit dem Ziel einer qualitätsgesicherten, unmittelbar auf den Abnahmeprozess folgenden Inbetriebnahme.

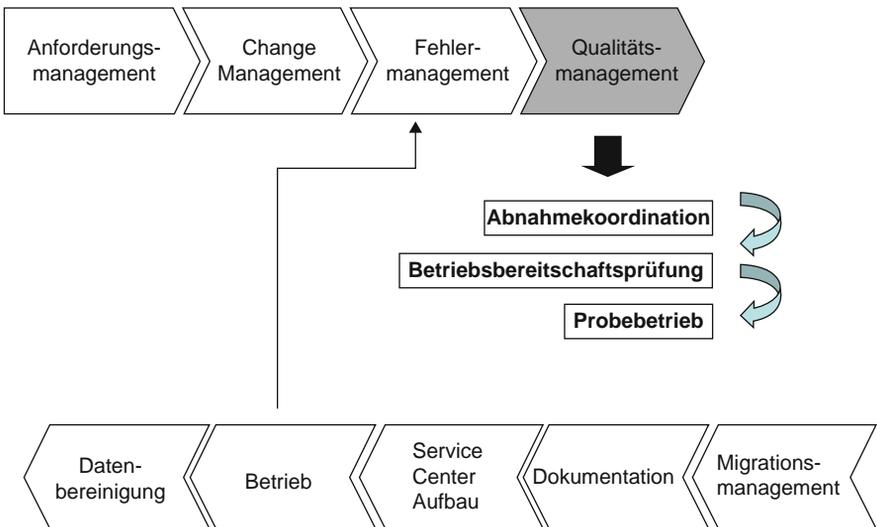
Das Qualitätsmanagement des Kunden erfüllt damit nicht nur seine klassische technische Kernaufgabe, sondern gibt letztendlich auch die Rechnung des Lieferanten zur Zahlung durch den Kunden frei. Insofern kommt ihm innerhalb des Einführungsprojektes eine strategische Funktion zu. Es muss deshalb einen entsprechenden Platz in Organisation und Prozessgeschehen des Projekts finden.

## 2.2 Vollständigkeit der Prozesse

Die Leitung des Einführungsprojektes, in der sowohl Entscheider des Lieferanten wie auch des Kunden vertreten sind, stellt sicher, dass die gesamte Kette zwischen ursprünglicher Anforderung bis zur Fehler- und Datenbereinigung, die über den Inbetriebnahmezeitpunkt hinaus gehen kann, zeitgerecht und unter Berücksichtigung der vereinbarten Qualitätskriterien abgearbeitet wird. Dabei ist es unerheblich, ob es sich z. B. um komplette Releases oder nur um die Umsetzung von z. B. Change Requests handelt. Der Ablauf bleibt grundsätzlich der gleiche.

Im Prozess zu berücksichtigen (s. Abb. 2.1) sind die Folgeschritte:

- Anforderungsmanagement,
- Change Management,
- Fehlermanagement,
- Qualitätsmanagement,
- Migrationsmanagement,



**Abb. 2.1** Aufgaben aus Sicht der Projektleitung

- Dokumentation,
- Betrieb,
- Datenbereinigung.

Zum Qualitätsmanagement wiederum gehören:

- die Abnahmekoordination,
- die Betriebsbereitschaftsprüfung und
- der Probetrieb.

Die in Abb. 2.1 sequentiell dargestellten Funktionsblöcke werden in der Regel überlappend oder auch Phasen verschoben wahrgenommen. Die Sequenz in der Darstellung unterstreicht lediglich die logische Verkettung. Einzelne Aufgaben erfahren naturgemäß differenzierte Ausprägungen entsprechend des letztendlichen Abnahmegegenstandes. Ein punktuell umgesetzter Change Request wird von der Intensität her über den planerischen Aufwand bis hin zum Einspielen zur Produktivsetzung eine andere Aufmerksamkeit beanspruchen als ein komplettes Release.

## 2.3 Teilprojektstrukturen

Entsprechend der oben geschilderten Aufgabeninhalte werden innerhalb eines Einführungsprojektes Zuständigkeiten vergeben, die strukturell als Teilprojekte ausgewiesen werden können (s. Abb. 2.2):

In dieser Darstellung sind die Aufgaben

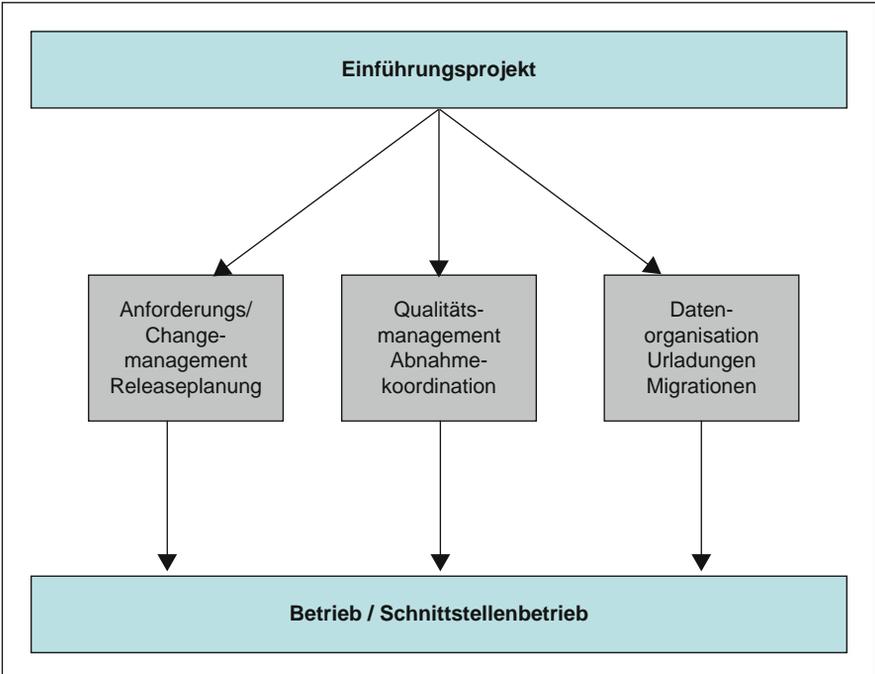
- Anforderungsmanagement
- Change Management
- Fehlermanagement

mit der übergeordneten Releaseplanung zusammengefasst. Das braucht nicht immer so zu sein. So könnte z. B. das Fehlermanagement in einer eigenen Verantwortung liegen.

Das Qualitätsmanagement beinhaltet die schon in Abb. 2.1 herunter gebrochenen Verantwortlichkeiten. Hier wäre die Herauslösung eines einzelnen Aufgabenbereiches nicht sinnvoll.

Alle Problemfelder, die mit Datenhaltung, Datenqualität etc. zu tun haben, sind unter einer einzigen Teilprojektverantwortung zusammen gefasst.

Betrieb, Service Center und Dokumentation sind hier nicht als separate Teilprojekte ausgewiesen. Sie können entweder einem der drei Zuständigkeitsbereiche zugeordnet werden – z.B. Dokumentation dem Qualitätsmanagement – oder aber als Linienfunktion projekt-mäßig einbezogen werden (Service Center und Betrieb). Ausdrücklich erwähnt wird an dieser Stelle der Schnittstellenbetrieb. Der Grund dafür liegt in dem relativ häufigen Erfordernis, bestimmte Schnittstellen Enduser-unabhängig über eine separate Jobsteuerung durch z. B. ein Service Center abzuwickeln.



**Abb. 2.2** Teilprojektorganisation

Die in den Abb. 2.1 und 2.2 aufgeführten Prozesse und Verantwortlichkeiten sind allerdings nur ein Ausschnitt aus dem gesamten Projektgeschehen. Entwicklung, Projektplanung und -controlling und andere Funktionen sind bewusst ausgeblendet, da an dieser Stelle nur qualitätssicherungsnahe Funktionen betrachtet werden sollen. Diese werden im Folgenden in der Reihenfolge ihrer logischen Verkettung abgehandelt.

### 2.3.1 Anforderungsmanagement

Das Anforderungsmanagement bündelt möglichst alle Anforderungen zu einem zu realisierenden oder zu erweiternden Software-Paket innerhalb des Einführungsprojektes. Diese Anforderungen können aus unterschiedlichen Quellen kommen:

- Beschreibung des Leistungsumfanges des gesamten Paketes oder wesentlicher Teile (Module) daraus; dabei kann es sich um Workshopergebnisse handeln oder auch um komplette Prozessbeschreibungen der relevanten Geschäftsvorfälle in einem Unternehmen,
- Wünsche aus dem Benutzerumfeld nach neuen Funktionalitäten,
- Wünsche aus dem Benutzerumfeld nach spezifischen Funktionserweiterungen,

- Erkenntnisse und Vorschläge des Lieferanten zur Arbeitserleichterung mit seinem System,
- technische Erfordernisse aus dem Umgang mit der vorhandenen Software.

Das Anforderungsmanagement sammelt all diese Anforderungen. Sie können sporadisch einlaufen oder aufgrund von Workshop-Abstimmungen zur Vorbereitung eines größeren Updates oder eines kompletten Releases entstehen. Bewährt hat sich die strukturierte Zusammenfassung aller Anforderungen in einer Datenbank mit entsprechender Statusverfolgung, wie

- offen,
- beauftragt,
- in Realisierung,
- abgenommen,
- in Betrieb.

Bevor konkrete Anforderungen weiterverfolgt werden können, sollten sie in Form von Lastenheften formuliert worden sein. Dabei empfiehlt sich eine einheitliche Struktur. Erst nach Vorliegen eines Lastenheftes kann eine erste Aufwandschätzung erfolgen. Die tatsächliche Beauftragung zur Realisierung durch die Projektleitung ist natürlich abhängig von Priorisierungen und Budgetrestriktionen.

Nach der Beauftragung kann der Lieferant das korrespondierende Pflichtenheft erstellen. Während das Lastenheft den Funktionsinhalt abdeckt (das „Was“), beschäftigt sich das Pflichtenheft mit der Art der technischen Umsetzung (das „Wie“). Damit ergeben sich zwei zusätzliche Statusinformationen für Anforderungen:

- Lastenheft vorhanden und abgenommen,
- Pflichtenheft vorhanden und abgenommen.

Die jeweiligen Abnahmen sollten durch Vertreter des Fachbereichs des Kunden erfolgen.

Um das der Realisierung nachfolgende Abnahmegeschehen handhabbar zu gestalten, sollten Einzelanforderungen gesammelt werden und zu einem festgesetzten Meilenstein als Release bzw. Teilrelease (Version) zusammengefasst und gemeinsam zur Abnahme bereitgestellt werden. Dieses Prinzip lässt sich nicht immer durchhalten. Aufgrund von Dringlichkeiten, die technisch-inhaltlicher Art und auch von außerhalb des Projektes beeinflusst sein können, besteht immer die Möglichkeit von zwischenzeitlichen Sonderauslieferungen einzelner Funktionalitäten. Grundsätzlich jedoch sollte das Anforderungsmanagement zusammen mit dem Lieferanten einen verbindlichen Releaseplan erstellen mit vereinbartem Horizont und mit den Releases als Meilensteine auf der Zeitachse.

Zur Verfolgung des jeweiligen Anforderungsstatus sind abgestimmte Kommunikations- und Rückmeldeprozesse zwischen dem Anforderungsmanagement und vor- bzw. nachgelagerten Bereichen erforderlich, so insbesondere zu:

- Benutzern/Anforderern,
- Entwicklungsabteilung des Lieferanten,
- Qualitätsmanagement/Abnahmekoordination,
- Betriebsunterstützung.

### 2.3.1.1 Ausschreibungsverfahren

Zusammen mit dem Einkauf, den IT-Verantwortlichen und den Fachbereichen sollte das Anforderungsmanagement in das Ausschreibungsverfahren für neue Software bzw. Softwareerweiterungen mit einbezogen werden. An dieser Stelle soll der kommerzielle Auswahlprozess nicht behandelt werden. Wichtig ist, dass eine Bepreisung und die Vereinbarung von Lieferterminen nur auf der Basis einer allseits akzeptierten Dokumentation erfolgen können. Diese Dokumente in Form von Lasten- und Pflichtenheften sind bereits weiter oben erwähnt worden.

### 2.3.2 Change-Management

Unterhalb der Organisation des eigentlichen Anforderungsmanagements wird häufig das sog. Change Management angesiedelt, da es ähnlichen Prozessen folgt (s. Abb. 2.3).

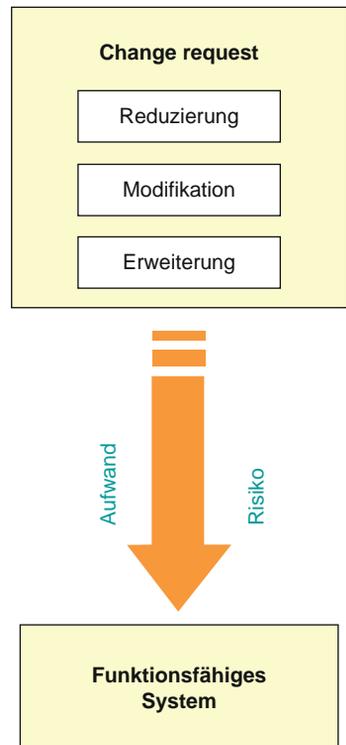


Abb. 2.3 Change-Management

Aufgrund von Erkenntnissen im Umgang mit im Betrieb befindlicher Software kommt es häufig zu sog. Change Requests, also Einzelanforderungen, die Änderungen an vorhandenen Funktionalitäten verlangen. Dabei kann es sich um eine Reduzierung des bestehenden Funktionsumfangs handeln, weil z. B. das Geschäftsfeld sich vereinfacht hat. Oder es geht um Modifikationen z. B. auf Grund von Prozessänderungen. Die Verursacher für solche Change Requests finden sich

- im Benutzerumfeld oder
- in der Entwicklungsabteilung des Lieferanten.

Bevor solche Anfragen in die Datenbank des Anforderungsmanagements zur Weiterverfolgung eingestellt werden, wird eine sorgfältige Aufwands-Risiko-Betrachtung durchgeführt, wobei sich das Risiko auf systemtechnische und fachliche Gegebenheiten bezieht, aber auch Budgetrisiken d. h. Kosten berücksichtigt werden müssen. Unstrittig sind sicherlich solche Anforderungen, die für das korrekte Funktionieren des Geschäftsprozesses inklusive des technischen Funktionierens des Systems von Bedeutung sind. Nicht immer werden bei der ursprünglichen Lastenhefterstellung alle Konsequenzen im Voraus durchdacht, und prozessuale Inkonsistenzen werden erst bei der intensiven Nutzung des umgesetzten Produktes sichtbar!

Die Behandlung der Change Requests folgt den gleichen Richtlinien wie den übrigen Anforderungen innerhalb des Anforderungsmanagements. Aufgrund von unterschiedlichen Priorisierungen kann es aber häufiger zu Realisierungen und Inbetriebnahmen zwischen den längerfristig geplanten Release-Meilensteinen kommen.

### 2.3.3 Fehlermanagement

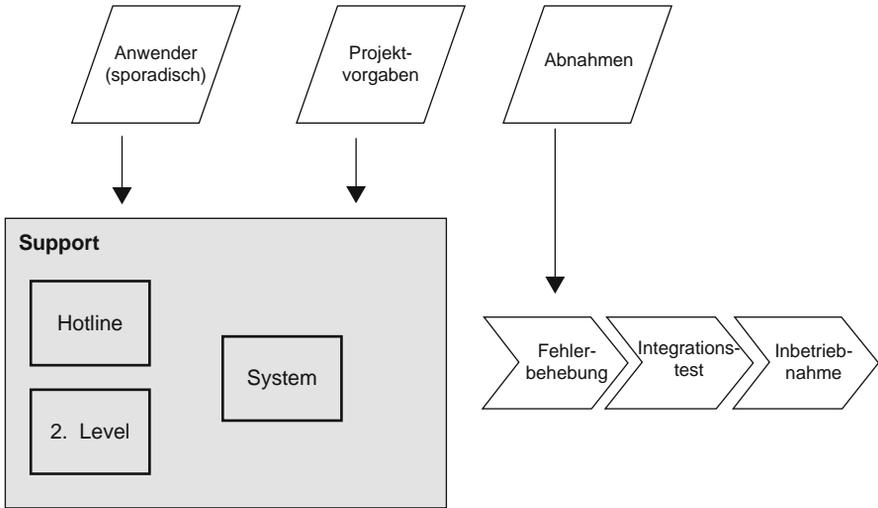
Software fehler werden in unterschiedlichen Konstellationen erkannt und weitergemeldet:

- bei der Nutzung der Anwendung im Tagesbetrieb durch die Enduser,
- bei Weiterentwicklungen im Projekt, die auf schon vorhandene Funktionalitäten aufsetzen und
- bei den Abnahmen vor der Inbetriebnahme, d. h. im Zuge der eigentlichen Qualitätssicherung.

Abbildung 2.4 zeigt den prinzipiellen Verfahrensweg nach der Fehlererkennung.

Mit Ausnahme der während einer Abnahme erkannten Fehler durchlaufen alle anderen eine Supportschleife, die normalerweise mit der Aktivierung einer Hotline, hinter der sich ein Call Center verbirgt, beginnt. Der 2nd Level Support greift erst, wenn die Hotline nicht in der Lage ist, einen nutzbaren Beitrag zur Fehlerbehebung zu leisten (dabei muss bedacht werden, dass sich häufig vom Enduser als Fehler gemeldete Probleme später als Bedienungsfehler herausstellen; hier sollen nur echte Softwarefehler behandelt werden). Die Rückmeldung der technischen Instanzen soll hier nicht weiter verfolgt werden.

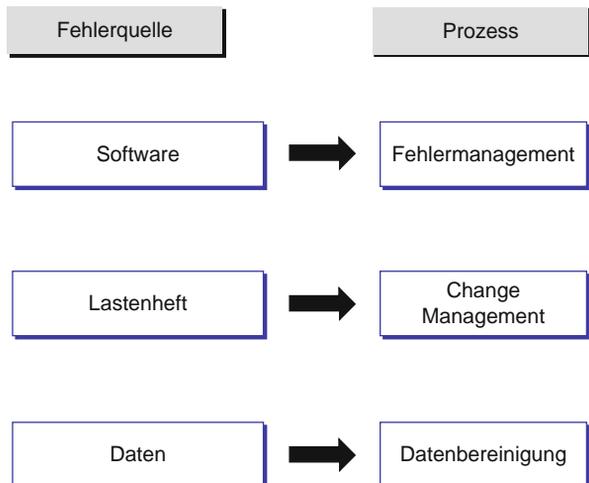
**Initiatoren:**



**Abb. 2.4** Fehlermanagement

Die Behandlung von Fehlern aus Abnahmen wird weiter unten erläutert. Das betrifft gleichermaßen die Auslieferung von Korrekturen während des Abnahmeprozesses.

Alle Korrekturen, die auf Grund von Fehlermeldungen vorgenommen werden und zur Auslieferung kommen, unterliegen – ähnlich den Bereitstellungen aus dem Change-Management oder den übrigen Abnahmegegenständen – dem gleichen Abnahmeprozedere, wenn auch möglicherweise in verkürzter Form. Die folgende Abb. 2.5 differenziert noch einmal nach Fehlertypen und zugehörigem Reinigungsprozess.



**Abb. 2.5** Fehlerdifferenzierung

Was subjektiv als Fehler erscheint, kann auf verschiedene Ursachen zurückzuführen sein:

- echte Softwarefehler, die bei der Umsetzung vom Pflichtenheft auf den Code zustande gekommen sind,
- korrekt realisierte Anforderungen, die aber in der Anwendung zu fachlich falschen Ergebnissen führen (Algorithmen, Plausibilitätsprüfungen etc.),
- Datenfehler, die z. B. durch Übernahme von Daten aus anderen, vernetzten Systemen entstehen können; hier gibt es eine Vielfalt von Typisierungen, die an dieser Stelle nicht weiter behandelt werden.

Je nach Fehlertypus durchläuft der Korrekturprozess dann eine eigene Schleife:

- Fehlerbereinigung wie oben beschrieben,
- als Change Request zur Erzeugung einer modifizierten oder neuen Anforderung oder
- Weitergabe an das Teilprojekt, das sich mit Datenbereinigung befasst.

### 2.3.3.1 Technisches Fehlermanagement

Im Falle eines Problems ruft der User bei der Service-Hotline (1st Level Support) an (s. Abb. 2.6): Wird das Problem als Fehler erkannt, und kann dieser Fehler nicht

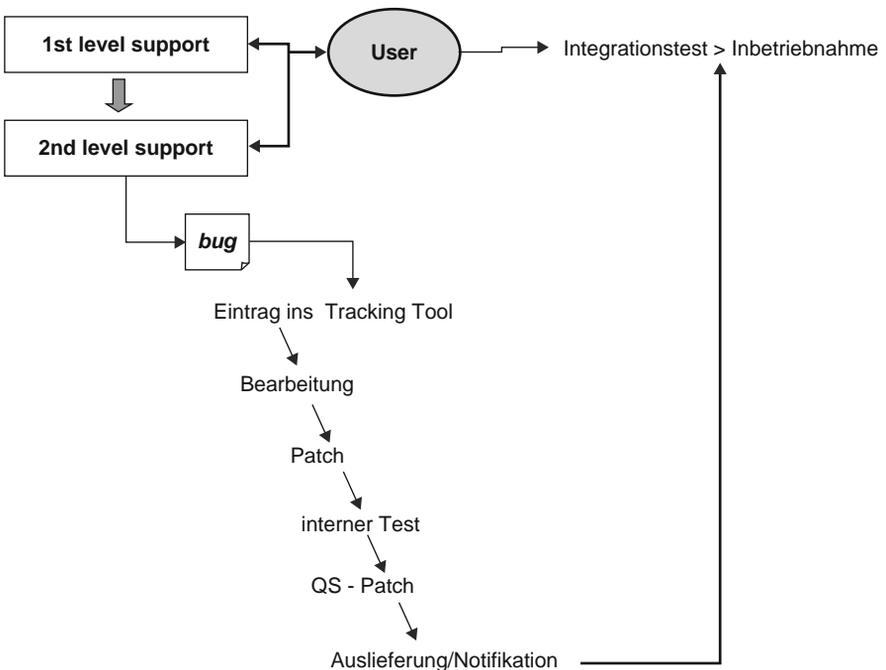


Abb. 2.6 Fehlerprozess

während des Gesprächs behoben werden, erhält der Anrufer eine Referenz-Nr. Die Hotline reicht das Problem mit einer entsprechenden Beschreibung an den 2nd Level Support weiter. Nach Überprüfung und Reproduktion des Fehlers erfolgt ein Eintrag ins Tracking Tool (s. Abb. 2.7).

Beim Tracking Tool handelt es sich um ein Software gestütztes Werkzeug zur Verfolgung von Fehlerbearbeitungen. In diesem Werkzeug sollten mindestens die folgenden Felder gepflegt werden:

- *Bewertungsstatus*: offen, abgelehnt, geschlossen
- *Bearbeitungsstatus*: erfasst, in Bearbeitung, realisiert, qualitätsgesichert, ausgeliefert
- *Modul*: Identifizierung des betroffenen Teils des Software – Pakets
- *Typ*: Fehler, Erweiterung
- *gemeldet von*: Name des ersten Problem Melders (User)
- *aktueller Bearbeiter*: Person, die gerade das Problem behandelt (Owner)
- *Beschreibung*: Kurzttext
- *Priorität*: 1, 2, 3
- *ID*: numerische Identifikationsnr.

Ist der Eintrag ins Tool erfolgt, beginnt die Bearbeitung entsprechend der Arbeitsplanung der Entwicklungsabteilung bzw. bei Dringlichkeit (Priorität „1“) nach Vereinbarung mit dem Kunden. Steht Bearbeitungsstatus auf „realisiert“, wird ein erster Patch zum internen Test auf ein Entwicklungssystem ausgebracht. Ist dieser erfolgreich, wird ein qualitätsgesicherter Patch auf das Testsystem des Kunden geliefert, der Bearbeitungsstatus rutscht auf „qualitätsgesichert“ und der Kunde wird zur Abnahme benachrichtigt.

Jetzt greifen die üblichen Abnahmeverfahren. Nach erfolgreicher Abnahme wird der neue Softwarestand in Betrieb genommen. Der Bearbeitungsstatus geht auf „ausgeliefert“ und der Bewertungsstatus auf „geschlossen“.

## 2.4 Teilprojektautonomie

Die unter 2.2 gelisteten Teilprojekte müssen einerseits untereinander in Phase sein und bestimmte formale Gemeinsamkeiten aufweisen. Andererseits macht es wenig Sinn, sie zentral bis ins Detail zu steuern. Hier sollte das Subsidiaritätsprinzip

Bewertungsstatus	Bearbeitungsstatus	Modul	Typ	gemeldet von	aktueller Bearbeiter	Beschreibung	Prio	ID
offen	erfasst	xyz	Fehler	Müller	Schneider	Kurzttext	2	#123

**Abb. 2.7** Tracking Tool

gelten, nachdem sowohl Sachverstand als auch Verantwortung auf der Durchführungsebene liegen. Es reicht deshalb aus, wenn bezüglich der folgenden Elemente Einverständnis herrscht:

- Gesamtmeilensteinplan
- Budget auf die einzelnen Teilprojekte verteilt
- Verantwortlichkeiten
- Berichtswesen.

Den Teilprojekten bleibt, die Abarbeitung ihrer Aufgaben eigenständig zu organisieren. Die Teilprojektverantwortlichen setzen dazu die ihnen als geeignet erscheinenden Instrumente ein. Es muss sicher gestellt sein, dass an den Schnittstellen zu den anderen Teilprojekten entsprechende Formate eingehalten werden. Das lässt sich unter Umständen durch interne Service Level Agreements (SLA) regeln. Selbstverständlich unterliegt das Gesamtprojekt einem zentralen Controlling, das insbesondere für die Budgeteinhaltung sorgt.

In größeren Organisationen bleiben die meisten Teilprojektfunktionen über das Projektende hinaus organisatorisch erhalten, um in Folgeprojekten zur Geltung zu kommen. In kleineren Unternehmen werden Aufgaben wie Migration und Datenbereinigung mit Projektende abgeschlossen sein, um bei Bedarf später wieder reaktiviert zu werden. Dabei ist zu berücksichtigen, dass aufgebautes Know-how unter Umständen verloren geht, wenn externes Personal freigestellt wird.

## 2.5 Templates

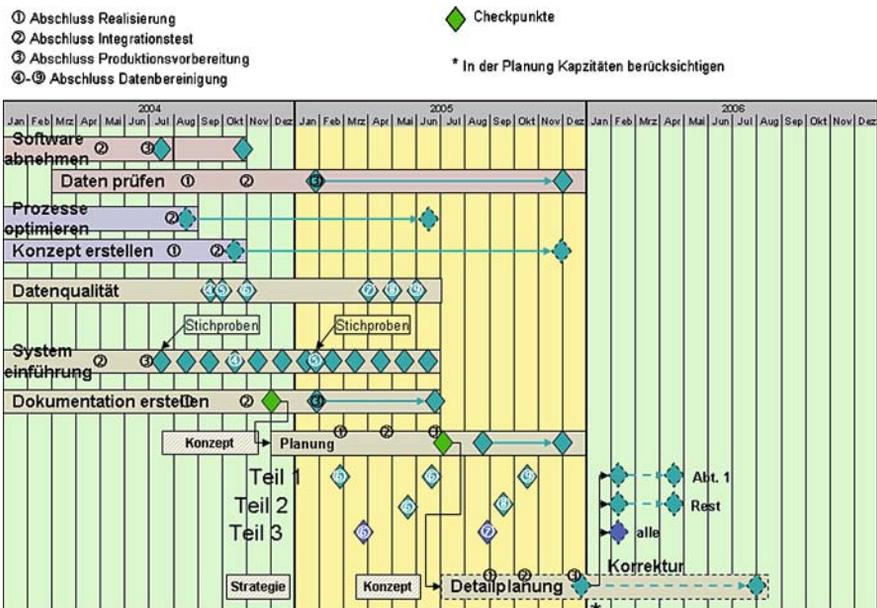


Abb. 2.8 Meilensteinplan

Monat1		Anforderung	Anf.-Nr.	Budget (PT)
	Name1			
	Name2			
	Name3			
<b>Monat2</b>	Name1			
	Name2			
	Name3			
.....				
<b>Monatx</b>	Name1			
	Name2			
	Name3			
<b>Summe</b>	Name1			
	Name2			
	Name3			

Abb. 2.9 Budgetplan

**Projektstatus: Projektbezeichnung**

**Berichtsdatum:**

Meilenstein	Erfüllungsgrad	Probleme	Maßnahmen
Bezeichnung Aktivität1 Aktivität2			
Bezeichnung Aktivität1 Aktivität2 ...			
Bezeichnung Aktivität1 Aktivität2 ...			
Bezeichnung Aktivität1 Aktivität2 ...			

Abb. 2.10 Statusbericht

## **2.6 Lösungsweg**

Ausrichtung des Abnahmevorhabens als Projekt > Sicherstellen, dass alle Teilprozesse in den Vor- und Nachlaufstrecken berücksichtigt werden > Teilprojekte möglichst autonom laufen lassen > konsolidiertes Berichtswesen einführen.

# Kapitel 3

## Qualitätsmanagement im Rahmen von Abnahmen

### 3.1 Qualitätsmanagement

Obwohl die im vorhergehenden Kapitel abgehandelten Prozesse nicht zur Kernkompetenz des Qualitätsmanagements gehören, sind sie doch in essentieller Weise als Vorlaufstrecke bzw. über Rückkopplung mit ihm verbunden. Im Gesamtzusammenhang stellt das Qualitätsmanagement ein eigenständiges Teilprojekt dar. Es verfügt über ein entsprechendes internes Berichtswesen mit Schnittstelle zum Gesamtprojekt. Es wird im Folgenden dargestellt nach

- Zielsetzung,
- Verantwortlichkeiten,
- Detailaufgaben.

### 3.2 Ziele des Qualitätsmanagements

Ganz allgemein dient das kundenseitige Qualitätsmanagement bei der Einführung komplexer IT-Systeme der Sicherstellung einer hohen Qualität der realisierten Software inklusive der Schnittstellen vor der Auslieferung zur Inbetriebnahme. Alle Industrienormen, die dazu entwickelt wurden, und die teilweise oder in ihrer Gänze dafür relevant sind, finden dabei ihre Berücksichtigung. Die Einhaltung von DIN ist jedoch kein Ziel in sich selbst, sondern folgt einer Interessenlage und damit wirtschaftlichen Überlegungen, der die Einhaltung solcher Vorschriften entgegen kommen. Insofern lassen sich die allgemeinen Ziele wie folgt konkretisieren:

#### 3.2.1 Vermeidung von langen Pilotbetrieben

Zu der Praxis wird weiter unten Stellung genommen. Das Teilziel selbst orientiert sich an Aufwandsreduzierung (Kosten) und Engpassbeseitigung (Meilenstein bezo-

gen). Software wird häufig unter hohem zeitlichem Druck entwickelt. Gründe dafür liegen in der Vorgabe, dass bestimmte Geschäftsvorfälle zügig und konkurrenzfähig am Markt zum Einsatz kommen müssen. Von dem auf diese Weise vorgegebenen Meilenstein wird sodann rückwärts terminiert, um den Start der Inbetriebnahme und damit des Entwicklungsbeginns und anderer Elemente der Vorlaufstrecke (Lastenheft, Pflichtenheft) zu ermitteln. Da jedoch nur endliche Entwicklungskapazität zur Verfügung steht, entsteht jener Zeitdruck fast zwangsläufig.

Eine Möglichkeit der Reduzierung auf der Zeitskala besteht eben nun im Vermeiden langer Pilotbetriebszeiträume. Pilotbetrieb meint hier die Simulation des Echtbetriebes auf einem möglichst gespiegelten Abnahmesystem mit einem Komplettabzug der Echtdata entweder vor (besser) Inbetriebnahme der Software oder parallel dazu. Ein Pilotbetrieb benötigt bis zur endgültigen Freigabe nicht nur Kalenderzeit, sondern bindet auch in hohem Maße Entwicklungs-, Qualitäts- und ganz besonders Enduserressourcen.

Einen entscheidenden Beitrag, Zeiten und Ressourcen zu optimieren, leistet das Qualitätsmanagement, indem es dafür sorgt, möglichst fehlerfreie Software in den Pilotbetrieb zu geben, sofern er dann noch für notwendig erachtet wird. Im Idealfalle sollte der Abnahmeprozess ausreichen, sodass der Pilotbetrieb gänzlich entfallen kann.

### ***3.2.2 Vermeidung von Produktionsausfällen***

Mit oder ohne Pilotbetrieb geht es bei diesem Teilziel um die Abwendung von Konsequenzen aus der Inbetriebnahme einer unzureichend getesteten Software. Diese Selbstverständlichkeit ist nicht so banal, wie sie klingt. Nicht immer sind komplette Systemausfälle erforderlich, um das Tagesgeschäft zum Erliegen zu bringen. Falsche Bildschirmsteuerung z. B., systematische Datenfehler aus Schnittstellenübertragungen oder ein kritischer Umrechnungsalgorithmus reichen aus, um das Arbeitsgebiet eines einzelnen Sachbearbeiters zu blockieren. Und gerade solche Fehler fallen häufig beim Entwicklertest nicht auf.

Eine saubere Auswahl des Testdatenbettes sowie sorgfältige und vollständige Entwicklung von Testskripten sorgen für die Prüfung einer maximal denkbaren Variantenzahl für einen Geschäftsvorfall. Wie überall gilt hier ganz besonders: die Arbeit muss ohnehin getan werden. Was im Vorfeld an Aufwand nicht geleistet wird, kostet im Nachhinein ein Vielfaches davon!

### ***3.2.3 Sofortige Korrektur von Mängeln***

Die Ausführungen zum Fehlermanagement zeigen klar auf, dass die Schleife der Fehlererkennung und -korrektur im Zuge der Abnahme durch das Qualitätsmanagement die kostengünstigste und beherrschbarste ist. Sowohl die Identifikation

als auch die Steuerung der einzuleitenden Maßnahmen aus einer Fehlererkennung aus dem Betrieb heraus sind kommunikationsaufwendig. Dem gegenüber erfolgt die Fehlererkennung während einer Abnahme durch Projektfachleute, die meistens auch an der Abfassung der Lastenhefte mitgewirkt haben. Häufig haben diese Spezialisten auch den direkten Draht zu den zugehörigen Entwicklern, sodass inhaltliche Klärungen rasch erfolgen können.

Ganz besonders hilfreich ist der für eine Abnahme im Vorhinein festgelegte Patchprozess, der praktisch eine unmittelbare Fehlerkorrektur (mit geringen zeitlichen Einschränkungen für den Testbetrieb) durch einen Hotfix oder Ähnliches ermöglicht. Dem kann dann sofort die Nachabnahme dieser Korrektur folgen.

Software-Korrekturen, die aus Erkenntnissen aus dem Betrieb selbst heraus erfolgen, unterliegen dagegen dem Planungsprozess von Updates. Eine zeitnahe Umsetzung ist deshalb nur in Ausnahmefällen, z. B. bei extremen Störungen, machbar – mit allen Unwägbarkeiten für den laufenden Betrieb durch das Einspielen und dann Anwenden von unerprobten Änderungen. Bei Änderungen am Datenmodell sind längere Ausfallzeiten des Regelbetriebes unvermeidlich.

### **3.2.4 Vermeidung von Dateninkonsistenzen**

Beim Zusammenschalten von unterschiedlichen Systemen, die jedes für sich konkrete Geschäftsbereiche eines Unternehmens abdecken und dabei aus Notwendigkeit oder Rationalisierungsgründen Daten austauschen oder auf eine gemeinsame Datenbasis zugreifen müssen, stellt sich das Problem der Datenkonsistenz. Bei komplett neu entwickelten Systemen sollte es gering sein oder überhaupt nicht auftreten. Es potenziert sich allerdings, wenn ein oder mehrere Altsysteme in den Verbund treten, da jedes System – und damit die zugehörigen Datentöpfe – eine eigene Geschichte haben.

Im späteren Kapitel über Datenmigration (Kap. 6) wird auf die Präventivmaßnahmen eingegangen. Unabdingbar bleibt jedoch ein sorgfältig vorbereiteter Schnittstellentest möglichst mit realen Daten, bei dem nicht nur die Schnittstellenfunktionalität sichergestellt wird, sondern auch die Datenqualität im Sinne von Konsistenz bzw. Rückweisungsquoten. Datenbereinigungsprojekte, die im Ernstfall später auf Produktionsdatenbestände angewendet werden müssen, weil im Vorfeld keine ausreichende Prüfung geschah, sind enorm kostenintensiv und dauern sehr lange. Fast immer laufen solche Maßnahmen dem sich während des Betriebs ständig neu wandelndem Bestand hinterher. Ganz teuer kann es werden, wenn Daten betroffen sind, die nach außen sichtbar werden, wie z. B. Kundenrechnungen im Massengeschäft.

Zusammenfassend lässt sich feststellen:

Wirtschaftliches Ziel eines Qualitätsmanagements bei der Einführung komplexer IT-Systeme soll die Vermeidung von erhöhtem Folgeaufwand sein. Erfahrungen zeigen, dass unterlassene Investitionen im Vorfeld nicht selten Nachbesserungen nach sich ziehen, die 200% und mehr der Kosten für Präventivmaßnahmen betragen.

### 3.3 Verantwortlichkeiten des Qualitätsmanagements

Grob umschrieben liegt die Hauptverantwortlichkeit des Qualitätsmanagements auf der Planung und Durchführung von Abnahmetests vor Auslieferung einer Software oder von Schnittstellen zur Inbetriebnahme. Sämtliche anderen Verantwortlichkeiten sind entweder peripher oder haben Regelcharakter. In letzteren Fällen wirken sie in vor und nach geschaltete Prozesse hinein, so wie oben und im Weiteren unten beschrieben. Erleichtert wird die Wahrnehmung der Gesamtverantwortlichkeiten des Qualitätsmanagements durch die Verabschiedung einer Abnahmespezifikation mit allgemeiner Gültigkeit innerhalb des Einführungsprojektes.

Wie schon angedeutet, übernimmt das Qualitätsmanagement zunächst die Funktion einer Wareneingangsprüfung nach der Regel:

1. Bestellte Ware = Pflichtenheft,
2. Gelieferte Ware = Software.

Abweichungen werden als Fehler dokumentiert, bevor über eine Zurückweisung der Auslieferung oder Teilen davon entschieden wird.

Die Verantwortlichkeit einer banalen Wareneingangsprüfung wird dann überschritten, wenn der Qualitätssicherung auch die Aufgabe zugeordnet wird, solange die Nachlieferungen in Form von Korrekturen zu betreuen, bis der gesamte ursprünglich bestellte Lieferumfang fehlerfrei läuft.

Die Verantwortlichkeiten im Einzelnen gliedern sich in folgende Schritte:

- Identifikation von Testanforderungen,
- Festlegen personeller Zuständigkeiten für den Abnahmeprozess,
- Koordination von Testskripten und Testdaten,
- Gesamtplanung der Abnahmen zusammen mit dem Lieferanten,
- Durchführung der Abnahmen und
- Abschlussbewertung mit Empfehlung.

#### 3.3.1 Identifikation von Testanforderungen

Je nach Abnahmegegenstand

- Release,
- Sonderfunktionen/einzelne Change Requests,
- Fehlerkorrektur,
- Datenmigration.

gibt es unterschiedliche Anforderungen an einen Abnahmetest. Die Anforderungen gliedern sich zunächst nach den erforderlichen Ressourcen:

- Hardware und Systemumgebung, und hier wiederum nach

1. Serverlandschaft mit entsprechender Performance für die Aufgabe,
  2. Konfiguration inklusive aller Sub- und Dienssysteme, die erforderlich sind, um die zu testende Software ohne Konfigurationsfehler laufen zu lassen,
  3. ausreichendem Plattenspeicher,
  4. Anzahl und Typ von Endgeräten und deren Lokalisation, ggf. die spezielle Einrichtung von Remote-Zugriffen über Netzwerke,
  5. sowie Planung und Vorbereitung der notwendigen Testlokalitäten;
- Testpersonal aus dem Enduserumfeld mit entsprechenden Freistellungsvereinbarungen durch die Fachbereichsverantwortlichen für den Testzeitraum, Spezialisten aus dem Einführungsprojekt, die an der Erstellung von Lastenheften beteiligt waren, und Support vom Lieferanten – meistens Entwickler, die an der Realisierung beteiligt waren;
  - Zuordnung von abzunehmenden Funktionalitäten zu Spezialisten, die in der Lage sind, Testskripte dafür zu erstellen;
  - Identifikation und Anforderung von Testdaten, insbesondere die Grundentscheidung, ob mit realen und/oder synthetischen Daten getestet werden soll;
  - Anforderung von Schnittstellen, deren Input/Output für bestimmte Funktionsnachweise erforderlich sind, unter Umständen auch Schnittstellensimulationen z. B. über Logfiles, falls erforderlich.

Die Gesamtheit dieser aufgelisteten Ressourcen muss von der Abnahmekoordination, die gleichzeitig Teilprojektleitung Qualitätsmanagement sein kann, berücksichtigt werden und im Detail mit einem ausreichenden zeitlichen Vorlauf vor Abnahmebeginn bzgl. Zielterminen und verantwortlichen Personen verplant werden.

### **3.3.2 *Festlegen personeller Zuständigkeiten für den Abnahmeprozess***

Nach der grundsätzlichen Identifikation der Personalressourcen müssen nunmehr die Einzelaufgaben mit Namen (nicht mit Organisationseinheiten!) hinterlegt werden. Sofern diese Dokumente noch nicht vorliegen, müssen Autoren für Lastenhefte und Co-Autoren für Pflichtenhefte (mit dem Lieferanten zusammen) benannt werden. Innerhalb eines beordneten Projektes kann man jedoch erwarten, dass solche Dokumente vor dem Beginn der eigentlichen Abnahme vorliegen und freigegeben sind.

Häufig sind die ausgesuchten Personen, die an den Spezifikationen mitarbeiten, gleichzeitig federführend bei der Erstellung von Testskripten. An dieser Stelle müssen unbedingt Vertreter des Fachbereichs mit hinzugezogen werden. Mit der Erstellung von Testskripten geht die Festlegung von Testdaten einher. Die Autoren der Testskripte werden dann auch die fachliche Abnahme ihrer speziellen Funktionalitäten durchführen. Unterstützt werden sie dabei vom Fachbereich. Falls noch nicht geschehen, muss die Teilprojektleitung Qualitätsmanagement jetzt eine möglichst

schriftliche Vereinbarung mit den Fachbereichsverantwortlichen für die Freistellung der betroffenen Personen für den Testzeitraum und den Abnahmeort treffen. Das ist besonders wichtig bei Abnahmen, die nicht am Arbeitsplatz der betroffenen Personen stattfinden. Letzteres empfiehlt sich sogar, da dann die Ablenkung durch das Tagesgeschäft gering ist.

Der Lieferant sollte aufgefordert werden, für den Abnahmezeitraum Entwicklungsressourcen für die entsprechenden Funktionalitäten zu mindest im Standby-Modus bereitzustellen. Außerdem ist die Beteiligung der Lieferanten-internen Qualitätssicherung erforderlich, sollten während des Abnahmezeitraums Fehlerpatches oder Hotfixes eingespielt werden müssen.

Je nach Zuständigkeit müssen Systemspezialisten für den Betrieb der Testumgebung benannt und eingeplant werden.

Sofern nicht über die eigentliche Struktur des Einführungsprojektes bereits festgelegt, müssen Eskalationspfade und -instanzen nominiert werden – und das aus drei Gründen:

- um Entscheidungen im Konfliktfall bzgl. der Klassifizierung von Fehlern zu treffen,
- um den Abbruch oder die Weiterführung einer Abnahme zu regeln, sollten schwerwiegende Störungen auftreten (Systemprobleme, Softwarestillstand, Engpässe bei personellen Ressourcen etc.) und
- als Adresse für den Abnahmebericht.

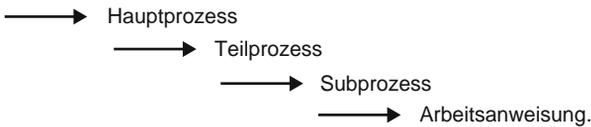
### **3.3.3 Koordination von Testskripten und Testdaten**

Testskripte orientieren sich an drei Quellen:

- Lastenheft,
- Pflichtenheft,
- Geschäftsprozess.

Alle drei hängen voneinander ab. Man sollte annehmen, dass Lastenhefte auf Basis dokumentierter Geschäftsprozesse erstellt werden. Diese wiederum sind Grundlage für die von den Entwicklern zu erstellenden Pflichtenhefte. So ist zunächst die Theorie. Nur, wenn diese Voraussetzungen gegeben sind, können Testskripte letztendlich auf der alleinigen Basis von Geschäftsvorfällen entwickelt werden. Leider finden sich in den wenigsten Unternehmen vollständig und aktuell dokumentierte Geschäftsprozesse. Für Einzelfunktionen, deren Entwicklung möglicherweise auch noch aus Ad-hoc-Erkenntnissen über die Nutzung der vorhandenen Software getrieben wird, ist das noch weniger der Fall. Man muss deshalb von einer ziemlich heterogenen Dokumentenhistorie ausgehen, wenn dann unter Zeitdruck plötzlich Abnahmen erfolgen sollen.

Der Lieferant wird sich bei seiner Rechnungsstellung immer auf den letztgültigen Stand des entsprechenden Pflichtenheftes beziehen. Das ist rechtlich völlig

**Die Zerlegung des Gesamtprozesses folgt der Hierarchie:**

Die Ebene Arbeitsanweisung lässt ihrerseits Varianten über die Variablen „Ausgangssituation“ und „Input“ zu.

Die Kombination dieser drei Felder wird über eine laufende Nummer, die der Prozesshierarchie folgt, eindeutig identifizierbar.

**Akzeptanzkriterien**

Für jeden identifizierten Testfall existieren Akzeptanzkriterien, die ihrerseits zerlegt sind in:

- erwarteter Output und
- Ziel ( bezogen auf die Arbeitsanweisung ).

**Abb. 3.1** Prozesshierarchie

in Ordnung. Der Enduser wird allerdings seine Zustimmung nur geben, wenn die neue Funktionalität seinen fachlichen Prozess so unterstützt, wie er es als richtig empfindet. Dieser Forderung ist Priorität einzuräumen. Um hier keinen Konflikt aufkommen zu lassen, ist die Hinzuziehung aller Betroffenen notwendig. Gegebenenfalls sind Kompromisse zu schließen, sollte es trotz aller Kommunikation bei der Erstellung der Spezifikationen zu Meinungsverschiedenheiten kommen. Denn – das Softwareprodukt ist jetzt da.

Auf keinen Fall sollte sich der Kunde damit begnügen, die Testskripte aus den Entwicklungstests des Lieferanten zu übernehmen, da diese rein technisch ausgelegt sind und den Geschäftsprozess nicht im Blick haben. Wichtig ist an dieser Stelle, festzuhalten, dass auf der Verrichtungsebene getestet werden muss. Diese folgt der Hierarchie in Abb. 3.1.

Bei der Abb. 3.1 handelt es sich um ein Grundsatzschema. Je nach Komplexität des Hauptprozesses kann die Hierarchisierung ausgeprägter oder flacher sein. Entscheidend ist die unterste Ebene der Arbeitsanweisung (Verrichtungsebene), an der sich die Testskripte zu orientieren haben.

Bei der Festlegung von Testdaten werden naturgemäß die Erfordernisse der Testskripte berücksichtigt – insbesondere, was die Kombinatorik der Datenfeldvarianten betrifft. Daraus leiten sich die inhaltlichen Anforderungen ab. Zum einen muss abgeprüft werden, ob sich diese durch einen Auszug aus der Produktivdatenbank abdecken lassen. In diesem Falle wird zu entscheiden sein, ob partielle Auszüge ausreichen, oder ob die Produktivdatenbank komplett abgebildet werden muss. Zu beachten sind hier möglicherweise die Kapazitätsgrenzen des Testsystems und die zu erwartende Performance bei großen Datenmengen.

Wenn es sich bei den abzunehmenden Funktionalitäten darum handelt, auch neue Daten zu erzeugen oder neue Algorithmen zu prüfen, reichen reale Daten oft nicht aus. In diesem Falle müssen synthetische Daten angelegt werden – oftmals in einer leeren Datenbank. Wenn solche Anforderungen über einige Dutzend Datensätze hinausgehen, muss man den Aufwand einplanen, der für den Aufbau komplexer und zahlreicher synthetischer Daten anfällt. Wenn möglich, sollte man große Mengen von synthetischen Daten über eigens geschriebene Programme generieren.

Ein weiterer Gesichtspunkt bei der Testdatenauswahl ist die Performance. Wie bereits erwähnt, sollten negative Performanceeinflüsse das Abnahmegeschehen nicht unnötig aufhalten. Andererseits gibt es durchaus Testskripte, die explizit das Performanceverhalten bestimmter Funktionalitäten zum Gegenstand haben – insbesondere natürlich dann, wenn Performance-Verbesserungen durch z. B. Umstellungen am Datenmodell angesagt sind.

### ***3.3.4 Gesamtplanung der Abnahmen zusammen mit dem Lieferanten***

Einem größeren Einführungsprojekt liegt normalerweise ein abgestimmter Gesamtprojektplan mit allen Aktivitäten und Meilensteinen zu Grunde. Softwareabnahmen – sofern es sich um Releases handelt – erscheinen dort allerhöchstens als individuelle Meilensteine, andere Abnahmegegenstände sind überhaupt nicht abgebildet, es sein denn, es handelt sich um unternehmensstrategische Funktionen. Das bedeutet, dass um die Abnahmen herum eine zusätzliche tagesgenaue Detailplanung erfolgen muss. Da darin auch Ressourcen des Lieferanten eingebunden werden sollen, ist eine enge Abstimmung mit dessen Gegebenheiten unerlässlich.

Die Abnahmeplanung beinhaltet nicht nur die Eckdaten von Bereitstellung und Abnahmezeitraum als solche, sondern hat als Planungsobjekte jede zu testende Funktionalität mit den dazugehörigen Fachressourcen. Bestimmte Konstellationen können dazu führen, dass ein paralleles Testen von spezifischen Funktionalitäten auf ein und derselben Datenbasis ausgeschlossen werden muss (konkurrierende Zugriffe, Locking, Gefahr der Korruption der Datenbank etc.). Dafür müssen also separate Zeitfenster vorgesehen werden. Das planerische Ergebnis, das zunächst für den gesamten Abnahmezeitraum verbindlich ist, ordnet also jede Ressource einer Funktionalität und einem Testzeitfenster zu. Im Laufe des Abnahmegeschehens kann es allerdings auf Grund z. B. von Nachtests korrigierter Funktionalitäten, die als fehlerhaft erkannt worden waren, zu Verschiebungen kommen.

Das Format eines Abnahmeplans ist zweitrangig. Nur bei Erstabnahmen umfangreicher Software wird man einen durchterminierten Projektplan entwickeln. Aber auch hierbei kann man Teilbereiche recht einfach in EXCEL-sheets darstellen. Entscheidend sind Übersichtlichkeit und leichte Pflege.

### ***3.3.5 Durchführung der Abnahmen***

Bei dieser Kernverantwortlichkeit ist die Akzeptanz bei allen Beteiligten – Fachbereich, Projekt, Lieferant – wichtig. Dazu gehört eine entsprechende Rückendeckung durch die Gesamtprojektleitung und eine damit einhergehende Vorabkommunikation. Neben diesen Flankierungen sollte der verantwortliche Funktionsträger fachlich und sozial anerkannt sein, Durchsetzungsvermögen besitzen, sowie diplomatisches Geschick zur Herbeiführung von Kompromissen haben. Wichtig ist es, eine zufrieden stellende Software-Lösung in guter Qualität und zeitgerecht dem Unternehmen zur Verfügung zu stellen, sodass der Lieferant sein Geld bekommen kann. Dies ist das gemeinsame Ziel, dem ein kompromissloser Perfektionismus nicht entgegenstehen sollte.

### ***3.3.6 Abschlussbewertung mit Empfehlung***

Der Entwicklung einer Software geht eine Beauftragung zur Realisierung voraus, es sei denn, es handelt sich um einen Standard, mit dem ein Lieferant Marktanteile gewinnen möchte. Ansonsten erfolgt die Beauftragung in Form eines Vertrages, der gleichzeitig Bestellung ist. In diesem Vertrag sind Auslieferungszustand und sonstige Eigenschaften der Software festgeschrieben sowie Zahlungsmodalitäten. Dieser Vertrag ist somit ein Dokument mit juristischer Relevanz.

Die Begleichung der gelieferten Leistung kann unmittelbar mit Bezug zu einem solchen Vertrag erfolgen. Revisionssicherheit erfordert jedoch, dass diese Leistungserbringung in irgendeiner Form dokumentiert wird. Das geschieht am Besten an der Stelle und zu dem Zeitpunkt, an dem die Korrektheit der Lieferung festgestellt wird – im vorliegenden Falle am Ende einer Abnahme durch ein entsprechendes Protokoll.

Einzelheiten dazu werden weiter unten abgehandelt. Unterschieden werden muss jedoch zwischen einer eher technischen Bewertung durch ein Abnahmeprotokoll mit Fehlerdokumentation und einer Schlussfolgerung daraus im Sinne einer Empfehlung, wie jetzt mit dem Ergebnis zu verfahren ist. Beide Gesichtspunkte können selbstverständlich in ein und demselben Dokument erscheinen. Die Abschlussbewertung einer Abnahme enthält also Referenzen zum Abnahmegegenstand, zum Abnahmeverfahren und eine Darstellung des Ergebnisses mit allen Fehlerprotokollen. Darüber hinaus kann sie nur zwei mögliche Schlussfolgerungen aussprechen:

- nicht abgenommen, oder
- abgenommen.

Im letzteren Falle ist eine Qualifizierung möglich. Es können nämlich Auflagen ausgesprochen werden, unter denen die Abnahme gilt. Hierbei handelt es sich meistens um einen zwischen Kunde und Lieferant abgestimmten Maßnahmenplan

zur Behebung weniger schwerwiegender Softwarefehler, die bis zum Ende der Abnahme noch nicht behoben sind.

### 3.4 Grundsätze des Abnahmeverfahrens

Zunächst gilt es, einige Konventionen und Regularien vorzustellen:

#### 3.4.1 *Bereitstellungstermine*

Es werden drei unterschiedliche Bereitstellungstermine unterschieden:

- BzT = Bereitstellung zum Test.

Damit ist der Zeitpunkt nach dem vorläufigen Abschluss der Entwicklungsarbeiten gemeint. Die Entwickler übergeben ihre Ergebnisse der internen Qualitätssicherung des Lieferanten. Dazu gehören technische Testskripte, die die Software auf rein funktionale Eigenschaften prüfen. Die Software befindet sich entweder auf den individuellen Entwicklungssystemen oder einem eigens für die interne (Werks)Abnahme eingerichteten Stand. Getestet wird auf Basis synthetischer Daten. Eine Teilnahme des Kunden ist formell nicht vorgesehen, kann aber aus praktischen Gründen sinnvoll sein. Wenn schon funktionale Fehler im Vorfeld bekannt werden, brauchen später Geschäftsvorfälle gar nicht erst in Angriff genommen zu werden. Der BzT-Termin ist ziemlich entkoppelt vom restlichen Geschehen, außer dass er unbedingt vor den beiden Folgeterminen zu liegen hat:

- BzA = Bereitstellung zur Abnahme.

Dieser Termin ist definitiv der erste Tag der Abnahme unter Beteiligung des Kunden. Es wird erwartet, dass alle Systeme bereit stehen, die Infrastruktur stimmt, die Testskripte vorliegen und Testdaten aufgespielt sind. Das Bereitstellungsprotokoll liegt vor.

- BzI = Bereitstellung zur Inbetriebnahme.

Die Abnahme ist erfolgt; der Ergebnisbericht liegt vor. Die neue Software kann in Betrieb gehen. Dazu ist entweder ein Releasewechsel mit allen Migrationsdetails einzuplanen (hier nicht Gegenstand) oder ein Update bereitzustellen. Das Update wird während eines Betriebsstillstands eingespielt und steht anschließend produktiv zur Verfügung. Im Interesse des Kunden sollte der BzI-Termin nach der Abnahme liegen, was manchmal jedoch nicht so stringent durchgehalten wird.

### 3.4.2 *Einleitung des Abnahmeverfahrens*

Das Abnahmeverfahren selbst beginnt weit vor dem BzA-Termin. Schon bald nach bekannt werden des funktionalen Inhalts eines Abnahmegegenstandes (Prozess: Anforderungsmanagement) sollte das Qualitätsmanagement ein Kick-off einberufen mit allen wesentlich Beteiligten. Hier werden der Status der Spezifikationsdokumente ermittelt, die Grobplanung der Abnahme vorgestellt und Verantwortlichkeiten zugeordnet. Nach Möglichkeit sollte ein Vertreter der Gesamtprojektleitung zugegen sein, um durch sein Gewicht die Bedeutung des Vorhabens zu unterstreichen. Die Ergebnisse des Kick-offs werden protokolliert und als Teil der Abnahmedokumentation archiviert.

### 3.4.3 *Problem- und Ideenspeicher*

Vor dem BzA-Termin werden zwei Austauschmedien auf der Abnahmeumgebung eingerichtet:

- ein Problemspeicher und
- ein Ideenspeicher.

Diese Medien, bei denen es sich um Word- oder EXCEL-Tabellen handeln kann, stehen allen Beteiligten des Testteams schreibberechtigt zur Verfügung. Der Zugriffspfad wird vorab bekannt gegeben.

In den Problemspeicher werden alle bei der Abnahme auftretenden Fehler oder sonstige Funktionsprobleme dokumentiert – und zwar zunächst entsprechend der subjektiven Fehlerkriterien der Tester. Der Problemspeicher ist später Grundlage für die regelmäßigen Abnahmereviews, in denen die Einträge qualifiziert werden.

Der Ideenspeicher dient zur Aufnahme von Wünschen und Verbesserungen sowie sonstigen Beobachtungen, die den Testern während der Abnahme auf- bzw. einfallen, um die Software später zu verbessern (Prozess: Change-Management). Per Definition handelt es sich also **nicht** um Fehler. Außerdem ist es wohl möglich, dass im Zuge des Reviewprozesses ursprünglich als Fehler deklarierte Einträge aus dem Problemspeicher in den Ideenspeicher wandern.

Beide – Problemspeicher und Ideenspeicher – werden nach Abschluss der Abnahme geschlossen und als Teil der Abnahmedokumentation archiviert.

SAP bietet ein eigenes Fehlermanagementtool an, den Solution Manager. Dieses Produkt sowie diejenigen anderer Hersteller soll an dieser Stelle nicht weiter behandelt werden.

### **3.4.4 *Reviewprozess***

Mit dem Beginn der Abnahme, also noch vor dem Ende des ersten Abnahmetages, spätestens jedoch bei Vorliegen erster Abnahmeergebnisse finden möglichst täglich zur gleichen Zeit Abnahmereviews unter Beteiligung des gesamten Abnahmeteams statt. Aus dieser Runde heraus wird der Abnahmefortschritt verfolgt und die Abnahme gesteuert. Während des Abschlussreviews wird über den Gesamterfolg der Abnahme entschieden und eine entsprechende Empfehlung ausgesprochen. Die Reviews benötigen außer am Anfang und am Ende der Abnahme erfahrungsgemäß wenig mehr als eine halbe Stunde.

### **3.4.5 *Patchzyklus***

Bereits beim Kick-off, wenn die Ecktermine bekannt sind, sollte mit der Qualitätssicherung des Lieferanten ein auf den Wochentag genauer Zyklus abgestimmt werden (mit schriftlichem Commitment), der festlegt, wann eine Softwarekorrektur nach Behebung von Fehlern, die während der Abnahme erkannt werden (Prozess: Fehler-Management), in Form eines Patches eingespielt wird. Die Vorabkenntnis der exakten Termine ist wichtig für die Neueinplanung von Testressourcen im Laufe der Abnahme.

### **3.4.6 *Nachabnahme***

Es macht Sinn, den gesamten für eine Abnahme zur Verfügung stehenden Zeitrahmen aufzuteilen in ein Zeitfenster für die eigentliche Abnahme und ein anschließendes für eventuelle Nachabnahmen – Verhältnis etwa 2:1 auf der Zeitachse. Anzustreben ist der vollständige Abschluss der Abnahme im ersten Zeitfenster, sodass im zweiten Teil nur noch Korrekturen nachgetestet werden müssen. Das ist insofern schwer durchzuhalten, als dass das Abnahmezeitfenster wegen des Zeitdrucks aus der Rückwärtsterminierung vom BZI meistens zu klein dimensioniert wird, sodass in den Nachabnahmezeitraum unweigerlich auch Nachläufer aus der eigentlichen Abnahme einlaufen. Die Zielplanung sollte allerdings zunächst diese Unterscheidung beinhalten, um ein Mindestmaß an Disziplin zu gewährleisten.

### **3.4.7 *Detailaufgaben bei der Abnahme***

Bei den folgenden Ausführungen kommen die Regeln der Verantwortung und die Konventionen aus dem oben Gesagten zum Tragen.

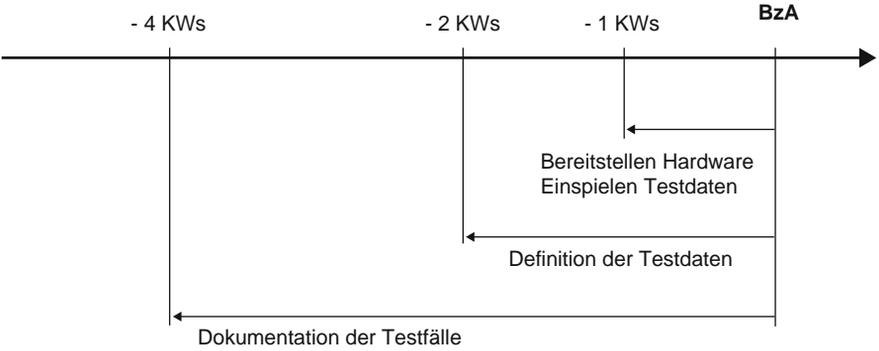


Abb. 3.2 Rückwärtsterminierung

Voraussetzung zum Kick-off ist, dass der zu testende Funktionsumfang bekannt ist. Lasten- und Pflichtenhefte sollten vorliegen. Ein Patchzyklus-Plan ist ebenfalls kommuniziert worden. Dann ergeben sich folgende planerische Sachverhalte, die in den Abb. 3.2 und 3.3 zu sehen sind.

Angelpunkt ist der BzA-Termin. Es wird von einer Dauer der eigentlichen Abnahme von drei Kalenderwochen (Beispiel) ausgegangen, wovon die letzte Woche für die Nachabnahmen reserviert ist. Nach rückwärts ergibt sich die Forderung, dass die erforderliche Systemumgebung spätestens eine Woche vor Abnahmebeginn bereitstehen muss. Dieser Puffer ist erforderlich, da es beim Aufspielen von Software und Testdaten noch zu technischen Problemen kommen kann. Des Weiteren sollten die anzulegenden Testdaten spätestens eine Woche vor Beginn der Systeminstallation bekannt sein. Das Erstellen von Testskripten benötigt Zeit. In Abb. 3.2 sind dafür vier Wochen vorgesehen. Dabei handelt es sich um eine Minimalvorgabe. Der Vorlauf kann durchaus länger sein.

Nach vorne gibt es wenig Spielraum. Abbildung 3.3 zeigt zwei hypothetische Patch- Einspieltermine. Die erste Woche kommt dabei nicht in Frage, da die Tests gerade erst anlaufen, und Korrekturprogrammierungen ebenfalls eine endliche Zeit benötigen. Da häufig der BzI-Termin kurz nach der Abnahme liegt, ist es kaum möglich, die Nachabnahme nach hinten zu verlängern.

Mit dieser Planung vor Augen können im Kick-off folgende Festlegungen getroffen werden:

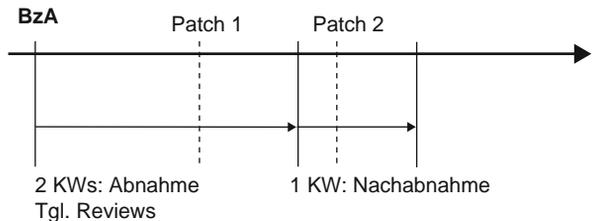


Abb. 3.3 Vorwärtsterminierung

- Zuordnung kundenseitige Ressourcen zu Testskripterstellung/Testdatenidentifikation und für die eigentliche Abnahme selbst; zugehörige Feinplanung,
- konkrete, möglichst schriftliche Anforderung an System- und Entwicklungssupport vom Lieferanten,
- konkrete Anforderung an die Testumgebung.

Die Testskripte selbst können nach dem Muster im Abschn. 3.5 (Templates, Tab. 3.3) dokumentiert werden.

Zwischen Kick-off und BzA sollte – möglichst zeitnah zum BzA-Termin – mindestens noch eine Abstimmung mit allen Beteiligten über den Stand der Abnahmepreparierungen stattfinden. Dies ist die letzte Möglichkeit zur Eskalation an Projektleitung oder Auftraggeber, wenn die Gefahr besteht, dass bestimmte Voraussetzungen nicht geschaffen werden können.

Ebenfalls noch vor Abnahmebeginn sollte zusammen mit dem Betreiber des Testsystems eine Abnahme der Systembereitstellung erfolgen, die durch ein Abnahmeprotokoll belegt wird. Das Muster für in solches Protokoll ist im Abschn. 3.5 (Templates) zu finden.

Die täglichen Reviews während der Abnahme handeln im Wesentlichen die Einträge im Problemspeicher ab:

**Tab. 3.1** Problemspeicher

Lfd. Nr.	Tester	Problembeschreibung	Fehlerklasse	Patch Datum	Status
----------	--------	---------------------	--------------	-------------	--------

–

Die Felder im Einzelnen:

- Lfd. Nr.: Sequenz nur für diese eine Abnahme, startet bei „1“
- Tester: Person, die das Problem erkannt hat und es meldet
- Problembeschreibung: möglichst ausführliche Beschreibung des Problems; Umstände, unter denen es auftrat; Fehlerbeschreibung, evtl. durch screen shots hinterlegt (Anlage)
- Fehlerklasse: Klassifizierung des Fehlers nach Abstimmung in der Reviewgruppe
- Patch-Datum: geplante Korrektur und Auslieferung während der laufenden Abnahme oder zukünftiges Korrekturdatum bei nicht so wichtigen Fehlern
- Status: Bearbeitungsstatus der Fehlerbehebung:
  1. offen,
  2. analysiert,
  3. in Arbeit,
  4. in Auslieferung,
  5. ausgeliefert,
  6. erneut abgenommen.

Entscheidend ist die Priorisierung. Im Gegensatz zu dem subjektiven Gefühl der Tester soll im Review eine Objektivierung stattfinden, die die definitiv als Fehler anerkannten Probleme in Klassen unterteilt, z. B. folgendermaßen:

- Produktionsverhindernd: die Funktionalität produziert falsche Ergebnisse, bricht ab, folgt einer unbrauchbaren Logik etc. (Prio 1),
- Produktionsverhindernd, aber temporär nutzbar durch einen Workaround (Dauer des Einsatzes dieses Workarounds muss vereinbart werden, bis der Fehler behoben ist), (Prio 2),
- alle anderen Fehler (Prio 3).

Wenn die Testsequenz es erlaubt, sollte eine fall-back-Planung erstellt werden. Eine solche Eventualität ist sinnvoll, wenn zu erwarten ist, dass das Testsystem instabil wird (Datenüberlauf, Tabellenüberlauf, sonstige Systemfehler) und erneut aufgesetzt werden muss. Die Zeitstrecke von nur drei Abnahmewochen schränkt diese Möglichkeit im Beispielfall allerdings erheblich ein.

Das Abnahmeprotokoll sollte folgende Elemente enthalten:

- Datierung,
- Gegenstand der Abnahme, evtl. Bezug zu einer Bestellung,
- Funktionalitäten im Detail (Nennung – nicht Beschreibung),
- Abnahmeumgebung,
- Testdatenarten,
- Verantwortliche Personen (Teilprojektleitung),
- Namen der Testpersonen,
- Ergebnisse pro Funktionalität (abgenommen j/n/mit Auflagen),
- Verbleibende Fehler mit vereinbartem Korrekturziel (Termin),
- Gesamtempfehlung.

### **3.4.8 Betriebsbereitschaft und Probetrieb**

Bei den bisherigen Betrachtungen wurde davon ausgegangen, dass das Abnahmeverfahren für ein gegebenes Release ausreicht, um den Regelbetrieb für die Produktion frei zu geben. Bei hochkomplexen Systemen mit vielen Schnittstellen und separat getesteten Einzelmodulen sind die Anforderungen höher. Hier verlangt der Kunde mehr Sicherheit. Um dem entgegen zu kommen, kann eine zusätzliche Betriebsbereitschaftsprüfung mit oder ohne anschließendem Probetrieb durchgeführt werden.

#### **3.4.8.1 Betriebsbereitschaftsprüfung**

Ziel ist es, festzustellen, ob alle durch den Kunden bestellten und vom Lieferanten zu erbringenden Leistungen für den Einsatz des Systems geliefert und einsatz-

bereit sind. Dazu kann mehr gehören als die reine Software. Zusätzlich sind zu berücksichtigen:

- Dienstprogramme, die den laufenden Betrieb unterstützen,
- Berechtigungskonzepte,
- Mandantenfähigkeit,
- separates Standardberichtswesen,
- Dokumentation,
- unter Umständen Betriebskonzepte und Betriebsunterstützung,
- Hotline und sonstige Supportfunktionen.

Da die funktionalen Tests zu diesem Zeitpunkt bereits gelaufen sind, sollte die Betriebsbereitschaftsprüfung solche eigentlich nicht mehr enthalten. Ausreichend ist eine Bestandsaufnahme der gesamten organisatorischen, technischen und betrieblichen Situation zu einem definierten Stichtag. Dazu wird eine entsprechende Dokumentation erstellt (s. Template „Betriebsbereitschaftserklärung“ unter Abschn. 3.5).

### 3.4.8.2 Probetrieb

Auch für den Probetrieb gilt, dass explizite Tests nicht mehr vorgesehen sind, da alle Abnahmen gelaufen sein müssen. Ihm geht auf jeden Fall eine Betriebsbereitschaftsprüfung voraus. Die Art des Probetriebes richtet sich nach der zu Anfang festgelegten Einführungsphilosophie. Er kann also auf einem Parallelsystem stattfinden. Das empfiehlt sich, wenn bereits eine Produktion besteht, die lediglich migriert werden soll. Der Nachteil besteht darin, dass für einen echten Probetrieb dann die gesamte Produktion mit allen betrieblichen Facetten gespiegelt werden müsste unter Hinzuziehung aller fachlichen Ressourcen zur Bedienung, die auch sonst das Tagesgeschäft betreiben. Ein solches Vorgehen stößt sehr schnell an praktische und finanzielle Grenzen.

Eine andere Möglichkeit besteht in der Betrachtung des Produktivsystems (nach allen erfolgten Abnahmen!) unter Last. Der Nachteil ist offensichtlich: er liegt im Risiko der Gesamtinbetriebnahme mit allen Einflüssen auf die Geschäftsprozesse ohne vorherige Probe als solche. In der Praxis würde man zwei Stichtage definieren: Anfang und Ende. Während des laufenden Betriebes würde genau für diesen logisch definierten Zeitraum das Systemverhalten nach Performance und Fehlerhaftigkeit dokumentiert und anschließend bewertet. Ein solcher Probetrieb kann sich über mehrere Monate hinziehen. Währenddessen läuft die Produktion bereits. Gravierende Mängel würden mit hoher Wahrscheinlichkeit schon kurz nach Beginn auftreten. Aus diesem Grunde ist eine wasserdichte fall-back-Strategie mit kurzen Reaktionszeiten unerlässlich.

## 3.5 Templates

Folgende Templates sind nützlich: Zuständigkeitstabelle (Tab. 3.2), Testskripte (Tab. 3.3), Abnahmeprotokoll, Taskliste (Abb. 3.4), Problemspeicher (Abb. 3.5), Ideenspeicher (Abb. 3.6), Bereitstellungsprotokoll (Abb. 3.4), Betriebsbereitschaftsprüfung (Tab. 3.5).

**Tab. 3.2** Zuständigkeitstabelle

Rolle	Name	Org.-Einheit
Leitung	z. B. Vertreter Lieferant	
Leitung	z. B. Vertreter Kunde	
	z. B. QS-Leiter Kunde	
	z. B. QS-Leiter Lieferant	
	z. B. Berater	
	z. B. Controller	

Rolle	Name	Org.-Einheit
Tester	Tester 1	
Tester	Tester 2	
Tester	...	
Tester	Tester N	

**Tab. 3.3** Testskripte

Teilprozess
Subprozess
Arbeitsanweisung
Lfd. Nr.
Variante
Input
Output
Akzeptanzkriterien

- Teilprozess (entspr. Abb. 3.1)
- Subprozess (entspr. Abb. 3.1)
- Arbeitsanweisung: kleinster Prozessschritt, auf der Verrichtungsebene, z. B. Dateneingabe für einen Geschäftsvorgang
- Lfd. Nr.: eindeutige Ident-Nummer für jede Variante der Arbeitsanweisung, gewöhnlich aus der Hierarchie des Prozesses (z. B. 1.2.3.4)
- Variante: Differenzierung der Ausgangssituation für die Arbeitsanweisung, z. B. Kundensegment bei einer Vertragsanlage

- Input: batch- oder online-Eingabe
- Output: zu erwartender Output nach Exekution eines Algorithmus, einer Sortierung oder einer Auflistung etc.
- Akzeptanzkriterien: zusätzliche Kriterien zu „Output“ (z. B. Freishaltung für Zugriffe auf bestimmte Daten, Performance etc.)

## **Abnahme-Protokoll**

Software xyz - Version n.m

Verfasser:

Verteiler:

**Datum der Abnahme:**

## ***Inhaltsverzeichnis***

- 1 Ziel des Dokuments
- 2 Abnahmeverfahren Software xyz
- 3 Teilnehmer der Abnahme
- 4 Gegenstand der Abnahme
- 5 Ergebnis der Abnahme
- 6 Abnahme durch den Auftraggeber

### ***1 Ziel des Dokuments***

Im vorliegenden Dokument sind die Ergebnisse der Abnahme des Systems xyz Release n.m aufgeführt.

### ***2 Abnahmeverfahren Software xyz***

- Das Software-System xyz wurde zum dd.mm.jjjj dem Kunden ABC in der Version n.m zur Abnahme bereitgestellt.
- Der Sinn und Zweck der Abnahme des Systems ist die Überprüfung, ob die vereinbarten und dokumentierten Projektziele und Anforderungen an das entwickelte Software-System erreicht wurden.
- Als Kriterium zur Messung des Zielerreichungsgrades standen zwischen dem Kunden ABC und dem Lieferanten DEF abgestimmte Testszenarien als Basis für die systematische und sorgfältige Durchführung der Tests zur Verfügung.
- Das abgestimmte Abnahmeverfahren ist im Dokument „Abnahmespezifikation Abnahme xyz Version n.m“ vom dd.mm.jjjj beschrieben. Weitergehende Änderungen/Korrekturen wurden in protokollierten Absprachen (vgl. Protokolle) getroffen und jeweils zeitnah umgesetzt.
- Erfüllt das System xyz n.m am Ende der Abnahmefrist die Abnahmekriterien, so wird der Kunde ABC die formale Abnahme des Systems erklären.
- Sollten die Abnahmekriterien nicht zu 100% erfüllt werden, so kann die Abnahme unter Auflagen erklärt werden. Die Auflagen sind dem korrespondierenden Protokoll zu entnehmen.
- Nach der Abnahme wird das Projekt bekannte Fehler und, je nach Priorität, unmittelbar nach Produktivsetzung beheben. Während der Abnahme wurden bereinigte Fehlerkorrekturen (Patches) eingespielt.
- Als Abnahmeumgebung diente ein dezidierter Server mit folgenden Utilities:  
....

- Die Bereitstellung zur Abnahme des Systems xyz\_Rel\_n.m erfolgte durch Installation des Systems xyz in der Abnahmeumgebung durch .... in <Ort>, <Adresse> in den Räumen des Kunden ABC.
- Termine: Beginn der Abnahme: dd.mm.jjjj
- Ende der Abnahme: dd.mm.jjjj

Die Ergebnisse der Abnahme werden unter Punkt 5 (Abnahmeergebnis Rel\_n.m) detailliert dargestellt.

Die durch den Kunden ABC während der Abnahme festgestellten Fehler sind in den Fehlerprotokollen beschrieben und priorisiert. Dieses Protokoll wird frühestens am Ende der Abnahmephase an den Lieferanten DEF übergeben. Es enthält die Erklärung oder Verweigerung der Abnahme, die eventuelle Begründung für eine Verweigerung der Abnahme und das abschließende Fehlerprotokoll.

Die Fehler werden folgendermaßen priorisiert:

- *Priorität 1:* Der Fehler führt zum Produktionsstillstand und kann nicht umgangen werden.
- *Priorität 2:* Der Fehler führt zur Produktionsbehinderung, kann aber umgangen werden.
- *Priorität 3:* Der Fehler führt weder zu Produktionsstillstand noch zu Produktionsbehinderung.

Zum Ende der jeweiligen Abnahmephase wurde die Liste der identifizierten Fehler einer abschließenden Bewertung unterzogen. Dabei wurden die während der Abnahmephase bereinigten Fehler als behoben gekennzeichnet, sofern diese durch die QS des Lieferanten qualitätsgesichert sind.

Die Abnahme des Systems xyz n.m erfolgt, wenn am Ende der Abnahmephase zum dd.mm.jjjj keine Fehler vorliegen, die das Einführungsprojekt wesentlich behindern.

Nach der Abnahme verbleibende Fehler der Priorität 2 oder 3 werden im Rahmen der Produktion nach einem gemeinsam zu erstellenden Zeitplan schnellstmöglich behoben.

Anforderungen, die sich während der Abnahme ergeben haben, werden gemäß dem im Projekt xyz etablierten Anforderungsmanagement/Änderungsverfahren behandelt oder in den täglichen Jour-fix-Protokollen zeitnah abgestimmt und protokolliert

### ***3 Teilnehmer der Abnahme***

s. Template Tab. 3.2 „Zuständigkeitstabelle“.

### ***4 Gegenstand der Abnahme***

Die Auslieferungsbestandteile sind detailliert im Dokument „Abnahmespezifikation Abnahme xyz n.m“ aufgeführt.

## 5 *Ergebnis der Abnahme*

### Anzahl und Priorität der Fehler je Testfall:

Komponente	Testfall	Prio 1	Prio 2	Prio 3	Bemerkung
divers	Trackermeldungen				
	<b>Gesamtsumme der Fehler</b>				

### Liste der vorher bekannten Fehler:

Kurzbeschreibung	Prio	Status	Thema

### Bewertung und Abnahmestatus

- A = ohne Mängel
- B = vorbehaltlich der Beseitigung der Mängel abgenommen (die Beseitigung der Mängel wird nachvollziehbar über den Tracker verwaltet und von der lieferantenseitigen QS qualitätsgesichert).
- C = nicht abgenommen

Abnahmegegenstand	Abnahmestatus
Software-System xyz n.m	

## 6 *Abnahme durch den Auftraggeber*

Die Abnahme der oben aufgeführten Abnahmegegenstände erfolgt nach eingehender Untersuchung der vorgelegten Unterlagen und Ergebnisse.

Die Ergebnisse und die Bewertungen der Abnahme sind im Abschn. „5 Ergebnis der Abnahme“ – auch mit Einschränkungen – aufgeführt und im Tracker festgehalten.

Datum                      Leiter der Abnahme

**Abb. 3.4** Taskliste

Task	Ausführender	Zieldatum	Status
Software abnehmen	Wagner	31.10.2004	erledigt
Daten prüfen	Junker	30.11.2005	In Arbeit
....			

lfd. Nr.	Tester	Testfall-Nr.	Problembeschreibung	Prio	Datum	Status

**Abb. 3.5** Problemspeicher

lfd.Nr.	Tester	Arbeitsanweisung/ Teilprozess	Change Request	Datum

**Abb. 3.6** Ideenspeicher

**Tab. 3.4** Bereitstellungsprotokoll

Systemkomponente	Zustand
Serverlandschaft	
Peripherie	
Laufwerke	
Datenbank	
Systemprogramme	
Kommunikationsprogramme	
Abzunehmende Software	
Sonstige Dienstprogramme	
Customizing	
Logfiles	
Abnahme-Verantwortlicher	
-	

**Tab. 3.5** Betriebsbereitschaftsprüfung

Lfd. Nr.	Gegenstand	Status/Kommentar
	Systemkomponenten	
	Serverlandschaft	
	Peripherie	
	Software	
	Systemsoftware	
	Customizing	
	Dienstprogramme	
	Datenbank	
	Urladungen/Migrationen	
	Zyklische Schnittstellen	
	Standard-Berichte	
	Durchgeführte Schulungen	
	Dokumentation	

### 3.6 Lösungsweg

Vereinbarung der Ziele zwischen Lieferanten und Kunde > Identifikation spezifischer Testanforderungen > Zuordnung von Testverantwortlichkeiten > Einforderung von Testskripten und Testdaten > Planung, Durchführung, Bewertung > Bereitstellung der Fehlerdokumentation > Steuerung des Reviewprozesses > Festlegung des Patchzyklus > eventuell Probebetrieb.

# Kapitel 4

## Teilprojektorganisation Qualitätsmanagement

### 4.1 Projektumfeld

Neben der Festlegung von Aufgaben und der allgemeinen Organisation ist es unerlässlich, das gesamte Projektumfeld instrumentell so auszustatten, dass Zuständigkeiten transparent sind und Reibungsverluste auf allen Arbeitsebenen minimal gehalten werden. Die wichtigsten Pfeiler einer effizienten Organisation auf der Durchführungsebene sind die Kommunikationswege und die Dokumentation des Projektgeschehens. Zur Kommunikation gehören auch die Eskalationspfade. Die Dokumentation teilt sich in die Meta-Ebene der Projektfortschrittsdokumentation und in die eigentliche Dokumentation von Arbeitsinhalten.

Um Kommunikation und Dokumentation zusammen mit den zugehörigen Verantwortlichkeiten durchhalten zu können, sind entsprechende organisatorische und technische Strukturen zu schaffen – und zwar vor Projektstart. Insbesondere bei den technischen Voraussetzungen bieten sich heute vielfältige Möglichkeiten aus dem Fundus der Office- und Webtechnologie an. Damit einher gehen auch entsprechende Freiheitsgrade, die durch die nachfolgende Auswahl keinesfalls eingeschränkt werden sollen. Letztendlich sind Art und Ausprägung der zu nutzenden Instrumente sekundär im Verhältnis zur Zielerreichung der Aufgabenstellung.

### 4.2 Kommunikation und Dokumentation

#### 4.2.1 *Regelkommunikation*

Es gibt mehrere Regelkommunikationsprozesse, die für ein Abnahmeprojekt relevant sind. Grundsätzlich unterscheidet man die Kommunikation nach innen und nach außen. Intern sind damit die Kommunikationswege innerhalb des Teilprojektes „Qualitätsmanagement“ gemeint. Die Außenkommunikation teilt sich in die Berichterstattung an das Gesamtprojektmanagement und in diejenige, die die Linie zum Adressaten hat.

### 4.2.1.1 Kommunikation Teilprojekt intern

Im Vorlauf zu den Abnahmen gibt es das so genannte Kick-off. Die Teilprojektleitung ist über den Funktionsinhalt des Abnahmegegenstandes sowie die Ecktermine informiert. Aus den Abnahmeinhalten lassen sich die erforderlichen fachlichen Kompetenzen der Tester erschließen. Häufig handelt es sich um Personen, die auch an der Erstellung der Lastenhefte mitgewirkt haben. Die Projektleitung wendet sich an die zuständigen Fachbereichsleiter mit der Bitte um Freigabe für die identifizierten Ressourcen bzw. um Benennung derselben. Damit stehen fachbereichsseitig die Teilnehmer des Kick-offs fest.

Im Vorfeld zum Kick-off sollte zwischen der Abnahmeleitung und dem Pendant auf der Lieferantenseite ein Planungsgespräch stattfinden, indem die Kommunikationsregeln für das Kick-off festgelegt werden. Insbesondere muss geklärt sein, wo eventuell kritische Schwerpunkte zu erwarten sind. Das hängt ganz von den Funktionalitäten ab: Schnittstellen, Datenmodelländerungen, Verkettung von Batch-Läufen usw. Nach Klärung dieser Randbedingungen kann das Kick-off stattfinden – in der Regel einige Wochen vor Abnahmebeginn.

Zugegen sein sollten alle Testverantwortlichen, die Teilprojektleitung, Ansprechpartner des Lieferanten während der Abnahme, sonstige Spezialisten und – wenn möglich – ein Vertreter des Anforderungsmanagements sowie Vertreter der Auftraggeber entweder aus dem Fachbereich oder der IT-Linienorganisation. Im Kick-off müssen folgende Fragen geklärt werden:

- Zuordnung der Testverantwortlichen zu den Einzelfunktionalitäten,
- erforderliche Testdaten (Abzüge aus Produktivsystemen oder synthetische Daten),
- Schnittstellenläufe,
- die zu testenden Geschäftsprozesse – Testskripte (zur Erstellung oder Verteilung),
- User Berechtigungen,
- Testreihenfolgeplanung,
- Patch-Zyklus,
- Review-Zyklus.

Die Ergebnisse sind zu protokollieren und an die Teilnehmer bzw. die Hierarchie zu kommunizieren. Möglicherweise gibt es vor der Abnahme noch eine weitere gemeinsame Zusammenkunft oder Arbeitsgruppen, die bestimmte Details, wie z. B. die Testskripte, erarbeiten.

Während der Abnahme finden regelmäßige Reviews statt. Wie bereits an anderer Stelle erwähnt, werden dort

- Fehler priorisiert,
- Patch-Auslieferungen terminiert,
- der zeitliche Testplan angepasst,
- Klärungsfälle abgearbeitet,
- neue Change Requests formuliert.

Das letzte Review vor Ablauf der Abnahmephase dient der Entscheidungsfindung, ob und unter welchen Bedingungen die Abnahme empfohlen werden kann.

### 4.2.1.2 Regelkommunikation nach außen

Mit der Gesamtprojektleitung

Die Gesamtprojektleitung, unter der das Qualitätsmanagement lediglich als Teilprojekt fungiert, konsolidiert normalerweise alle Teilprojektberichte und erwartet deshalb einen Meilenstein bezogenen Status in fester Frequenz: wöchentlich oder vierzehntägig. Die Details aus dem Vorfeld zum Kick-off und aus den Reviews sind dafür uninteressant. Erwartet werden:

- Abarbeitungsgrad,
- Gefährdungen des Projektzieles,
- Einhaltung des Kostenrahmens,
- Entscheidungsbedarf.

Mit der Linie

Die Liniverantwortlichen erwarten Sicherheit gegenüber der Inbetriebnahme der angekündigten Funktionalitäten, was Zeitpunkt und Qualität betrifft. Meistens wird die Abnahmeleitung zu einer der regelmäßigen Besprechungen eingeladen und kann zu einem eigenen Tagesordnungspunkt dazu berichten. Als Berichtsformat kann das gleiche dienen wie unter „Mit der Gesamtprojektleitung“.

Eskalationspfade

Da es unterschiedliche Probleme geben kann, sind dafür auch unterschiedliche Eskalationspfade denkbar:

- Ressourcenengpässe bei Testern (Fachbereichsleitung),
- Systemverfügbarkeit und Support (Lieferantenhierarchie),
- terminliche Gefährdung (Gesamtprojektleitung),
- andere.

Aus Effizienzgründen ist es jedoch sinnvoll, für alle Probleme einen einzigen Eskalationspfad zu definieren – und zwar denjenigen über die Gesamtprojektleitung. Letztendlich spitzen sich alle Pfade auf die Unternehmensleitung zu, an die Linie und Projekt beide berichten.

## 4.2.2 Dokumentation

Auf die Dokumentation von Testfällen wurde bereits weiter oben hingewiesen.

An dieser Stelle soll die Dokumentation der Projektvorbereitung und des Projektfortschritts behandelt werden. Alle relevanten Dokumente sollen online in einem eigenen Verzeichnis verfügbar gehalten werden, wobei Lese- und Schreibberechtigungen abhängig von der Funktion im Projekt zugeteilt werden.

Es bietet sich an, für den eigentlichen Abnahmeprozess eine eigene Abnahmerichtlinie anzulegen (s. Kap. 5 „Abnahmerichtlinie“). Diese Richtlinie enthält alle Vereinbarungen von der Anforderung bis zur Inbetriebnahme. Darüber hinaus sollte eine Dokumentenstruktur geschaffen werden, die den Projektfortschritt belegt. Eine beispielhafte Struktur ist in Abb. 4.1 gegeben.

Daraus gehen folgende Schwerpunkte hervor, die je nach Ausprägung des Projektes auf jeden Fall dokumentiert werden sollten:

- Abnahmerichtlinie,
- sonstige Richtlinien
- Planung,
- Lastenhefte,
- Pflichtenhefte,
- Testskripte,
- Systemumgebung,
- Budget,
- Protokolle,
- Statusberichte,
- Abnahmeergebnis.

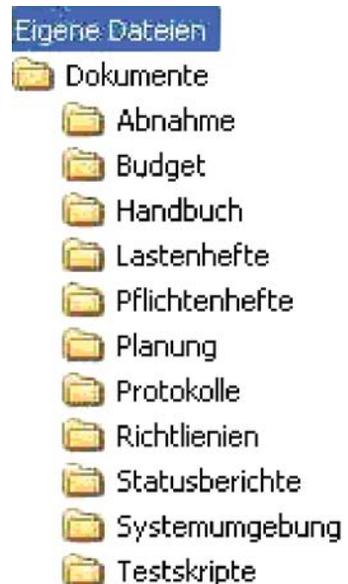
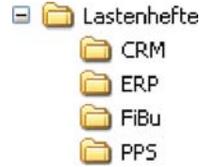


Abb. 4.1 Dokumentenstruktur

Abb. 4.2 Unterstrukturen



Die Hauptordner sind dann wiederum unterteilt in Unterstrukturen, wie Abb. 4.2 beispielhaft für Anwendungen zeigt.

### 4.2.2.1 Pflichtenheftphase

Eine Sonderbetrachtung verdient die Pflichtenheftphase. In bestimmten Umfeldern werden Pflichtenhefte auch als Business Blueprints bezeichnet, obwohl die Bedeutung nicht ganz identisch ist. Auf diese Unterschiede soll hier nicht eingegangen werden. Wie bereits erwähnt, bilden die Pflichtenhefte die Grundlage für Testskripte und dienen somit als Messlatte für die Abnahmeinhalte (Muster s. Abb. 4.3 bis 4.7).

In vielen Fällen werden für ein Update (Release, Service Pack) mehrere Pflichtenhefte benötigt: je eines für eine bestimmte funktionale Änderung oder

<p><b>Pflichtenheft</b> -Muster-</p> <p><b>Verfasser:</b> <i>Autor oder Team</i></p> <p><b>Gegenstand:</b> <i>Anforderungstitel</i></p> <p><b>Version:</b> <i>x.y</i> <i>&lt;x: beginnend bei 0; Erhöhung um 1 jeweils bei Statuswechsel;</i> <i>y: beginnend bei 1; Fortschreibung innerhalb des gleichen Status bei Überarbeitungen&gt;</i></p> <p><b>Status:</b> <i>Entwurf</i> <i>in Bearbeitung</i> <i>in Prüfung</i> <i>freigegeben</i></p>	<p><b>Inhaltsverzeichnis</b></p> <ol style="list-style-type: none"> <li>1. Allgemeines             <ol style="list-style-type: none"> <li>1.1 Zielsetzung</li> </ol> </li> <li>2. Systemüberblick             <ol style="list-style-type: none"> <li>2.1 Kurzbeschreibung</li> <li>2.2 Komponentenarchitektur</li> <li>2.3 Objektmodell</li> <li>2.4 Datenmodell</li> <li>2.5 Schnittstellen</li> </ol> </li> <li>3. Prozesssteuerung</li> <li>4. Berechtigungskonzept</li> <li>5. IT-Sicherheit</li> <li>6. Qualitätsanforderungen             <ol style="list-style-type: none"> <li>6.1 Betriebsumgebung</li> <li>6.2 Entwicklungsumgebung</li> </ol> </li> <li>7. Referenzierte Dokumente</li> <li>8. Versionshistorie</li> </ol>
---	---

Abb. 4.3 Pflichtenheft 1

<p><b>1. Allgemeines</b></p> <p>1.1 Zielsetzung</p> <p><i>Gegenstand des Pflichtenheftes ist das WIE der Anforderung. Die detaillierte Systembeschreibung ist Grundlage für die Realisierung durch die Entwicklungsabteilung des Lieferanten.</i></p> <p><i>Die inhaltlichen Forderungen des Lastenheftes werden auf diese Weise durch die technischen Lösungswege und deren Voraussetzungen übersetzt</i></p>	<p><b>2. Systemüberblick</b></p> <p>2.1 Kurzbeschreibung</p> <p><i>Einordnung des Systemteils innerhalb der Anwendung bezogen auf die hier zu behandelnde Teilfunktion.</i></p> <p>2.2 Komponentenarchitektur</p> <p><i>Übernahme bestehender Strukturen bzw. Neugestaltung mit der neuen Funktion.</i></p> <p><i>Zusammenwirken der relevanten Systemkomponenten und deren Abhängigkeit untereinander.</i></p>
--	---

Abb. 4.4 Pflichtenheft 2

<p>2.3 Objektmodell</p> <p><i>Aufbauend auf die Komponentenarchitektur werden die Objektmodelle der individuellen Komponenten entwickelt, ergänzt oder belassen und untereinander im Sinne der neuen Architektur miteinander in Beziehung gebracht.</i></p> <p>2.4 Datenmodell</p> <p><i>Umsetzung des Objektmodells auf eine physische Datenstruktur mit Datenbankschema, -tabellen und Attributen als Basis für die direkte Implementierung der angepassten Datenbank.</i></p>	<p>2.5 Schnittstellen</p> <p><i>Die Schnittstellenbeschreibung enthält Informationen über auszutauschende Datenfelder und – formate, Plausibilitätsprüfungen, Mappingtabellen, Zugriffmodalitäten usw.</i></p> <p><i>Sollten Schnittstellenbeschreibungen in gesonderten Dokumenten vorliegen, können sie an dieser Stelle referenziert werden.</i></p>
--	---

Abb. 4.5 Pflichtenheft 3

<p><b>3. Prozesssteuerung</b></p> <p><i>Entscheidung über den Anstoß der Anwendung (synchron / asynchron), Einbettung in einen Workflow, Sequenzierung von Eingabemasken usw.</i></p> <p><b>4. Berechtigungskonzept</b></p> <p><i>Die Forderungen aus dem Lastenheft sind abzubilden auf Zugriffswege und Attribute, in der Regel über Profile, die über Tabellen zueinander in Beziehung gebracht werden.</i></p>	<p><b>5. IT-Sicherheit</b></p> <p><i>Identifizierung geeigneter Tools, Systemkomponenten und physikalischer Voraussetzungen, die die im Lastenheft geforderten Standards sicherstellen.</i></p> <p><b>6. Qualitätsanforderungen</b></p> <p>6.1 Betriebsumgebung</p> <p><i>Beschreibung der umgebenden Konfiguration, die zum betreiben der Software erforderlich ist</i></p> <p>6.2 Entwicklungsumgebung</p> <p><i>Technische Entwicklungsstandards, Werkzeuge, Schnittstellensimulation, Testdaten, Freigabeverfahren, Konfigurationsparameter</i></p>
--	---

**Abb. 4.6** Pflichtenheft 4

<p><b>7. Referenzierte Dokumente</b></p> <p><i>Alle Dokumente, die zum Verständnis und zur Herleitung der neuen Realisierungsvorgaben von Belang sind:</i></p> <ul style="list-style-type: none"> <li>- Lastenhefte</li> <li>- vorangegangenen Pflichtenhefte, die thematisch mit dem vorliegenden zusammenhängen</li> <li>- Prozessdokumentationen</li> <li>- Objektmodelle</li> <li>- Besprechungsprotokolle</li> <li>- Fehlerprotokolle</li> </ul>	<p><b>8. Versionshistorie</b></p> <p><i>Es bietet sich folgendes Template an:</i></p> <table border="1" style="width: 100%; border-collapse: collapse; text-align: center;"> <thead> <tr style="background-color: #e0f2f1;"> <th>Version</th> <th>Status</th> <th>Datum</th> <th>Verantwortlich</th> <th>Bemerkungen</th> </tr> </thead> <tbody> <tr> <td> </td> <td> </td> <td> </td> <td> </td> <td> </td> </tr> </tbody> </table>	Version	Status	Datum	Verantwortlich	Bemerkungen					
Version	Status	Datum	Verantwortlich	Bemerkungen							

**Abb. 4.7** Pflichtenheft 5

Ergänzung. Diese Dokumente werden teilweise von ganz verschiedenen Personen freigegeben – je nach erforderlicher Fachkompetenz. Zusätzlich werden die Unterlagen nicht alle gleichzeitig in Umlauf gebracht, sondern nach und nach. Das bedeutet, dass das Teilprojekt ein Tracking einrichten muss. Diese Tracking-Instanz kommuniziert zwischen den Pflichtenheft-Erstellern auf Lieferantenseite und den Freigabern auf der Kundenseite. Ziel ist es, die Freigaben so zeitig zu erwirken, dass ausreichende Realisierungszeit bleibt, um den Abnahmetermin zu halten. In dieser Phase kommt es häufig auch noch zu Änderungswünschen seitens der Fachbereiche, wenn diese zum ersten Mal ihre Vorgaben schriftlich fixiert sehen. Auch um die Erledigung dieser Anforderungsanpassungen sollte sich die Tracking-Instanz kümmern.

### 4.3 Strukturen

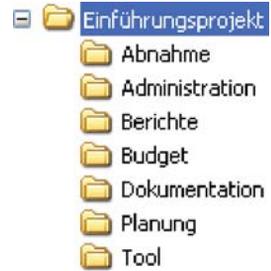
Auch das Teilprojekt Qualitätsmanagement muss bei entsprechender Größe noch differenziert ausgestaltet werden. Inhalte und Verantwortlichkeiten müssen sauber definiert und unter Umständen verschiedenen Funktionsträgern zugeordnet werden. Zu beachten sind die Blöcke:

- Allgemeine Administration,
- Projektmanagement: Tool,
- Dokumentation,
- Planung und Steuerung,
- Berichtswesen,
- Budgetverwaltung,
- Abnahmeprozedere,
- Abnahmeergebnisse.

Man benötigt nicht für jeden Funktionsblock eine eigene Zuständigkeit. Häufig lassen sich bestimmte Aufgaben von ein und derselben Ressource bearbeiten, zum Beispiel Administration und Nutzung des Projektmanagement-Tools. Wichtig ist, dass Verantwortlichkeiten festgelegt und Arbeitsinhalte nicht vergessen werden. In regelmäßigen Teambesprechungen (nicht in den Reviews!) wird der Erledigungsstand ermittelt und dient als Grundlage für das Berichtswesen an die übergeordneten Instanzen.

### 4.4 Abbildung der Projektstrukturen

Die detaillierte Struktur von Verantwortlichkeiten und zugehörigen Inhalten lässt sich perfekt als Ordnerstruktur auf einem eigens dafür vorgesehenen logischen Laufwerk abbilden. Eine beispielhafte Struktur ist in Abb. 4.8 zu sehen.

**Abb. 4.8** Statusbericht

### 4.4.1 Budgetplanung

Ohne Budget (in Aussicht) kann kein Projekt beginnen. Der oberste Controllinggegenstand ist das Gesamtbudget, hinter dem eine einzige Zahl steht, die am Ende des Projektes nicht überschritten werden darf. Meistens wird dieses Budget aber unterteilt in einen Investitions- (Hardware)Part und in einen Personalkostenplan. Der Personalkostenplan läuft über die Zeitachse des Projekts und bricht die pro Projektressource erforderlichen bzw. genehmigten Projektstage pro Zeiteinheit (meistens pro Monat) herunter. Dem individuellen Tageskontingent sind entsprechende Tagessätze in Geldwert hinterlegt. Auf diese Weise lassen sich Engpässe und aus den Rückmeldungen über Zeitrückstände nicht verbrauchte Projektstage ermitteln. Eine regelmäßige Neuplanung während des Projektfortschritts ist meistens unerlässlich.

## 4.5 Templates

Folgende Templates sind nützlich: Agenda Kick-off (Abb. 4.9), Ablaufsplan (Abb. 4.10), Patch-Zyklus (Abb. 4.11), Statusbericht (Abb. 4.12)

### Agenda Abnahme-Kick-off

- Einführung
- Vorstellung des Zeitplans
- Erläuterung Testvorgehen
- Zuordnung Tester – Testfälle
- Definition Testdaten
- Technische Voraussetzungen
- Reviewprozess
- Patch-Zyklus
- Berechtigungsliste

**Abb. 4.9** Agenda Kick-off

Schritt-Nr.	Datum	Uhrzeit	batch Bezeichnung	Ergebnis	Bewertung

Abb. 4.10 Ablaufplan

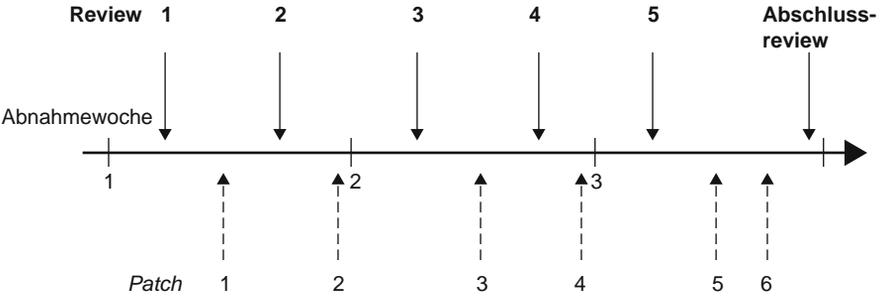


Abb. 4.11 Patch-Zyklus

**Projektstatus:** *Projektbezeichnung*

**Berichtsdatum:**

Meilenstein	Erfüllungsgrad	Probleme	Maßnahmen
Bezeichnung Aktivität1 Aktivität2			
Bezeichnung Aktivität1 Aktivität2 ...			
Bezeichnung Aktivität1 Aktivität2 ...			
Bezeichnung Aktivität1 Aktivität2 ...			

Abb. 4.12 Statusbericht

## **4.6 Lösungsweg**

Detaillierung des Teilprojektumfeldes > Einrichten von Regelkommunikation und Berichtswesen sowie von Eskalationspfaden > Festlegen einer Dokumentenstruktur > Abbildung von Teilverantwortlichkeiten innerhalb der Teilprojektstruktur > Budgetplanung und Verfolgung.

# Kapitel 5

## Abnahme-Richtlinie

### 5.1 Sinnfälligkeit

Wie aus den bisherigen Ausführungen hervorging, sind unterschiedlichste Parteien an Test und Abnahme beteiligt. Die wichtigsten Partner sind:

- Lieferant (und dessen Entwicklungsabteilung sowie seine interne Qualitätssicherung),
- Kunde,
- Projektleitung.

Beim Kunden treten in Erscheinung:

- die Fachbereiche und
- die IT-Organisation.

Aus der Projektorganisation ist das IT-Qualitätsmanagement das wichtigste Teilprojekt, aber es spielen mit:

- Anforderungsmanagement,
- Change-Management,
- Sicherheit und andere.

Es hat sich deshalb bewährt, die unterschiedlichen Interessen und Verantwortlichkeiten bei einer Software-Abnahme nicht nur vertraglich zu regeln, sondern praxisorientiert auch in einem gemeinsamen Dokument, einer Abnahmerichtlinie, zu bündeln. Diese Richtlinie ist einer spezifischen Abnahmesituation angepasst und kann im Konfliktfall immer wieder referenziert werden. Der wichtigste Vorteil aber liegt darin, dass bei ihrer Erstellung alle Beteiligten von vorn herein Klarheit über ihre Rolle gewinnen. Diese Vorklärung und die anschließende Fixierung als Richtlinie ersparen nachher viele Reibungsverluste und Klärungsgespräche. Die Richtlinie kann später als Vorlage für weitere Abnahmen wieder verwendet und entsprechend angepasst werden.

## 5.2 Verhältnis zwischen den Vertragspartnern

Wie bereits angedeutet, vertreten die unterschiedlichen Partner je ganz eigene Interessenlagen. Dem Lieferanten geht es darum, möglichst termingerecht seine Bestellung abzuliefern und dafür eine Freigabe zu erhalten mit dem Ziel einer zügigen Vergütung und einem Minimum an Gewährleistungsfällen. Der Fachbereich möchte natürlich die von ihm bestellte Funktionalität erhalten, wobei oftmals die Erwartungshaltung zwischen Lieferant und Fachbereich divergieren kann. Erst nach Realisierung erkennt in manchen Fällen der Enduser, was er wirklich wollte, was aber aus Sicht des Lieferanten anders spezifiziert worden war. Diese Konfliktfälle gilt es während der Abnahme zu klären. Neben dem technisch-fachlichen Know-how erweist sich hier die Kernkompetenz der Projektleitung als maßgeblicher Faktor.

Die Interessenslage der Kunden-IT-Organisation ist geprägt von Folgeaktivitäten nach der Abnahme wie z. B. der operativen Einführung und der Betriebsunterstützung durch ein Service Center. Es ist deshalb darauf zu achten, dass zwischen Fachbereich und IT Einvernehmlichkeit und gleiches Verständnis im Sinne einer Service Mentalität herrscht. Ähnliches gilt für die Teilprojekte Anforderungsmanagement, Change Management, Sicherheit und andere, die häufig stärker durch administrativ-restriktives Verhalten aus der Natur ihrer Arbeit heraus geprägt sind.

Allen Beteiligten muss klar sein, dass sie auf ein gemeinsames Ziel zusteuern. Letztendlich ist das Interesse aller das Gleiche: gegenseitige Zufriedenheit. Aus diesem Grunde wird es unausweichlich sein, dass während einer Abnahme Kompromisse und Entgegenkommen von allen Seiten erforderlich sind. Hier zeigen sich dann auch die Grenzen jeglicher formalen Vereinbarung.

## 5.3 Konsensfindung

Die Richtlinie sollte von der Abnahmekoordination, sprich: vom kundenseitigen Qualitätsmanagement, entworfen werden. Wie unten zu sehen ist, werden dort alle grundsätzlichen Aspekte, aber auch schon viele operative Details festgelegt. Anschließend unterliegt das Dokument einem Freigabeprozess, dessen Historie festgehalten wird. An der Freigabe sollte mindestens die Vertretung des Lieferanten beteiligt sein – unter Umständen auch Fachbereichsleiter. Sie dient der Gesamtprojektleitung als Referenz, ist aber Arbeitsgrundlage für die Teilprojektleitung.

Der Erstellungsprozess ist also ein iterativer Prozess. Das Ergebnis kann zu den Vertragsunterlagen genommen werden.

## **5.4 Die Richtlinie (Beispiel)**

### **5.4.1 Einführung**

#### **5.4.1.1 Aufgabe dieses Dokumentes**

Abnahmen zum System xyz sind generell durch das Dokument „Abnahmerichtlinie, Grundsätze für Organisation und Ablauf“, beschrieben.

Das vorliegende Dokument stellt die Abnahmespezifikation für die Abnahmephase „Release n.1“ durch den Auftraggeber dar und betrifft den vorher freigegebenen Leistungsumfang in den entsprechenden Pflichtenheften. Diese Pflichtenhefte wurden auf Basis der relevanten internen Geschäftsprozesse erstellt und sind zusätzlich im „Testfallkatalog zur Abnahme des Systems xyz Release n.1“ dokumentiert.

Grundlage der Abnahme sind die Pflichtenhefte. Weitere Arbeitsdokumente, die zur Festschreibung dieser Spezifikation herangezogen wurden, sind weiter unten referenziert. Die Rangfolge ist absteigend gemäß der Auflistung.

Für die Abnahme gelten außerdem die QS-internen Prozesse des Herstellers, zusammengefasst in dem Dokument Abnahmeverfahren xyz\_Rel\_n.1.

#### **5.4.1.2 Geltungsbereich**

Das Dokument ist gültig für die Abnahme der Ergebnisse des zugehörigen gleichnamigen Projektes xyz durch den Auftraggeber bzgl. des Abnahmeobjektes „xyz Version n.1“.

Es wird im Vorfeld der Abnahme zwischen dem Auftraggeber und dem Systemhersteller abgestimmt und ist fortan in dieser aktuellen Version gültig.

Einzelne Aspekte dieser Spezifikation können aus organisatorischen oder verfahrenstechnischen Gründen bis zur oder während der Abnahme Änderungen unterworfen sein. Solange diese in einvernehmlich akzeptierten Protokollen dokumentiert sind, gelten sie als legitime Bestandteile der Abnahmespezifikation.

### **5.4.2 Begriffsdefinitionen**

#### **5.4.2.1 Abnahme**

Die Abnahme beim Hersteller schließt sich an die Phasen der Umsetzung kundendefinierter Testfälle in Testprozessschritte sowie funktionaler Nachbesserungen an, welche

auf die System-Entwicklung folgen. Sie wird abgeschlossen durch die QS-Freigabe zur Auslieferung. Ihr Ziel ist der Nachweis der allgemeinen Funktionstüchtigkeit als Voraussetzung für den Betrieb (als Voraussetzung für ein Einführungsprojekt).

Schwerpunkt der Abnahme ist die Prüfung der funktionalen Vollständigkeit in Bezug auf die oben genannten freigegebenen Pflichtenhefte unter Hinzuziehung der zugrunde liegenden Geschäftsprozesse.

Basis und Verfahren der Testfallumsetzung auf Testprozessschritte sowie deren Planung werden in einem gesonderten Kapitel behandelt.

Die gesamte Abnahme zieht sich über einen Zeitraum von 10 Arbeitstagen hin. Am Ende eines jeden Abnahmetages erfolgt ein Review durch Mitglieder der Steuerungsgruppe mit einer Fehlerbewertung (Kriterien s. u.).

Die Nachbesserung kritischer Fehler erfolgt zweimalig nach jedem Mittwoch durch das Entwicklungsteam. Die dabei produzierten Patches werden donnerstags eingespielt und freigegeben. Daraufhin erfolgt der wiederholte Test der fehlerhaften Funktionen. Als einziges Dokumentationstool neben den Protokollblättern für die Testpersonen dient der beim Hersteller eingesetzte Fehler-Tracker.

Die weitere Nachbesserung der unkritischen Fehler/Mängel erfolgt im Rahmen des vorhandenen Fehler- und Konfigurationsmanagements.

Die Abnahme des Systems xyz n.1 erfolgt, wenn am Ende der Abnahmephase zum 30.08.2008 keine Fehler vorliegen, die das anschließende Einführungsprojekt xyz n.1 wesentlich behindern.

### **5.4.3 Mitwirkungsleistungen**

Mitwirkungsleistungen werden von Hersteller- und Kundenseite gemäß der Aufgabenverteilung weiter unten erbracht. Hierzu werden eine Steuerungsgruppe und ein Abnahmeteam installiert, die einen Teil der Aufgaben des Herstellers und des Kunden übernehmen.

Der Hersteller hat im Rahmen der Abnahme für folgende Mitwirkungsleistungen Sorge zu tragen:

- Herstellung der Abnahmebereitschaft
- Bereitstellung des einsatzfähigen Testsystems (einschließlich installierter Software)
- Bereitstellung aller abzunehmender Systemkomponenten
- Auslieferung der vollständigen Software und Dokumentation auf geeignetem Medium
- Umsetzung von Customizing-Vorgaben durch den Kunden
- Bereitstellung von Testdaten, die gemeinsam mit dem Kunden definiert werden
- gegebenenfalls Bereitstellung von zusätzlichem Support bzgl. Rechenzentrumsbetrieb und Service Center
- Bereitstellung von weiteren Abnahmeressourcen, z. B. in Bezug auf das Arbeitsumfeld des Abnahmeteams
- Bereitstellung von Testprozessschritten, die gemeinsam mit dem Kunden aus den relevanten Prozess basierenden Testfällen entwickelt werden

- Erstellung eines detaillierten Abnahmetestplans
- gemeinsame Leitung des Abnahmeteams zusammen mit dem Kunden
- Mitarbeiter als Teilnehmer für die Steuerungsgruppe stellen
- Schulung des Abnahmeteams für die Durchführung der Abnahme und die Bedienung der Systeme
- Fehlermanagement betreiben und korrigierte Software innerhalb des vereinbarten Zeitraums liefern

Der Kunde hat im Rahmen der Abnahme für folgende Mitwirkungsleistungen Sorge zu tragen:

- Mitarbeiter als Teilnehmer der Steuerungsgruppe stellen
- Mitarbeiter als Teilnehmer des Abnahmetestteams stellen
- Testfälle mit Akzeptanzkriterien erstellen
- Vorgaben für das Customizing an den Hersteller übergeben
- gemeinsame Leitung der Steuerungsgruppe mit dem Hersteller
- gemeinsame Leitung des Abnahmeteams mit dem Hersteller
- Erarbeitung von Testdatenanforderungen mit dem Hersteller
- Umsetzung der Testfälle in Testprozessschritte mit dem Hersteller

Die Steuerungsgruppe übernimmt von den vorstehenden Aufgaben folgenden Umfang:

- Installation des Abnahmeteams,
- sonstige Ressourcenplanung,
- Überwachung der Vorbereitung der Abnahme gemäß Abnahmespezifikation und Abnahmetestplan sowie Entwicklung und, Verfolgung von geeigneten Maßnahmen bei Abweichung anhand einer Taskliste,
- Bewertung der Abnahmeergebnisse durchführen und durch Unterschrift der gemeinsamen Teilprojektleitung auf dem Abnahmeprotokoll bestätigen.
- strittige Bewertungen von Abnahmeergebnissen klären und ggf. strittige Punkte an die Projektleitung eskalieren.
- Maßnahmen zur Fehlerbehebung in die Wege leiten und deren Bearbeitung verfolgen

Das Abnahmeteam übernimmt von den vorstehenden Aufgaben folgenden Umfang:

- Teilnahme an den vorbereitenden Schulungen
- durchführen der Abnahmetests mit Erstellung der Testprotokolle und Eingabe der Fehlermeldungen in das Fehlertool
- tägliches Review der Abnahmetests mit Empfehlungen für Folgemaßnahmen
- Testberichte an die Steuerungsgruppe leiten

#### **5.4.4 Abnahmebereitschaft und Zeitplan**

Zu Beginn der Abnahme wird das Testsystem bereitgestellt, und die abzunehmenden Komponenten durch ein Bereitstellungsprotokoll dokumentiert (s. Anhang). Das

Abnahmeteam überprüft es auf seine Abnahmebereitschaft anhand der Anforderungen aus dem Abschnitt über Referenzdokumente.

Fehlende Voraussetzungen werden im Prüfprotokoll protokolliert und in Bezug auf eine etwaige Beeinträchtigung der Abnahmebereitschaft überprüft. Sind die Bedingungen erfüllt, erklärt das Abnahmeteam die Abnahmebereitschaft und protokolliert dies entsprechend unter Beifügung des Bereitstellungsprotokolls. Das Abnahmeteam entscheidet über eventuell notwendige Maßnahmen zur Herstellung der Abnahmebereitschaft.

Die für die Abnahme maßgeblichen Termine sind wie folgt:

30.06.2008	Auslieferung Software xyz n.1
01.07.2008	Bereitstellung Akzeptanzkriterien
12.08.2008	Bereitstellung Schulungssystem
16.08.2008	Bereitstellung der Testfälle und Testdaten zur Abnahme
19.08.2008	Bereitstellung der Software zur Abnahme
30.08.2008	Ende der Abnahme

#### 5.4.4.1 Teilabnahmen

Bei Abnahmen von Komponenten des Systems xyz kommen unterschiedliche Schwerpunkte zur Geltung, die durch so genannte Teilabnahmen abgedeckt werden. Speziell bei der vorliegenden Abnahme einer kompletten Version des xyz n.1, werden folgende Teilabnahmen durchgeführt:

- Anwender-Funktionen
- Berechtigungen
- Schnittstellen-Funktionen im simulierten Schnittstellenumfeld

### 5.4.5 *Aufbau der Tests*

#### 5.4.5.1 Grundlagen

Das gesamte TestszENARIO basiert auf zwei Säulen:

- die Zerlegung des Gesamtprozesses xyz in eindeutig definierte und identifizierbare Testfälle,
- die Umsetzung dieser Testfälle in funktionale, systemkonforme Testprozessschritte innerhalb einer definierten Konfiguration mit dazugehörigen Testdaten.

### 5.4.5.2 Zerlegung des Geschäftsprozesses xyz

Die Zerlegung des Gesamtprozesses xyz folgt der Hierarchie:

Hauptprozess

- Teilprozess,
- Subprozess,
- Arbeitsanweisung.

Die Ebene Arbeitsanweisung lässt ihrerseits Varianten über die Variablen „Ausgangssituation“ und „Input“ zu. Die Kombination dieser drei Felder wird über eine laufende Nummer, die der Prozesshierarchie folgt, eindeutig identifizierbar.

### 5.4.5.3 Akzeptanzkriterien

Für jeden identifizierten Testfall existieren Akzeptanzkriterien, die ihrerseits zerlegt sind in:

- erwarteter Output und
- Ziel (bezogen auf die Arbeitsanweisung).

### 5.4.5.4 Referenz

Sämtliche Testfälle mit ihren Akzeptanzkriterien sind in einem Testfallkatalog festgehalten. Hierin befinden sich auch pro Testfall Bewertungsmatrizen, die folgende Informationen festhalten:

- Relevanz,
- Testbereich,
- Testergebnis,
- Kommentar,
- Tester und
- Tracker-Nr.

(s. a. Fehlerprotokoll im Anhang!)

### 5.4.5.5 Testprozessschritte

Im Rahmen der Umsetzung in Testprozessschritte durch die jeweiligen Spezialistentteams aus Entwicklern und Testfallerstellern stellt der Hersteller seine Prüfergebnisse aus der Entwicklungsphase zur Verfügung.

Die Testprozessschritte werden wie folgt in einer Datenbank dokumentiert:

---

ID:	→ Identifikation
Kurztext:	→ Aufspaltung des Testfalls in Testprozessschritte
Schritt1:	→ Beschreibung des Testprozessschrittes
Schritt2:	→ Beschreibung des Testprozessschrittes

---

Beschreibung der Testdaten

....

Abbildung auf den Software-Testablauf

....

Erwartetes Ergebnis

....

Beschreibung der Testinfrastruktur

(Verweis auf Infrastrukturbeschreibung, beim ersten Mal zu erstellen und in zentralem Verzeichnis abzulegen, dann referenzieren)

Testdurchführung

Schritt1:

Datum:

---

Ergebnis:	z. B.: „Die Vertragsversion wird korrekt erstellt“.
	→ <b>Protokoll</b>

---

### 5.4.6 *Abnahmeobjekt: xyz n.1*

#### 5.4.6.1 Referenzdokumente

Abnahmerelevant sind alleinig die Pflichtenhefte, beginnend mit der laufenden Nr. 1 in der unter Templates (Abschn. 5.5) vorgestellten Tabelle. Die übrigen angeführten Dokumente werden Abnahme unterstützend hinzugezogen.

#### 5.4.7 *Abnahmegegenstand*

Abnahmegegenstand sind die Komponenten des Testsystems, wie sie gemäß der Aufstellung durch das Bereitstellungsprotokoll dokumentiert werden. (s. Template)

## **5.4.8 Voraussetzungen**

### **5.4.8.1 Abnahmeort**

Die Abnahme findet statt in den Räumen des Herstellers (Adresse). Für das Abnahmeteam sind entsprechend ausgerüstete Räumlichkeiten (Telefon, LAN-Anschluss, Zugriff auf das Testsystem: Arbeitsstationen, etc.) vorzusehen.

### **5.4.8.2 Abnahmesystem**

Zum Abnahmetermin ist ein über die Steuerungsgruppe eingefordertes Testsystem mit simulierten Schnittstellen und der im Zuge der Testprozessschritte entwickelten Datenladung bereitzuhalten. Dieses Testsystem gliedert sich grundsätzlich in einen für den Test des Produktionsplans dezidierten Server und einen separaten Server für Einzelfunktionstests. Die Software des xyz n.1 unterliegt einem vorhergehenden Customizing nach Vorgaben des Kunden.

Im Bereitstellungsprotokoll werden die Komponenten des bereitgestellten Testsystems spezifiziert.

### **5.4.8.3 Testdaten**

Die Abnahme des Abnahmeobjektes xyz Version n.1 erfolgt unter Verwendung von Testdaten, die im Rahmen der Entwicklung von Testprozessschritten identifiziert, spezifiziert und angelegt wurden.

Diese Testdaten sowie die Gesamtkonfiguration werden vor der Durchführung der Abnahme dupliziert und gesichert, um die Testprozessschritte jederzeit unter reproduzierbaren Bedingungen nachvollziehen zu können.

## **5.4.9 Durchführung**

### **5.4.9.1 Steuerungsgruppe**

(s. Template)

### **5.4.9.2 Abnahmeteam**

(s. Template)

## 5.4.10 Abnahmeprüfungen

### 5.4.10.1 Anwenderfunktionen: Vollständigkeit, Korrektheit

Die Abnahme der Anwenderfunktionen prüft die Korrektheit, die Vollständigkeit sowie das integrierte Zusammenspiel der einzelnen Grundfunktionen im Rahmen der Abbildung von Testfällen auf eine Folge von Testprozessschritten.

Bei jedem Testprozessschritt wird dokumentiert, ob die Software spezifikationsgemäß reagiert. Nach Durchlaufen des jeweiligen Testprozessschritts entscheiden Mitglieder des Abnahmeteam, ob Fehler und mit welcher Priorität diese aufgetreten sind. Dabei werden Fehler folgendermaßen priorisiert:

- **Priorität 1:** Der Fehler führt zum Produktionsstillstand und kann nicht umgangen werden.

Wenn das nicht der Fall ist, gilt die entsprechende Anwenderfunktion als abgenommen.

Ein Fehler gilt als Abnahme ver hindernd, wenn die betriebliche Nutzung der Software stark eingeschränkt ist, weil eine Funktion gar nicht oder so fehlerhaft ausgeführt wird, dass die beabsichtigte Wirkung auch auf einem anderen als dem vorgeschlagenen Wege nicht erreichbar ist.

Während der Abnahme werden solche Fehler innerhalb eines kontrollierten Patcherstellungsverfahrens im Rahmen des Konfigurationsmanagements nachgebessert.

- **Priorität 2:** Der Fehler führt zur Produktionsbehinderung, kann aber umgangen werden.
- **Priorität 3:** Der Fehler führt weder zu Produktionsstillstand noch zu Produktionsbehinderung.

Fehler der Prioritäten 2 und 3 werden als nicht– Abnahme ver hindernd eingestuft. Diese Fehler schränken die betriebliche Nutzung der Software nur bedingt ein. Solche Fehler werden zur Nachbearbeitung in eine Restpunktliste via Fehler-Tracker aufgenommen.

Dabei wird die Liste der identifizierten Fehler einer abschließenden Bewertung unterzogen. Die während der Abnahmephase bereinigten Fehler werden als behoben gekennzeichnet, sofern diese durch die projektinterne QS qualitätsgesichert sind. Bewertungsschema:

---

A	= ohne Mängel
B	= vorbehaltlich der Beseitigung der Mängel abgenommen (die Beseitigung der Mängel wird nachvollziehbar über den Fehler-Tracker verwaltet und von der projektinternen QS qualitätsgesichert).
C	= nicht abgenommen.

---

Nach bzw. während des Durchlaufens aller Testprozessschritte werden die Prüfergebnisse insgesamt dokumentiert. Sollten am Ende der Abnahme keine Abnahme

verhindernden Fehler vorhanden sein, wird ein Protokoll zur Abnahmebestätigung durch das Steuerungsteam erstellt.

### **5.4.10.2 Berechtigungen**

Das Berechtigungskonzept wird anhand eines Testszenarios mit Akzeptanzkriterien nach zwei Gesichtspunkten geprüft:

- Konformität mit den IT-Normen des Auftraggebers,
- Möglichkeit des Anlegens eines Benutzerprofils innerhalb der xyz - Funktionalitäten.

### **5.4.10.3 Schnittstellen-Funktionen im simulierten Schnittstellenumfeld**

Die Testfälle und damit die Testprozessschritte setzen Schnittstellendaten bzw. -funktionalität unter folgenden Gesichtspunkten voraus:

- explizite Testfallfunktionen (Arbeitsanweisungen),
- Inputs aus Schnittstellen zur Weiterverarbeitung,
- Outputs für Fremdsysteme über Schnittstellen.

Im Zuge der Umsetzung der Testfälle in Testprozessschritte wird das entsprechende Testdatenumfeld über Logfiles generiert und ist Voraussetzung für die Prüfung der relevanten Funktionalitäten.

## **5.4.11 Anhänge/Checklisten**

### **5.4.11.1 Prüfprotokoll**

Im Protokoll wird jede relevante Aktion eines jeden Teams fortlaufend mit Datum und Name festgehalten. Relevant sind u. a.:

- Veränderung/Erstellung des Testfalls,
- Definition oder Erstellung der Testdaten mit Verweis auf gemachte DB-Sicherungen,
- Durchführung eines Testprozess-Schrittes mit Angabe der verwendeten Softwareversion und Verweis auf gemachte DB-Sicherungen,
- Fehler in der Software mit Angabe der Trackernummer,
- fachliche Lücke,
- Mitwirkungsleistung erbracht oder nicht erbracht,
- Sonstiges.

### 5.4.12 Bereitstellungsprotokoll

(s. Template)

### 5.4.13 Abnahmeprotokoll

(s. Template)

## 5.5 Templates

Dieses Dokument wurde geprüft von:

Name des Prüfers	Datum	Gegenstand der Prüfung	Unterschrift
Fachbereichsverantwortlicher 1		Vollständigkeit/Richtigkeit	
Fachbereichsverantwortlicher 2		Vollständigkeit/Richtigkeit	
Auftraggeber		Vollständigkeit/Richtigkeit	
QS Lieferant		Vollständigkeit/Richtigkeit	
Teilprojektleiter		Vollständigkeit/Richtigkeit	
QS Kunde		Vollständigkeit/Richtigkeit	

Die in diesem Dokument beschriebene Vorgehensweise wurde genehmigt von:

**Tab. 5.1** Freigabevermerke für Abnahmerichtlinie

Genehmigt von...	Unternehmen	Datum	Unterschrift
Geschäftsleitung	Kunde		
Geschäftsleitung	Lieferant		

Dieses Dokument ist nur gültig, wenn es durch die oben genannte Projektleitung per Unterschrift freigegeben wurde.

**Tab. 5.2** Freigabevermerke

Version	Status	Datum	Verantwortlich	Hinweis
0.1	Zwischenstand			Ersterstellung
0.2	Zwischenstand			Überarbeitung
0.3	Zwischenstand			Überarbeitung
0.4	Zwischenstand			Überarbeitung nach Hinweisen durch ...
1.0	Freigabe		s. o.	

**Tab. 5.3** Referenzdokumente

Lfd-Nr	Datei-Name	Autor	Bezeichnung	Version	Datum
1	PH_xyz.doc	Lieferant	Pflichtenheft Software xyz n.m		
2	LH_xyz	Kunde	Lastenheft Software xyz n.m		
3		Einkauf	Vertrag über Lieferung der Software xyz n.m		
4		QS Kunde	Abnahme-Richtlinie		
5		Lieferant	Ergebnisse Entwicklertests		
6		Tester	Testfallkatalog		
7		Tester	Testprotokolle		
8		Lieferant	Bereitstellungsprotokoll		
9		Lieferant, Tester	Problemspeicher		

**Tab. 5.4** Abnahmegegenstände

Komponente	Inhalt: xyz Version n.m	Umfang
1. Funktionalität	Zum Nachweis des Funktionsumfangs werden die Systemkomponenten für die Version xyz n.m in Anlehnung an den Testfallkatalog geprüft. Grundlage hierfür ist das xyz-Pflichtenheft [PH_xyz].	Komplette Prüfung aller Testprozessschritte
2. Berechtigungen	Berechtigungskonzept für funktions- und datenbezogene Berechtigungen auf Basis von Berechtigungsprofilen	Wie im Pflichtenheft beschrieben
3. Schnittstellenfunktionen	Simulierte Schnittstellen <ul style="list-style-type: none"> <li>• SSt1</li> <li>• SSt2</li> <li>• Export A</li> <li>• Export B</li> <li>• Import C</li> <li>• Import D</li> <li>• ....</li> </ul>	Über Logfiles simulierte Schnittstellen für Import/Export

**Tab. 5.5** Abnahmeobjekte

Abnahmeobjekt xyz n.m		
Abnahme-Maßnahmen	Termin	Verantwortung
Abnahme ankündigen		Anforderungsmanagement
Steuerungsgruppe (SG) einsetzen		Teilprojektleitung
<u>Aktionen:</u>		
<ul style="list-style-type: none"> <li>• Terminplan für Abnahme erstellen</li> <li>• Abnahmeteam einrichten</li> </ul>		
Abnahme durch Abnahmeteam planen		Steuerungsgruppe/QS Kunde
<u>Aktionen:</u>		
<ul style="list-style-type: none"> <li>• Referenzdokumente zusammenstellen und prüfen</li> <li>• Testfälle / Testprozessschritte koordinieren</li> <li>• Testdaten definieren und anlegen</li> <li>• Akzeptanzkriterien festlegen</li> <li>• Terminplan für Abnahme aktualisieren</li> </ul>		

**Tab. 5.5** (Fortsetzung)

Abnahmeobjekt xyz n.m		
Abnahme-Maßnahmen	Termin	Verantwortung
Abnahmesystem vorbereiten		Steuerungsgruppe
<u>Input:</u> Bereitstellungsprotokoll		
<u>Aktionen:</u>		
<ul style="list-style-type: none"> <li>• Dump einspielen</li> <li>• Dokumentation xyz – u. Patchstand</li> <li>• Schnittstellen auf Startposition bringen</li> <li>• Fachlicher Check</li> <li>• Freigabe Tages-Szenarien</li> </ul>		
Teilabnahmen durchführen		alle
<u>Input:</u>		
<ul style="list-style-type: none"> <li>• Testprozessschritte</li> <li>• Ablaufplan</li> <li>• Einzelfunktionen</li> <li>• Berechtigungskonzept</li> </ul>		
<u>Output:</u> Tracker, Testberichte, Problemspeicher		
<u>Aktionen:</u>		
Abarbeiten der Testprozessschritte		

**Tab. 5.6** Zuständigkeitstabelle

Rolle	Name	Org.-Einheit
Leitung	z. B. Vertreter Lieferant	
Leitung	z. B. Vertreter Kund	
	z. B. QS-Leiter Kund	
	z. B. QS-Leiter Lieferant	
	z. B. Berater	
	z. B. Controller	

**Tab. 5.6** (Fortsetzung)

Rolle	Name	Org.-Einheit
Tester	Tester1	
Tester	Tester2	
Tester	....	
Tester	TesterN	

**Tab. 5.7** Bereitstellungsprotokoll

Systemkomponente	Zustand
Serverlandschaft	
Peripherie	
Laufwerke	
Datenbank	
Systemprogramme	
Kommunikationsprogramme	
Abzunehmende Software	
Sonstige Dienstprogramme	
Customizing	
Logfiles	
Abnahme-Verantwortlicher	

Abnahmeprotokoll s. Kap. 3.

## **5.6 Lösungsweg**

Identifizieren der Interessengruppen > gemeinsames Verständnis wecken > Ziele definieren > Verhandlungen führen > Richtlinie verfassen > sachliche Entscheidungen treffen > Vereinbarungen herbeiführen > Richtlinie umsetzen.

# Kapitel 6

## Methodik Daten- und Software-Migration

### 6.1 Abnahme und Daten

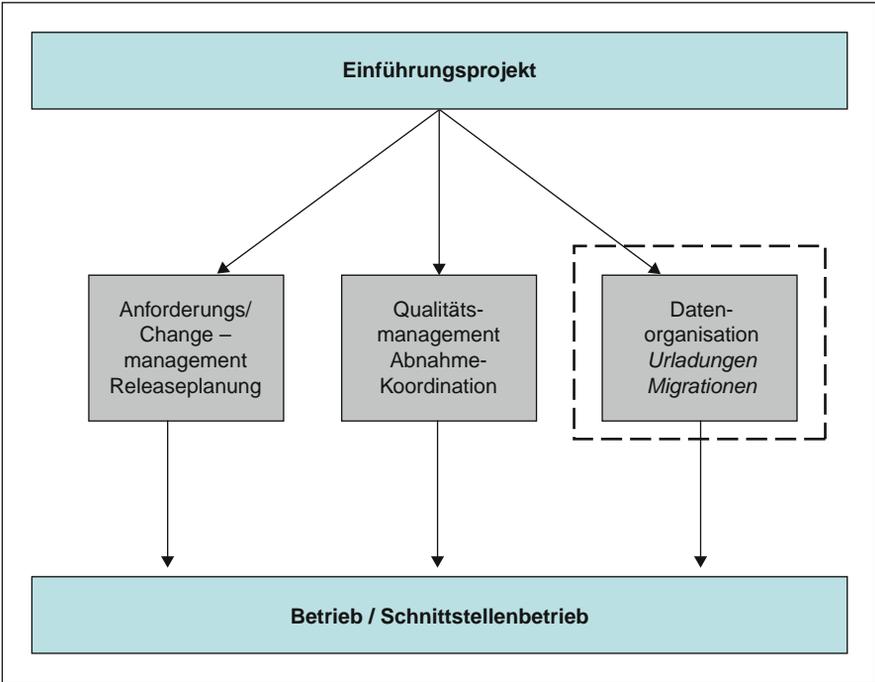
In vielen Fällen können neue Systeme im Zuge einer Einführung nicht mit einer leeren Datenbank starten. Vielmehr sind häufig Beladungen aus Altsystemen notwendig. In solchen Fällen spricht man von Urladungen, die getätigt werden müssen. Bei Software-Updates oder nach Änderungen am Datenmodell müssen dagegen vorhandene Bestände in die neue Software-Version integriert werden. Dabei handelt es sich um Migrationen.

Bei Urladungen müssen die Daten aus den Altsystemen in das Format des Empfängersystems umgewandelt werden. Da es sich meistens um größere Datenmengen handelt, geschieht das per Programm. Es empfiehlt sich daher, den gesamten Urladungsprozess ebenfalls abzunehmen. Das Gleiche gilt für Migrationen. In der Konsequenz bedeutet das mehr Zeit zwischen den Software-Abnahmen und der eigentlichen Produktivsetzung. Es wird dringend empfohlen, diese Zeit einzuplanen.

Es geht dabei nicht nur um die rechentechnisch benötigte Zeit, einen Dump einzuspielen oder ein Datenbank-Update durchzuführen. Vielmehr Zeit wird benötigt zur Prüfung der Datenqualität im Zielsystem, nachdem die Beladungen einmal durchgeführt worden sind. Oftmals haben die Datensätze eine differenzierte Struktur gegenüber den Quellsystemen, oder es sind Felder hinzugekommen oder fortgelassen worden. Die Qualität des Datentransfers misst sich allerdings nicht nur an der Anzahl von Abweisungen während einer Migration, sondern muss zumindest stichprobenartig auch inhaltlich an den transferierten Daten geprüft werden.

Dem Ganzen voraus geht natürlich die Selbstverständlichkeit, dass die Altdaten im Vorfeld schon so aufbereitet sind, dass diese von „totem Holz“ und Inkonsistenzen bereinigt sind. Diese Arbeit sollte nicht im Zielsystem geschehen.

Auch bei den Themen Urladungen und Migration spielt wie überall und immer Zeitdruck eine große Rolle. Dadurch sollte man sich nicht entmutigen lassen. Auch hier muss die Arbeit so oder so getan werden. Was an Vorbereitungen versäumt wird, muss später durch kostspielige Datenbereinigungsprojekte wieder eingefangen werden. Es wird deutlich, dass die Themen Urladung und Migration ein eigenes



**Abb. 6.1** Teilprojekt Daten-Migration/Urladung

Teilprojekt in der Gesamtprojekt-Organisation erfordern. Die Abb. 6.1 zeigt eine mögliche Einordnung.

## 6.2 Migrationsvorhaben

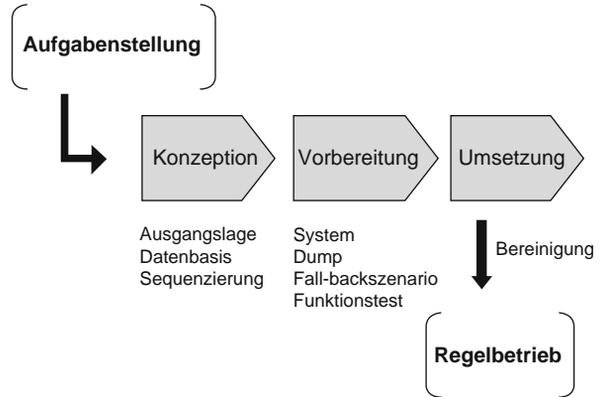
Die Systematik eines Migrationsvorhabens ist aus Abb. 6.2 ersichtlich.

Die Aufgabenstellung kann in Komplexität und Aufwand variieren. Bei Änderungen am Datenmodell spielen zusätzliche Gesichtspunkte eine Rolle gegenüber einem einfachen funktionalen Releasewechsel. Entsprechend der dezidierten Aufgabenstellung wird ein komplettes Migrationskonzept entwickelt und als Lastenheft dokumentiert. Dieses Konzept bedarf der Freigabe durch alle Beteiligten:

- Fachbereiche,
- IT,
- Gesamtprojektleitung,
- Teilprojektleitung Migration.

Das Dokument basiert auf folgender Grobstruktur:

- der fachliche und technische Hintergrund der Migration,
- die Ausgangsdatenbasis – strukturell und inhaltlich,

**Abb. 6.2** Migrations-systematik

- die Zieldatenbasis – strukturell und inhaltlich,
- die Reihenfolge der technischen Migrationsschritte mit Rückfallszenarien.

Für jeden Migrationsbaustein werden die Zuständigkeiten für Kommunikation und beteiligte Subsysteme festgelegt. Wichtig ist hierbei – wie auch sonst bei den Abnahmen – die gegenseitige Akzeptanz der Zuständigkeiten. Auch hier sollte alles im Vorfeld durch ein entsprechendes Kick-off geregelt werden. Dabei ist die Rolle des „internen Auftraggebers“ zu klären. Im Gegensatz zu den klaren Verhältnissen bei Release-Auslieferungen durch externe Lieferanten kommen die Anstöße zur Migration aus dem Unternehmen selbst. Sie sind ebenso häufig an Dienstleister gerichtet, die im selben Unternehmen sitzen. Diese Beziehungen und Zuständigkeiten müssen klar geregelt sein – am besten über Service Level Agreements.

Ein weiterer wichtiger Baustein des Lastenheftes ist eine Checkliste, die nicht nur die zeitliche Abfolge der Migrationsschritte dokumentiert, sondern ebenso alle vorbereitenden Maßnahmen, die das Vorhaben erst ermöglichen.

Anders als bei Funktionalitäten müssen die abzunehmenden Inhalte definiert werden. Dazu gehören Fragen der Datenkonsistenz, Toleranzen und Datenmengen – also Qualität und Quantität. Ein entsprechendes Daten-Controlling ist aufzusetzen.

### 6.3 Planung und Durchführung

Eine sorgfältige Vorbereitung erübrigt später kostspielige Wiederholungen und Datenbereinigungsmaßnahmen. Das System für die Migration muss den doppelten Datenbestand bewältigen können. Zu entscheiden ist, ob ein Testlauf auf einem separaten Testsystem mit einer repräsentativen Datenselektion vorab durchzuführen ist (empfehlenswert!). Vor Migrationsbeginn ist ein Dump des Altbestandes zu ziehen, der jederzeit unter kontrollierten Bedingungen den vorherigen Zustand wieder herstellen kann.

Die Planung beinhaltet folgende Komponenten:

- Zeitlicher Ablauf,
- Ressourcen: Personal und Hardware,
- Organisationsstruktur,
- Migrationsprozesse,
- Dokumentation,
- Schulung,
- Abnahme,
- Inbetriebnahme.

Im Detail ist besondere Aufmerksamkeit zu legen auf:

- Abgrenzung Verantwortlichkeiten (Fachbereiche/IT/Projekt),
- eindeutige Beschreibung des Auftrags,
- eindeutige Beschreibung des Migrationsgegenstandes,
- Risiken bezüglich der Außenwirkung bei Datenproblemen,
- Sicherstellung der Verfügbarkeit der Teammitglieder,
- Budgetierung,
- Festlegung der Abnahmekriterien.

Vor der ersten Datenbeladung müssen folgende Voraussetzungen geschaffen werden:

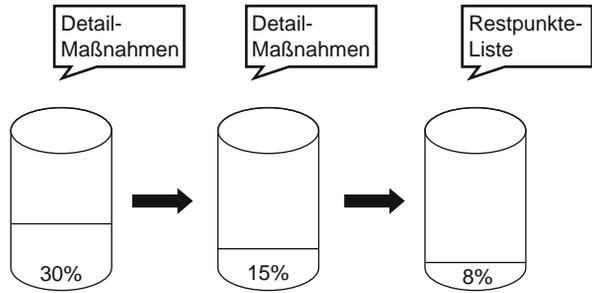
- Lastenhefterstellung,
- Identifikation des Altbestandes,
- Bereinigung des Altbestandes nach Inkonsistenzen, Datenfehlern, Redundanzen,
- logischer Abgleich der alten und neuen Datenstruktur,
- Pflichtenheft für das Anpassungsprogramm,
- Spezifikation für die Systemumgebung, auf der die Umladung stattfinden soll,
- Fall-back-Szenario definieren,
- Datencontrolling im Vorfeld,
- Rechtzeitige Anforderungen an zusätzlichen Daten z. B. für das Customizing,
- Bereitstellung Testmandant,
- Customizing des Migrationsmandanten,
- Bereitstellungstermin Umladungs SSt,
- Test der Umladungsschnittstelle.

Nach der Durchführung der Umladung muss das Ergebnis validiert werden. Dazu gehört eine intensive Qualitätsprüfung der beladenen Daten. Diese Prüfung wird innerhalb des Teilprojektes erfolgen unter Hinzuziehung der Kompetenz des Fachbereichs, wo erforderlich.

In einer ersten Grobplausibilisierung wird die Abweisungsquote der zu ladenden Daten ermittelt. Liegt diese unter einer vorher festgelegten Marge, gilt die Umladung zunächst als erfolgreich. Im zweiten Schritte sollten zwei Dinge geschehen:

- Stichprobenuntersuchungen des neuen Datenbestandes auf Korrektheit,
- Analyse der abgewiesenen alten Datensätze nach Gründen.

Abb. 6.3 Datenbereinigung



In Abhängigkeit von den Detailanalysen erfolgen manuelle Nachbesserungen oder Fehlerbehebungen im Umladungsprogramm mit nachfolgender Wiederholung der Umladung. In vielen Fällen ist auch eine manuelle Nachpflege der geladenen Daten erforderlich, wenn z. B. neue Datenfelder hinzugekommen sind, die im Alt-system nicht vorhanden waren.

Bei Migrationen greifen ähnliche Kriterien zur Validierung wie bei den Umladungen. Auch hier sind Nachbereinigungen nicht ausgeschlossen. Es kann sogar in der Natur der Aufgabenstellung liegen, dass mit größeren Datenbereinigungen zu rechnen ist, da der programmierte Migrationsalgorithmus von vorne herein nicht in der Lage ist, eine saubere Datenabbildung durchzuführen. In solch einem Falle wird nach der Migration (und das gilt gleichermaßen auch für Umladungen selbst) ein Teilprojekt „Datenbereinigung“ aufgesetzt, in dem nach definierten Qualitätskriterien die fehlerhaften Datensätze im Neusystem korrigiert werden. Leider kann es passieren, dass das im laufenden Regelbetrieb durchgeführt werden muss. Der Bereinigungsfortschritt ist in Abb. 6.3 dargestellt.

## 6.4 Aktivitätentracking

Am einfachsten stellt man alle erforderlichen Tätigkeiten in eine so genannte Taskliste (EXCEL) ein (s. Template):

Für die Statusinformation reichen grundsätzlich drei Kategorien:

- Offen,
- in Arbeit,
- erledigt.

Erledigte Tasks können der Übersicht halber noch als durchgestrichen markiert werden.

Ein weiteres unerlässliches Tracking-Tool ist das Instrument der Budgetverfolgung. Das Format dazu wurde bereits im Kap. 2 vorgestellt.

Ein Sonderthema ist das Customizing eines geeigneten Testmandanten. Das wird in Kap. 7 vorgestellt.

## 6.5 Templates

Folgende Templates sind nützlich: Checkliste (Abb. 6.4), Daten-Contolling (Abb. 6.5), Taskliste (Abb. 6.6)

**Abb. 6.4** Checkliste

Hauptaufgaben	Status
Zeitplan aufstellen	
Quellsystem identifizieren	
Projekt aufsetzen	
Migrationsprozess festschreiben	
Verantwortlichkeiten festlegen	
Risiken abschätzen	
Budgetierung	
Abnahmekriterien definieren	
Zielmandanten bereitstellen	
Abnahme	
Datenqualitätssicherung	
Inbetriebnahme	

Quelldatei1	Zieldatei1	Anzahl Abweisungen	Abweisungsgründe	Bereinigungsmaßnahmen	Fortschritt %	Status	Zieltermin	Verantwortlicher
Quelldatei2	Zieldatei2							
...								
QuelldateiN	ZieldateiM							

Abb. 6.5 Daten-Controlling

Aktivität	Termin	Verantwortlicher	Status
Migrationsplanung			
Lastenheft erstellen			
Altbestand identifizieren			
Altbestand bereinigen			
Datenstrukturabgleich			
Pflichtenheft für Migrationsprogramm			
Systemumgebung festlegen			
Fall-back-Szenario			
Datencontrolling aufsetzen			
Customizing durchführen			
Testmandant bereitstellen			
Test der Migrationsschnittstellen			
Migration durchführen			
Stichproben ziehen			
Datenanalyse			

Abb. 6.6 Taskliste

## 6.6 Lösungsweg

Zielsetzung: Datenqualität und Konsistenz > Planung Migrationsvorhaben als Teilprojekt: Verantwortlichkeiten und Aktivitäten > Durchführung und Tracking > Datenqualitätssicherung.

# Kapitel 7

## Sonderfälle

Die im Folgenden separat behandelten Sonderfälle können als Einzelproblemlösungen auftreten oder aber Teil des allgemeinen Abnahmeverfahrens sein, innerhalb dessen sie einen prominente Teilaspekt darstellen. Deshalb sind eventuelle Redundanzen mit bereits diskutierten Abschnitten beabsichtigt, wo nicht ein besonderer Hinweis auf ein Vorkapitel gegeben ist.

### 7.1 Abnahme von Schnittstellen

#### 7.1.1 *Sonderfall Schnittstellen*

Die Abnahme von Schnittstellen kann wegen höherer Aufwendungen zu einer Abweichung vom beschriebenen Abnahmeverfahren führen. Im Wesentlichen handelt es sich bei diesen Abweichungen um

- eine Beteiligung des Fachbereichs bereits in der Entwicklungsphase und/oder
- um die Bereitstellung besonderer Daten.

#### 7.1.2 *Vollständigkeit der Testlandschaft*

Die Abnahme von Schnittstellen außerhalb oder im Zusammenhang mit der Bereitstellung eines neuen Releases stellt besondere Herausforderungen an das Abnahmeverfahren aus folgenden Gründen:

- Bereitstellung von Schnittstellendateien zum Export oder Import im richtigen Format und einer ausreichenden Anzahl von Datensätzen, um einen Massentest zu gewährleisten,
- Verfügbarkeit von Testumgebungen von Partnersystemen,
- Ankoppelung von Partnersystemen,

- Verfügbarkeit von Testpersonal auf Seiten der Partnersysteme,
- Erzeugung von Logdateien zur Auswertung der Schnittstellenläufe,
- Alternativ: Erzeugung von synthetischen Datensätzen.

### **7.1.3 *Eigenständiges Verfahren***

Der durch diese zusätzlichen Anforderungen erhöhten Komplexität wird durch ein eigenständiges, bzw. aufgewertetes Verfahren Rechnung getragen. Dabei spielen insbesondere folgende Maßnahmen eine zentrale Rolle:

- Simulation von Schnittstellenläufen,
- frühzeitige Einbindung in die Entwicklungsphase,
- Auswertung von Testprotokollen aus der Entwicklungsphase,
- Kombination aus synthetischen und Echtdatenabzügen für die Abnahme,
- Einbindung dieser Elemente in den Standard-Abnahmeprozess.

### **7.1.4 *Simulationen und mehrstufige Verfahren***

#### **7.1.4.1 *Simulation***

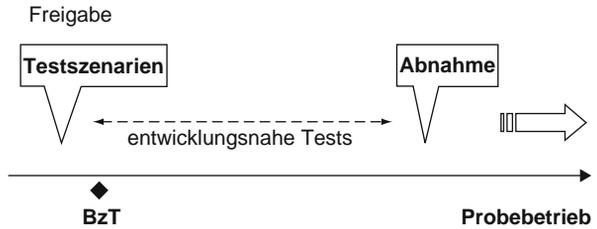
Ein echter Schnittstellentest setzt die Verkopplung von zwei oder mehreren Partnersystemen voraus. Die logischen Adressierungen der Systeme müssen bekannt und gegenseitige Zugriffsberechtigungen frei geschaltet worden sein. Sodann müssen entsprechende Datenpakete auf dem exportierenden System erzeugt und die Schnittstelle angestoßen werden, um den Transfer zu ermöglichen. Danach ist der Import ins andere System zu prüfen und mit dem Export abzugleichen.

Aufgrund häufig auftretender technischer (gleichzeitige Verfügbarkeit aller Systemkomponenten) und politischer (Systemhoheit) Restriktionen kann der oben beschriebene Vorgang abgekürzt werden, um dennoch plausible Ergebnisse zu erzielen. Hierzu bedient man sich der Schnittstellensimulation.

Das bedeutet, dass auf dem gleichen Hostrechner eine Exportdatei sozusagen synthetisch aufgebaut wird, welche das erwartete Übertragungsformat besitzt. Dabei sind gewisse Voraussetzungen zu beachten:

- Die Datei muss ausreichend groß sein, um auch Performancetests durchführen zu können. Dazu reicht häufig eine manuelle Erzeugung der Datensätze nicht aus, sodass ein Generierungsprogramm erstellt werden muss. Dabei sind die Kosten dafür zu beachten.
- In die Datei müssen gängige Fehler eingebaut werden, wie sie aus den tatsächlich anfallenden Geschäftsvorfällen zu erwarten sind. Diese Fehler dienen dazu, Schnittstellenabweisungen zu provozieren.

Abb. 7.1 BzA Schnittstelle



- Diese künstliche Testdatei wird nun über den Schnittstellenlauf des Imports in das abzunehmende System hineingezogen und das Ergebnis anschließend analysiert.
- Umgekehrt kann man auch einen tatsächlich erzeugten Export zu einem Fremdsystem in eine vor-Ort-Datei kopieren und diese dann anschließend mit dem zu erwartenden Ergebnis abgleichen.

#### 7.1.4.2 Einbindung in die Testphase

Wie an anderer Stelle erläutert, geht der eigentlichen Abnahmephase durch den Endkunden eine entwicklungsnahe Testphase während der Realisierung voraus. Während dieser Testphase finden Tests statt, die durch die Entwickler selbst vorgenommen werden. Im Zuge der entwicklungs-eigenen Qualitätssicherung werden solche Tests in Skripten dokumentiert.

Um nun für die Abnahme selbst Zeit zu gewinnen, bietet es sich an, dass Vertreter des Kunden vor der eigentlichen Bereitstellung der Schnittstelle zur Abnahme bereits den Entwicklern zur Seite gestellt werden, um die entwicklungs-nahen Tests zu begleiten. Das ist schematisch in der Abb. 7.1 dargestellt:

#### 7.1.4.3 Schnittstellenabnahme: Zeitphasenplan

Mit BzT ist der Termin der Bereitstellung zum Test gemeint. Hierbei handelt es sich, wie schon erwähnt, um interne Tests. Vor der eigentlichen Abnahme kann der Kunde diese Tests begleiten. Das ist Verhandlungssache mit dem Software-Lieferanten. Da aber eine zügige Einführung in der Regel im Interesse beider Seiten ist, wird die Bereitschaft lieferantenseitig meistens da sein.

Gelangen die entwicklungs-nahen Schnittstellentests zu einem für beide Seiten positiven Abschluss, kann die Abnahme unter der Auflage eines Probebetriebes gewährt werden. Das ist insofern sinnvoll, als die Testbedingungen in der Entwicklung andere sind als etwa jene bei einer formalen Kundenabnahme. Um also mehr Sicherheit zu erlangen, wird nach der Inbetriebnahme für den Echtbetrieb eine definierte Zeitstrecke als Probebetrieb festgelegt, während der noch die üblichen

Gewährleistungsbedingungen des Lieferanten Gültigkeit haben (kostenlose Nachbesserung bei eventuell auftretenden Fehlern).

### Stufiger Prozess

Insgesamt sind zwei Szenarien vorstellbar, die in der Praxis zu den oben angeführten Ergebnissen führen:

1. Schnittstellenabnahmen in Phase mit sonstigen Softwareabnahmen (z. B. Releases) und
2. Schnittstellenabnahme als Sonderaktion.

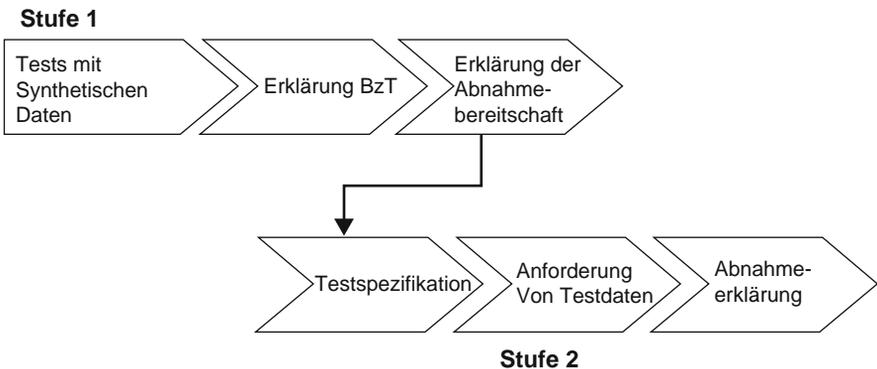
Die Abb. 7.2 beschreibt den 1. Fall.

Stufe 1: Der Lieferant testet mit synthetischen Daten gemäß Pflichtenheft und seinen eigenen Testspezifikationen. Im qualitätsgesicherten Zustand wird die Bereitstellung zum Test an den Kunden erklärt, der wiederum die generelle Abnahmebereitschaft bestätigt.

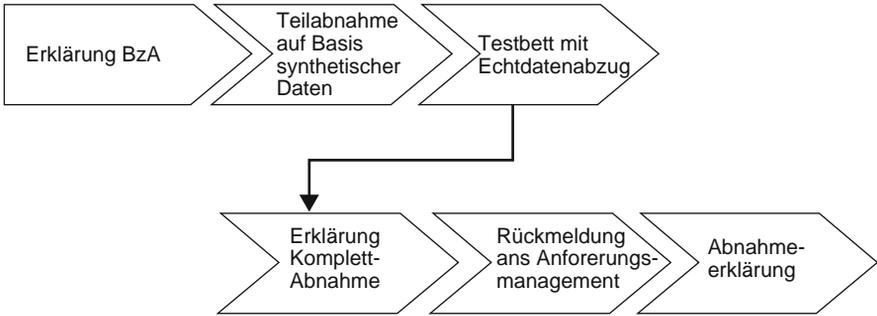
Stufe 2: Der Kunde erhält Einsicht in die Testspezifikationen der Entwicklung und fordert seinerseits die notwendigen Testdaten an. Nach erfolgter Prüfung wird die Abnahme erklärt.

In der nächsten Abb. 7.3 ist der Sonderfall beschrieben.

Der Lieferant testet wie oben mit synthetischen Daten gemäß freigegebenem Pflichtenheft und seinen eigenen Testspezifikationen. Er erklärt dann wie bei einer ganz normalen Abnahme die Bereitstellung zur Abnahme. Im ersten Schritt erfolgt die Abnahme durch den Kunden mit synthetischen Daten. Wegen des häufig hohen Aufwands wird erst nach erfolgreicher Prüfung ein Testbett mit Echtdateien erstellt. Im positiven Falle wird dann durch den Kunden die Abnahme aus beiden Verfahren (synthetisch und echt) bestätigt. Dabei sind Fehler aus den verwendeten Datenquellen für die abschließende Beurteilung irrelevant. Es folgt die Rückmeldung an das Anforderungsmanagement sowie die formale Abnahmeerklärung.



**Abb. 7.2** 2-stufiges Verfahren



**Abb. 7.3** Gesonderte Schnittstellenabnahme

### Logdateien-Analyse

Es gibt verschiedene Instrumente, mit deren Hilfe die Ergebnisse von Schnittstellentests analysiert werden können, darunter:

- funktionales Ergebnis,
- Vergleich Export-Import,
- Analyse der Logdateien.

Der Abgleich eines funktionalen Ergebnisses, beispielsweise die Durchführung einer Plausibilitätsprüfung, wird analog der Abnahme normaler Funktionalitäten durchgeführt und soll hier deshalb nicht weiter erläutert werden.

Die Instrumente „Vergleich Export-Import“ und „Analyse der Logdateien“ müssen im Zusammenhang gesehen werden. Der Vergleich der Daten, die aus einem System exportiert und in ein anderes importiert werden, bezieht sich sowohl auf die Gesamtzahl der betreffenden Datensätze als auch auf die Qualität des Imports, d. h. die formale Richtigkeit der Übertragung. Treten dabei Fehler auf, kann die Analyse der zugehörigen Logdateien eine große Hilfe sein. Dazu muss die Logfile-Registrierung beim Test angestoßen werden.

Logdateien können in beliebigen Formaten vorgehalten werden. Deshalb ist es wichtig, den Entwicklern ein lesbares Format vorzugeben, in das die Informationen hineingeschrieben werden. Das Format sollte es außerdem erlauben, den Inhalt mit vorhandenen Auswertetools zu bearbeiten (verdichten, selektieren etc.). Solche Tools können EXCEL-Tabellen und z. B. ACCESS-Datenbanken sein. Die Informationen, die dort hineingeschrieben werden, betreffen alle Transaktionen, die während des Schnittstellenbetriebs auftreten. Aber auch diese kann man durch geschicktes Filtern auf die relevanten reduzieren.

Treten nun Abweisungen an der Schnittstelle auf, sodass der Import nicht identisch mit dem Export ist, geben die Logdateien Auskunft über den Grund der Abweisungen, da der zugehörige Eintrag normalerweise einen Fehlerkommentar des Systems enthält. Auf diese Weise geben diese Informationen die Richtung vor, in der nach Ursachen geforscht werden kann.

### 7.1.5 Lösungsweg

*Schnittstellen-Abnahme eigenständig planen > alle beteiligten Systeme einbeziehen > Sonderverfahren je nach Konstellation > mehrstufiges Vorgehen > Logdateien-Analyse.*

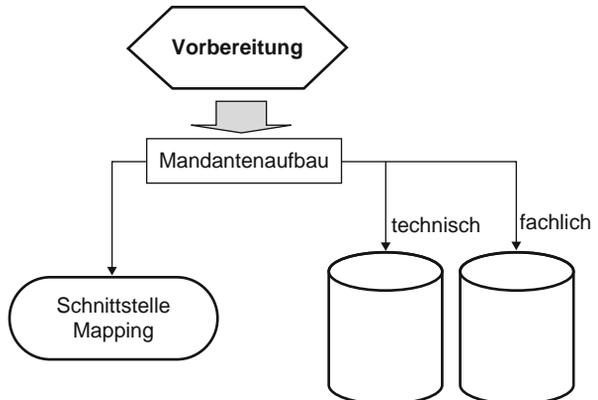
## 7.2 Customizing von Testsystemen

### 7.2.1 Simulation der Produktionsumgebung

Neue Software-Updates werden selbstverständlich mit Blick auf ihren Produktiveinsatz innerhalb einer ganz bestimmten Geschäftsumgebung des Kunden erstellt und zur Abnahme gegeben. Das bedeutet, dass gewisse kundenspezifische Parameter auch schon in der Abnahmephase gesetzt werden müssen. Aus Effizienzgründen werden diese Parameter nicht fest „verdrahtet“, d. h. im Code hinterlegt, sondern variabel gestaltet. Zum Aufbau einer sinnvollen Testumgebung gehört also auch die Anpassung der abzunehmenden Funktionalitäten auf die zukünftige Nutzung hin. Diesen Vorgang nennt man Customizing. Das Customizing der Abnahmeumgebung birgt noch den weiteren Vorteil, dass in dieser Phase auch das Customizing selbst abgenommen werden kann, sodass für die Produktivsetzung ein Abzug der Testumgebung verwendet werden kann.

### 7.2.2 Fachliches Know-how

Die Abb. 7.4 zeigt schematisch, was beim Customizing berücksichtigt werden muss.



**Abb. 7.4** Customizing Kriterien

### 7.2.3 *Eigenständiger Meilenstein*

Nach einer Vorbereitungsphase (s. u.) erfolgt der Aufbau des Testmandanten zunächst als leere Basisumgebung ohne Datenbeladung. Bevor die ersten Daten beladen werden können, muss die kundenspezifische Struktur angelegt werden. Das dazu erforderliche Customizing gliedert sich in einen technischen und einen fachlichen Part. Gesondert betrachtet werden müssen die Schnittstellen, weil hierbei auch die Partnersysteme in den Blick genommen werden müssen. Weitere Spezialthemen sind die Behandlung von Fehlercodes sowie das Layout von Logdateien.

Vor Abnahmebeginn bzw. während oder spätestens nach der Datenbeladung sollte das Customizing gesondert abgenommen werden. Nach der Abnahme kann dann der Transfer für das Produktivsystem durchgeführt werden.

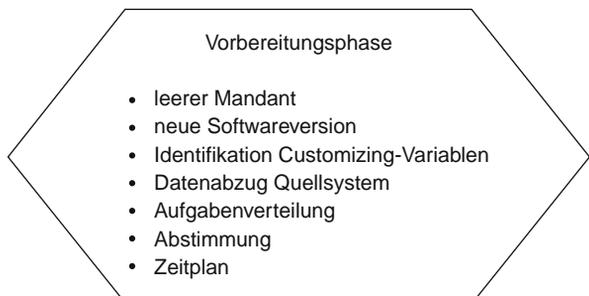
#### 7.2.3.1 Vorbereitungsphase

Für die Vorbereitung stimmen sich Vertreter des Rechenzentrumsbetriebs, Entwickler und Enduser unter der Leitung der Abnahmekoordination ab. Dazu wird die Checkliste aus Abb. 7.5 erstellt.

Ausreichend vor dem BzA-Termin werden die wichtigsten Handlungsfelder mit Zuständigkeiten und Terminen versehen. Insbesondere wird festgelegt:

- die Auslegung des Testmandanten mit den notwendigen Zugriffsrechten für die Abnahmetester; seine Adressierung,
- letzter Status der neuen Softwareversion mit allen endgültigen Komponenten (GUI, Funktionen, Schnittstellen),
- Identifikation der Customizing-Variablen (s. u.),
- Herkunft des Datenabzugs für die Erstbeladung: Quellsystem (s. u.).

Die Customizing-Variablen teilen sich in technische und fachliche. An dieser Stelle ist es wichtig, dass die inhaltliche Herkunft dieser Variablen festgelegt wird. Dazu ist es mitunter notwendig, andere Fachbereiche anzusprechen, z. B. bei Buchungskreisen. Neben der inhaltlichen Zuordnung sollte die personelle Verantwortung für die Beschaffung dieser Informationen festgehalten werden.



**Abb. 7.5** Vorbereitungscheckliste

Es kann durchaus sein, dass die inhaltliche Beladung aus mehreren Quellsystemen erfolgen muss. Dann muss die exakte Reihenfolge ermittelt werden, da es Abhängigkeiten zwischen den einzelnen Beladungen geben kann.

### **7.2.4 Einbindung der Fachbereiche**

Technisches und fachliches Customizing können parallel laufen. Mitunter sind allerdings bereits technische Annahmen bei der Erstellung der Basiskonfiguration eingeflossen. Hierbei handelt es sich um Variablen wie Tabellenkonfiguration, Datenpuffer, Konfigurationsdaten für job streams, logische Adressierungen, Dateipfade usw.

Das fachliche Customizing ist sehr anwendungsspezifisch und dazu noch abhängig von den unternehmensrelevanten Geschäftsvorfällen und organisatorischen Parametern. Eine wesentliche Komponente betrifft die ganze Regelung der Zugriffsrechte auf Basis vorgegebener technischer Möglichkeiten. Also: wer darf welche Daten sehen, ändern, welche Funktionen ausführen etc.? Dabei geht es dann nicht nur um die Berechtigungen an sich, sondern konsequenterweise auch um eine entsprechend angepasste Menu-Steuerung: dem Enduser sollen tatsächlich nur solche Funktionen angeboten werden, mit denen er auch arbeiten kann.

Weitere Spezialisierungen betreffen Locking-Mechanismen auf Datenfelder, sprechende Transaktionscodes, Nummernkreise, Buchungsschlüssel, Langtexte, Wertgrenzen, Mandantenstrukturen, nicht standardisierte Reports u. ä. Da hier eine große Vielfalt in Abhängigkeit der eigentlichen Anwendung herrscht, kann an dieser Stelle keine vollständige Liste vorgeschlagen werden.

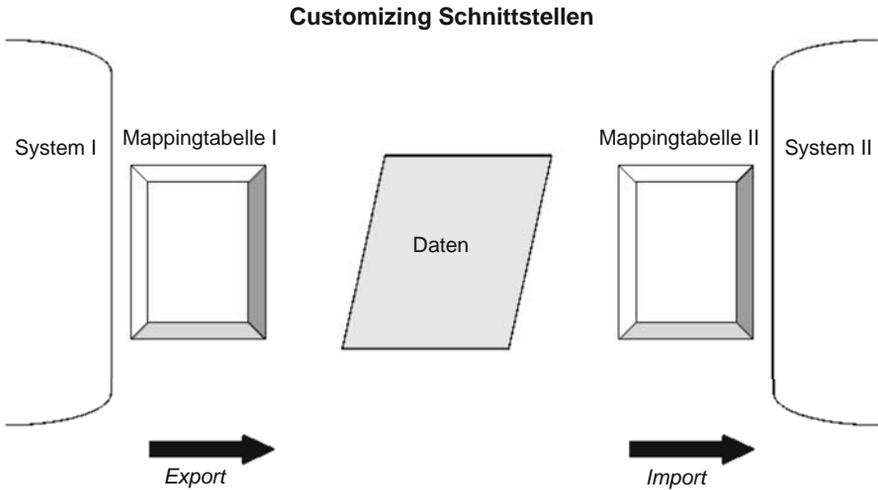
Zu beachten ist, dass das fachliche Customizing in größeren Unternehmen nicht von Einzelverantwortlichen vorgegeben werden kann, sondern nur im Konsens durch Beiträge aus unterschiedlichen Organisationseinheiten erarbeitet werden muss. Der Aufwand im Vorfeld zur Abnahme ist dabei nicht zu unterschätzen.

#### **7.2.4.1 Schnittstellen**

Eine Besonderheit bei den Customizing-Anforderungen bilden die Schnittstellen. Das ist schematisch in Abb. 7.6 dargestellt.

Nur in den seltensten Fällen korrespondieren die Datenformate auf beiden durch eine Schnittstelle verbundenen Systemen im Verhältnis 1:1. Meistens werden Schnittstellendatensätze auf dem exportierenden System aus den entsprechenden Tabellen zusammengefügt, die die inhaltlichen Informationen hergeben. Diese Daten werden dann in einem Exportformat an der Schnittstelle zur Verfügung gestellt.

Beim Import nimmt sich das korrespondierende System die Informationen, die es benötigt und baut sie so um, dass diese in sein Datenmodell passen. Es ist also eine Korrespondenz zwischen beiden Systemen im Sinne einer Formatanpassung herzustellen. Das geschieht über so genannte Mapping-Tabellen.



**Abb. 7.6** Customizing Schnittstellen

In den Mapping-Tabellen werden zu jedem zu übergebenden Datenattribut entsprechende Formatvorgaben bzw. zu belegenden Inhalte abgelegt. Auf diese Weise werden die Exportdatensätze erzeugt. Auf der Gegenseite wandelt die Import-Mapping-Tabelle die übergebenen Formate entsprechend um. Beim Export erübrigt sich häufig eine Mapping-Tabelle, wenn die Exportschnittstelle entsprechend ausgerichtet ist.

Die Inhalte der Mapping-Tabellen können technischer und fachlicher Natur sein: z. B. Vertragskennzeichen, Tarifklassen, Länge von Postadressen etc. All das ist stark abhängig von den Anwendungen, lässt sich also nicht generalisieren.

#### 7.2.4.2 Fehlercodes

Es ist allgemein bekannt, dass systemgenerierte Fehlermeldungen für den Benutzer unbrauchbar sind. Aus diesem Grunde erzeugen die meisten Anwendungen über entsprechende Routinen ihre eigenen Fehlermeldungen. Sowohl was Sprache als auch Inhalte von dezidierten Fehlermeldungen betrifft, sind die entsprechenden Meldungsdateien Gegenstand eines eigenen Customizing.

#### 7.2.4.3 Logdateien

Logdateien für die Aufzeichnung von Transaktionen bedürfen in zweierlei Hinsicht spezifischer Ausrichtung:

- Formatierung,
- Selektion.

Bei der Formatierung ist darauf zu achten, dass die Einträge möglichst von technischen Informationen befreit werden. D. h. alles, was Bezug zum Transaktionscode und zu Datenverschlüsselungen, Formatangaben usw. hat, sollte ausgespart werden. Lediglich aussagekräftige Textinformationen sollten hinterlegt werden, die dem Auswerter sofort einen Hinweis auf das tatsächliche Transaktionsgeschehen geben.

Die Selektion zur Aufzeichnung von Transaktionen sollte sich auf solche beschränken, die für eine spätere Analyse interessant sind. Nebensächliche Funktionsaufrufe wie „Datumsanzeige“, „Rückkehr zum Hauptmenue“ u. a. sollten ausgeblendet werden.

#### **7.2.4.4 Abnahme des Customizing**

Ganz ähnlich wie bei funktionalen Abnahmen lässt sich ein korrektes Customizing nur durch entsprechende Testfälle verifizieren. Diese sollten in Abstimmung mit den fachlichen Endusern definiert und durchgeführt werden. Bei den textlichen Anpassungen genügen Inaugenscheinnahmen. Voreingestellte Menu-Pfade müssen durchgespielt, Logdateien angestoßen werden. Aufwendiger kann es bei der Abnahme von Mapping-Tabellen werden. Dabei ist eine tiefer gehende Analyse von Fehlerprotokollen erforderlich, um Fehler des Customizing von anderen Schnittstellenabweisungen zu unterscheiden.

Einstellungen des technischen Customizing können sich manchmal erst unter Vollbetrieb der Anwendung bewähren (Konfigurationsparameter), es sei denn man hat eine komplette Systemspiegelung für Massentests zur Verfügung. Aber auch dann kann eine korrekte Adressierung von Zielsystemen erst beim realen Einsatz überprüft werden.

Die Abnahme des Customizing erfolgt prinzipiell nach den gleichen Regeln wie sonst mit entsprechender Abschlussdokumentation.

#### **7.2.4.5 Übergabe**

Nach dem oben Beschriebenen kann Customizing also eine aufwendige Angelegenheit sein. Deshalb sollte man sicherstellen, dass dieser Aufwand nur einmal betrieben wird. Da er ohnehin zunächst auf dem Testmandanten stattfindet, ist dieser gleichzeitig die Basis für die spätere Produktivsetzung. Die Verfahrensweise ist wie folgt:

1. Aufbau des Testmandanten (leere Datenbank)
2. Technisches Customizing
3. Fachliches Customizing
4. Datenbeladung
5. Dump
6. Abnahme
7. Software-Korrekturen
8. Aufbau des Produktivmandanten (leere Datenbank)
9. Dump einspielen
10. Freigabe zur Inbetriebnahme

Es kann sein, dass bestimmte Elemente des technischen Customizing unabhängig vom Dump bzw. erst nach der Beladung möglich sind. Das ist von der Architektur der Anwendung abhängig. Insofern ist die o. a. Reihenfolge idealtypisch zu interpretieren.

### 7.2.5 *Templates*

**Tab. 7.1** Gesamtcheckliste Customizing

<b>Aktivität</b>	<b>Bis wann</b>	<b>Status</b>
Auslegung Testmandant		
Letzter Status Softwareversion		
Identifikation Customizing-Variablen		
Quellen Datenabzüge		
Basiskonfiguration		
Technisches Customizing		
Customizing Schnittstellen		
Fachliches Customizing		
Beladungen		
Dump		
Abnahme Customizing		
Software-Korrekturen		
Aufbau Produktivmandant		
Dump einspielen		
Freigabe zur Inbetriebnahme		

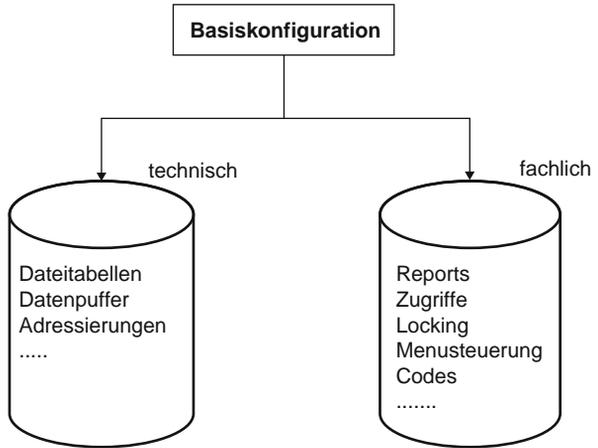
### 7.2.6 *Lösungsweg*

*Abbildung entsprechend Produktivumgebung > Einbeziehung Fachbereichs-Know-how > Meilensteinplanung mit Checkliste > Sonderfälle berücksichtigen (Schnittstellen, Fehlercodes, Logdateien) > Abnahme > Inbetriebnahme.*

## 7.3 Aufbau eines Testmandanten

### 7.3.1 *Technologische Voraussetzungen*

In diesem Abschnitt geht es um den Aufbau eines neuen Mandanten (s. Abb. 7.7), der eine bereits anderswo im Einsatz befindliche Software nutzen soll. Die Basissoftware wurde also bereits schon einmal abgenommen. Insgesamt sind beim Mandantenaufbau die Aktivitätenblöcke

**Abb. 7.7** Mandantenaufbau

- Basiskonfiguration,
- Technisches Customizing
- Fachliches Customizing,
- Beladung(en) (z. B. aus Vorversionen des Produkktivsystems; wird hier nicht weiter betrachtet)

zu berücksichtigen.

### 7.3.2 Geeignete Umgebung

Die Basiskonfiguration wird auf einer leeren Hardware bzw. Partition aufgespielt und beinhaltet die neue zu testende Softwareversion (Release, Service Pack etc.), sowie eine initialisierte, aber leere Datenbank. Diese Umgebung wird zur Verfügung gestellt und kann nunmehr entsprechend den spezifischen Kundenanforderungen weiter vorbereitet werden.

### 7.3.3 Eigenständiger Meilenstein

Im Folgenden werden die notwendigen Schritte beschrieben, die für den Aufbau eines neuen Mandanten vorhandener Anwendungen zu beachten sind. Dabei spielen die Aspekte der Qualitätssicherung und der Abnahmerelevanz die wesentliche Rolle. Die vorgestellte Schrittfolge gibt gleichzeitig auch die zeitliche Abfolge des Gesamtprozesses wieder. Insofern hat dieser Abschnitt Leitfadencharakter.

#### 7.3.3.1 Lastenheft

Alle Anforderungen zum Mandantenaufbau, wie sie nachfolgend abgearbeitet werden, sollten in einem Lastenheft beschrieben sein. Dieses Lastenheft beinhaltet nicht

nur die technische Abfolge des Vorgehens, sondern ebenso alle fach- und unternehmensspezifischen Informationen für die jeweils konkrete Anforderung. Bei Erstellung und Freigabe sind die üblichen Regeln des Anforderungsmanagements zu beachten.

### 7.3.3.2 Budget

Auf der Basis des Lastenheftes kann eine erste Budgetierung erfolgen. Dabei sind folgende Aufwendungen zu beachten:

- Hardware (Beschaffung, Leasing oder Miete),
- Speichermedien (dito),
- Software-/Schnittstellen-Anpassungen,
- Unterstützung IT-Technik,
- Customizing,
- Unterstützung aus den Fachbereichen,
- Overheads Projektmanagement,
- Dokumentation,
- Schulung,
- Abnahme.

Das Budget kommt häufig aus verschiedenen Töpfen. Ist der Auftraggeber mit dem Fachbereich identisch, trägt dieser die Hauptlast. Die hier aufgeführten Systemkomponenten beziehen sich zunächst auf eine Testumgebung. Das Budget für den laufenden Betrieb wird separat erstellt. Wichtig ist, dass schon zu Anfang Klarheit über das Gesamtbudget für den Mandantenaufbau besteht, damit nicht von vornherein Verzögerungen wegen finanzieller Engpässe und Zuständigkeiten auftreten.

### 7.3.3.3 Einbindung Fachbereiche

Die Bedeutung der Einbindung der betroffenen Fachbereiche wurde bereits bei der Budgetierung (s. o.) erwähnt. Auch inhaltlich ist die Mitwirkung der Fachbereiche unabdingbar. Ihre Beteiligungen sind insbesondere gefordert bei:

- der Lastenhefterstellung,
- der Auswahl von Testdaten,
- der Datenqualitätssicherung,
- dem Customizing,
- der Abnahme,
- der Prozessdokumentation,
- der Schulung.

Die Beteiligung der Fachbereiche ist kritisch. Deshalb ist eine ausreichende und verbindliche Termin- und Aufgabenplanung ganz zu Beginn des Projektes erforderlich. Fachbereichsressourcen müssen freigestellt und dem Projekt zeitweilig zugeordnet werden. Die Fachbereichsverantwortlichen sollten das Projekt formell mit unterstützen und für Eskalationen zur Verfügung stehen.

### 7.3.3.4 Unterstützung durch das Rechenzentrum

Neben der Unterstützung durch Entwickler ist der Support des Rechenzentrums für einen reibungslosen Ablauf notwendig. Die Aufgaben, die nur in Abstimmung mit diesem gelöst werden können, betreffen:

- Bereitstellung eines leeren Test-Mandanten mit dem aktuellen Release inkl. Datenbank und Speicherkapazitäten,
- Beladung des leeren Mandanten und anschließender Dump,
- Auslösung der erforderlichen jobstreams für die Datenmigration,
- Unterstützung bei der Abnahme,
- evtl. Bereitstellung eines Schulungsmandanten,
- Inbetriebnahme des Produktivsystems.

Die zuständigen Mitarbeiter des Betriebs sollten deshalb von Anfang an an der Projektplanung beteiligt werden.

### 7.3.3.5 Projektplanung

Bei der Planung und Durchführung des Mandantenaufbaus sind die Grundsatzregeln des Projektmanagement zu beachten. Da diese bereits an anderer Stelle erläutert werden, soll hier keine Wiederholung stattfinden. Nur soviel:

Zum Kick-off sind alle durch den einvernehmlich benannten Projektleiter für diese Maßnahme eingeladen, die in irgendeiner Weise einen Beitrag zum Gesamtprojekt leisten – einschließlich der Führungsverantwortlichen, die ihre Personalressourcen bereitstellen. Es ist wichtig, dass am Ende dieser Veranstaltung eine verbindliche Zuordnung

#### **Aufgaben – Termine – Personen**

festgehalten wird. Der Terminplan, der in Form einer Checkliste oder eines Meilensteinplans dokumentiert wird, ist Teil der offiziellen Projektdokumentation und dient als Messlatte für alle zukünftigen Reviews.

## 7.3.4 *Standbau*

### 7.3.4.1 Projektverfolgung

Neben den klassischen Werkzeugen des Projektmanagements soll an dieser Stelle noch auf vier Hilfsmittel hingewiesen werden, die das interdisziplinäre Umsetzen erleichtern können:

- regelmäßige Statusbesprechungen,
- dazwischen ad-hoc-Statusabfragen,
- Führen einer Offenen-Punkte-Liste (OP-Liste),
- Restpunkte-Liste.

Task	wer	bis wann	Status

Status: offen, in Arbeit, erledigt

Abb. 7.8 Statusbericht

Statusbesprechungen

Dazu werden alle Beteiligten eingeladen, die einen Beitrag seit dem letzten Status leisten können. Vorgelegt wird zu Anfang die gesamte Aufgaben-Tätigkeiten-Liste mit den Terminstellungen. Später werden nur noch die Abweichungen vom ursprünglichen Plan behandelt. Zu jedem Punkt wird ein Status festgehalten mit den nächsten Handlungsanweisungen (todos). Abbildung 7.8 zeigt so einen Statusbericht.

Dieser Bericht mit Teilnehmerliste, Ort und Zeit sowie vereinbarten nächsten Schritten dient als Protokoll des Arbeitsfortschritts.

Status-Abfragen

Häufig ist es für die Projektleitung notwendig, zwischen den formalen Status-Reviews einen Zwischenstand abzufragen. Bei der Aufgabenstellung Mandantenaufbau ist das dann der Fall, wenn kurzzeitige Schritte wie Datenbeladung, Customizing etc. durchgeführt werden, weil darauf wiederum Folgeaktivitäten aufbauen, die sofort ausgelöst werden müssen. Im Falle eines Fehlschlags oder einer Verzögerung der Vorläuferaktivitäten muss der gesamte Projektplan sofort angepasst werden. Die beteiligten Projektmitarbeiter werden über diese kurzfristigen Status-Änderungen per E-Mail informiert, ohne dass eine gesonderte Besprechung anberaumt werden muss.

OP-Liste

Mit dem Projektfortschritt wandeln sich die formellen Statusberichte zu einer OP-Liste, (s. Abb. 7.9) die meistens im Schnelldurchgang bei den Projektbesprechungen durchgearbeitet wird. In dieser Phase des Projekts sind alle aufeinander eingespielt, und die inhaltliche Materie ist so bekannt, dass lange Erläuterungen entfallen.

Restpunkte-Liste

Am Ende des Projekts, d. h. nach erfolgtem Mandantenaufbau, wird die letzte OP-Liste zur Restpunkte-Liste. Diese Liste beinhaltet offene Aktivitäten, die für die

Offene Punkte	Maßnahme	wer	bis wann

Abb. 7.9 OP-Liste

volle Funktionsfähigkeit des Mandanten zwar wichtig, aber nicht unerlässlich sind. In der Praxis bedeutet das, dass der Mandant mittlerweile abgenommen ist (s. u.), diese Aktivitäten aber trotzdem noch ausstehen. Die Restpunkte-Liste wird somit Teil der Abnahmedokumentation. Die termingerechte Abarbeitung der noch offenen Punkte muss nach der Inbetriebnahme des Mandanten weiterhin überwacht werden. Zu solchen Restpunkten können gehören:

- Schulungen,
- Datenbereinigungen,
- Berechtigungen u. a.

7.3.4.2 Eskalation

Wie in jedem anderen Projekt auch, sind zu Beginn ein oder mehrere Eskalationspfade festzulegen, falls es aus unterschiedlichen Gründen zum Stillstand kommen sollte. Für den Mandantenaufbau können diese Pfade von den nachfolgend aufgeführten Aufgabenstellungen abhängen. Dementsprechend kann es sich um unterschiedliche Adressaten der Eskalation handeln:

- Unklarheiten in der Zuständigkeit der Datenbeschaffung bzw. Dateneinpflege,
- plötzlich aufgetretener Anpassungsbedarf z. B. für Schnittstellen,
- Erstellen von mandantenspezifischen Schulungsunterlagen,
- Zuständigkeiten beim Customizing.

Unbenommen davon sind die üblichen Eskalationsthemen: Terminüberschreitung, Abnahmeverweigerung etc.

7.3.4.3 Customizing

Die gesamte Problematik des Customizing ist weiter oben beschrieben.

7.3.4.4 Schnittstellen

Sind durch den Mandantenaufbau bestimmte Schnittstellen in ihrer Funktionalität betroffen, müssen für die Anpassungen Lasten- und Pflichtenhefte erstellt werden.

Außerdem ist der Vorlauf des Realisierungszeitraums einzuplanen. Die gesonderte Abnahme von Schnittstellen ist weiter oben beschrieben.

In den häufigsten Fällen ist jedoch keine programmtechnische Anpassung notwendig, sondern lediglich ein Customizing der Mapping-Tabellen oder bestimmte Konfigurationseinstellungen. Auch dafür ist die Erstellung eines Lastenheftes sinnvoll.

#### **7.3.4.5 Testmandant**

Technisch wird zunächst ein leerer Mandant unter der vereinbarten Version des relevanten Betriebssystems bereit gestellt. Bei dieser Bereitstellung sind die gleichen Customizing-Regeln zu beachten wie für den Produktivmandanten mit Ausnahme von Systemadressierungen. Für die Projektplanung ist der genaue zeitliche Ablauf der Datenbeladung entscheidend. Festgelegt werden müssen folgende Schritte:

- Entscheidung bzgl. der diversen Quellsysteme, aus denen die Daten bezogen werden sollen,
- Prüfen auf Vorhandensein der erforderlichen Beladungsschnittstellen,
- Zeitpunkt der Urladung,
- Qualitätssicherung der Urladung,
- zusätzliche Migrationen zur Veredlung des ursprünglichen Datenbestandes,
- Qualitätssicherung der Migrationen,
- manuelle oder automatisierte Datenbereinigung,
- Festlegen der Verantwortlichkeiten für die weitere Datenpflege,
- Sicherung des gesamten Testdatenbestandes.

#### **7.3.4.6 Prozessdokumentation**

Die bereits vorhandene Prozessdokumentation ist auf den neuen Mandanten anzupassen. Liegt noch keine Dokumentation vor, ist eine zu erstellen. Hierbei sind die üblichen Ebenen zu berücksichtigen: Übersichtsdokument, Detailbeschreibung und Arbeitsanweisungen. Hier sollen nur die Letzteren berücksichtigt werden. Aus ihnen werden die Schulungsunterlagen erstellt. Die Arbeitsanweisungen beschreiben die Bedienung der Mandanten-Software durch den Enduser.

Darüber hinaus sind Dokumente zu erstellen, die den Betrieb, die Bedienung von Schnittstellen sowie das Auslösen von Berichten beschreiben. Wichtig ist auch der gesamte Prozess des Fehlermanagements: die support levels und Eskalationspfade inkl. Ansprechpartner und Telefonnummern. Die Dokumentation ist Teil des abzunehmenden Gesamtpakets.

#### **7.3.4.7 Berechtigungen**

Bei der Einrichtung von Userberechtigungen sind folgende Regeln zu beachten:

- Die Berechtigten eines neuen Mandanten sind (mit vielleicht der Ausnahme von Administratoren) meistens nicht dieselben Personen, die für bisherige Mandanten eingerichtet waren.
- Die Berechtigten für den Testmandanten sind nicht deckungsgleich mit denen des Produktivmandanten. Technisches Testpersonal, Schulungsverantwortliche etc. werden später keinen Zugriff mehr auf das Produktivsystem erhalten. Umgekehrt müssen die berechtigten Fachbereichsressourcen aus dem Testmandanten durch weitere User ergänzt werden.

#### 7.3.4.8 Schulung

Die Benutzerschulung kann möglicherweise auf dem Testmandanten erfolgen. Dann böte sich an, nach einer Grundschulung die Teilnehmer gleichzeitig am Testgeschehen zu beteiligen. Oder aber die Schulung erfolgt auf einem separaten Schulungsmandanten, der eine Spiegelung des Testmandanten wäre. Das bedeutet ein Zuwarten mit der Schulung, bis der Testmandant abgenommen ist, um das ungewünschte Auftreten von Fehlern während der Schulung zu vermeiden. In der Konsequenz würde sich eine Verzögerung zwischen der Abnahme des Testmandanten (s. u.) und der Produktivsetzung für die Schulungsdauer ergeben.

Schulungsunterlagen und Arbeitsanweisungen für die spätere Systemanwendung können aus der Prozessdokumentation (s. o.) erstellt werden. Schulungsdaten würden sich aus den Beladungen ergeben.

#### 7.3.4.9 Abnahme

Im Prinzip sind bei der Abnahme des neuen Mandanten auf dem Testsystem die gleichen Regeln zu beachten, wie sie ausführlich oben grundsätzlich beschrieben sind. Da es bei der Neueinrichtung von Mandanten Abkürzungen oder Varianten geben kann, sind hierzu besonders folgende Optionen zu prüfen:

- Welche Abnahmekriterien sind Mandanten spezifisch?
- Reicht eine Betriebsbereitschaftserklärung aus?
- Soll ein Probetrieb gefahren werden?

Solche und ähnliche Fragen sind in einem Freigabemeeting zu klären, wobei letztendlich dann auch der Mandant zur Migration auf das Produktivsystem freigegeben werden soll.

#### 7.3.4.10 Migration

Der Aufbau des Testmandanten und die Datenbeladungsstrategie sind so ausgelegt, dass eine mühelose Übernahme auf den Produktivmandanten möglich ist. So sind z. B. die Customizing-Aktivitäten nicht noch einmal zu wiederholen.

Der erste Schritt nach der Abnahme ist also die Sicherung des letzten Stands des Testmandanten: Software, Betriebssystem, Datenbank und alle Utilities. Dieses Paket wird dann auf den zunächst leeren Produktivmandanten aufgespielt. Danach erfolgt die Datenqualitätssicherung, die die Vollständigkeit und inhaltliche Richtigkeit der Datenbeladung prüft, gegebenenfalls bewertet oder Datenbereinigungsmaßnahmen vorschlägt und durchführen lässt. Erst danach kann das System zur produktiven Nutzung freigegeben werden.

### 7.3.4.11 Inbetriebnahme

Nach erfolgter Abnahme (s. o.) und entsprechend formalisierter Dokumentation kann der Mandant in den Regelbetrieb übergeben werden. Dazu ist noch einmal die Vollständigkeit bei der Übergabe festzuhalten. Hier die wichtigsten Elemente:

- Release,
- Datenbank,
- Utilities,
- Dokumentation,
- Schulung,
- Berechtigungen.

## 7.3.5 *Templates*

**Tab. 7.2** Gesamtcheckliste Mandantenaufbau

Aktivität	Bis wann	Status
Lastenheft erstellen und freigeben		
Budget absichern		
Einbindung Fachbereich		
System Support regeln		
Aufgaben und Termine dokumentieren		
Projektstruktur veröffentlichen		
Eskalationspfade festlegen		
Customizing vornehmen		
Schnittstellen ggf. anpassen		
Beladung Testmandant		
Datenmigration		
Prozessdokumentation erstellen		
Berechtigungen freigeben		
Schulung		
Abnahme		
Inbetriebnahme		

### 7.3.6 Lösungsweg

*Technologische Mandantenbasis schaffen > Basiskonfiguration herstellen > Lastenheft mit Meilensteinplanung, Budget, Support und Projektplanung > Tracking > Customizing > Dokumentation > Schulung > Migration > Inbetriebnahme.*

## 7.4 Integrationstests

### 7.4.1 Prozesstests

Alles, was bisher über Schnittstellentests und Simulationen abgehandelt wurde, entsprach der Sicht durch eine rein technologische Brille. Die Frage lautete: funktionieren die Schnittstellen einwandfrei, indem sie die korrekte Anzahl von Datensätzen in einem vorgegebene Format übertragen? Auf diese Weise lassen sich alle Schnittstellen in einem Systemverbund sukzessive testen. Treten keine Fehler auf, liegt technologisch gesehen kein Grund vor, keine Abnahme zu erteilen. Diese Sicht wird häufig von Entwicklern und Software-Lieferanten bevorzugt.

Im täglichen Leben von Anwendungen wird ein solches Funktionieren zwar vorausgesetzt. Im Einsatz sind die Schnittstellen an sich für den Nutzer jedoch transparent. Sein Interesse liegt in einem funktionierenden Prozess – und zwar des Teiles seines Prozesses, für den er in seinem Tagesgeschäft verantwortlich ist. In diesem Sinne interessiert er sich für das *inhaltliche* Ergebnis am Ende seiner Prozesskette. Dem hat eine vertiefende Testmethodologie Rechnung zu tragen

In der Abnahmepraxis bedeutet diese Sicht, dass integrativ getestet werden muss. Für die Testfälle müssen alle für den entsprechenden Teilprozess (bis hinunter zur Arbeitsanweisung) relevanten Systeme mit ihren Schnittstellen untereinander einbezogen werden (s. Abb. 7.10). Erst wenn diese Kette insgesamt positiv durchlaufen ist, kann dieser Prozessteil als abgenommen gelten. Eine Einzelbetrachtung der Schnittstellen geschieht lediglich dann, wenn spezifische Probleme beim Test auftreten.

Der Grund für mögliche Abweichungen bei Schnittstellen-Performance im Prozesstest gegenüber individuellen Schnittstellen-Tests liegt in der Auswahl von Akzeptanzkriterien durch den Fachbereich. Ein reiner Übertrag von sauberen Daten ohne Abweisungen reicht nicht unbedingt aus:

- die Verknüpfungen zwischen Datenfeldern müssen stimmen,
- Plausibilisierungen müssen funktionieren,
- bestimmte Felder sollen unter Umständen ausgeblendet werden usw.

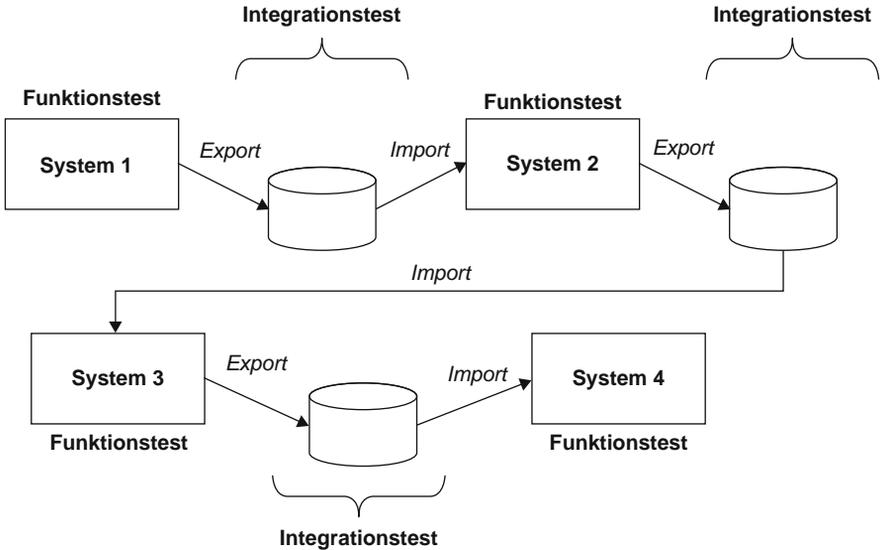


Abb. 7.10 Integrationstests

### 7.4.2 Verfügbarkeit von Partnersystemen

Ein Integrationstest bedingt also das Zusammenspiel aller an einem Prozess beteiligten Systeme. Das bedeutet:

- Vorhandensein von Testsystemen aller beteiligten Partner,
- Verfügbarkeit aller Testsysteme,
- Bereitschaft zur Mitwirkung,
- Einbindung aller in die Abnahmeplanung,
- gemeinsames Fehlertracking,
- allseitiger Systemsupport,
- Unterstützung aller Fachbereiche.

Gleichzeitig ist sicherzustellen, dass auf den Partnersystemen die Releasestände der Produktionsumgebungen vorhanden sind. Um selbst diese Grundvoraussetzungen zu schaffen, ist nicht nur ein erheblicher planerischer Aufwand vonnöten, sondern sind häufig auch politische Grabenkämpfe auszufeuchten. Schließlich geht es um temporäre Abstellungen von z. B. Testressourcen, die dem Vorteil eines anderen Fachbereichs dienen. Hier ist nicht immer mit Verständnis zu rechnen, da oftmals enge Budgets die Mitarbeiter an andere Aufgaben binden. Schwierig ist auch die technische Verfügbarkeit von Systemen, da in anderen Bereichen natürlich auch Systemänderungen oder Tests laufen. Aus diesem Grunde ist die Organisation von Integrationstests vorteilhaft auf einer höheren Ebene nahe bei der Unternehmensleitung anzusiedeln.

### 7.4.3 Integriertes Testvorgehen

Um die oben genannten Voraussetzungen in den Griff zu bekommen, sind plane-rische Maßnahmen erforderlich, die über Abnahmen auf einer einzelnen Systemum-ggebung hinausgehen. Dazu gehören:

- frühzeitige Identifizierung von Testern aller betroffenen Bereiche,
- gemeinsames Kick-off mit den Partnersystemen,
- allseits akzeptierte Testskripte,
- zeitliche Taktung der erforderlichen Schnittstellenläufe,
- gemeinsames Fehlertool,
- Vergabe und Kommunikation der Kommunikationsadressen (System),
- Freischalung der erforderlichen Zugriffe auf Fremdsysteme,
- Bereitstellung von Austauschverzeichnissen.

Die Abb. 7.11 zeigt die integrative Systematik schematisch.

Ansonsten gelten sinngemäß alle Kriterien, die auch für eine Individualabnahme bzw. stand alone Schnittstellen angeführt wurden. Bezüglich des endgültigen Abnahmedokumentes sollten die Unterschriften von den Verantwortlichen aller Partnersysteme eingeholt werden.

### 7.4.4 Integrierte Testskripte

Eine wesentliche Voraussetzung, die bereits erwähnt wurde, ist die Erstellung integrierter Testskripte. In der Regel werden solche von Testern des eigentlich betroffenen System entwickelt und den Partnersystemen zum Abgleich und zur Abstimmung vorgelegt. Die Abb. 7.12 zeigt schematisch solch ein Testskript.

Wichtig bei der Abarbeitung eines solchen Skripts ist die Einhaltung der Tak-tung, um Ressourcen (Tester und Systeme) punktgenau bereithalten zu können. Hier gibt es allerdings Grenzen der Planbarkeit. Sollten beim Durchlauf in irgendeinem System oder an irgendeiner Schnittstelle Probleme auftreten, tritt eine ungeplante Verzögerung ein. Die Ursache kann mannigfach sein:

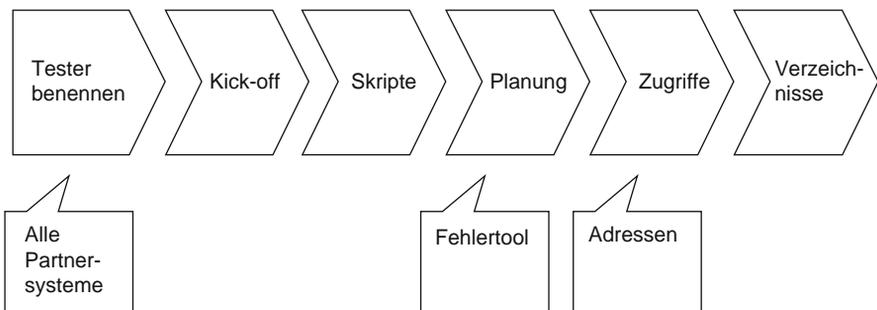


Abb. 7.11 Planung Integrationstests

Nr.	System	Testfall-Beschreibung	Variante	Ergebnis
1	System1	Funktionstest1	X1	
2	System1 System2	Funktionstest2, Export, Import, Funktionstest3	X1	
3	System2	Funktionstest4	X1	

**Abb. 7.12** Integriertes Testskript

- Konfigurationsfehler,
- Customizingproblem,
- Schnittstellenfehler,
- Softwarefehler in System A,
- Softwarefehler in System B.

Manchmal reichen aber auch Verständigungsprobleme zwischen den Ausführenden, um den Lauf zum Stocken zu bringen. Entsprechend gestalten sich die Verzögerung und deren Auflösung. In jedem Fall bedingt eine Verzögerung eine Neuplanung im Nahbereich mit all den Schwierigkeiten der Bereithaltung von Ressourcen.

Ziel der integrierten Abnahme ist nicht die Abnahme eines Systems, sondern die Abnahme eines kompletten Prozesses. Aus diesem Grunde sind auch die Unterschriften aller Beteiligten aus den betroffenen Fachbereichen auf dem Abnahmeprotokoll erforderlich.

### 7.4.5 *Templates*

**Tab. 7.3** Integriertes Testskript

	System
Teilprozess	
Subprozess	
Arbeitsanweisung	
Lfd. Nr.	
Variante	
Input	
Output	
Akzeptanzkriterien	
—	

## 7.4.6 Lösungsweg

*Identifikation der relevanten Prozessstrecke > organisatorische Einbindung aller Partnersysteme > Klärung der erforderlichen Rahmenbedingungen > Gesamtplanung für den Integrationstest > Erstellen und Abarbeiten integrativer Testskripte.*

## 7.5 Performance

### 7.5.1 Performance als Akzeptanzkriterium

Performance als Abnahmegegenstand birgt eine eigene Problematik in sich. Zum einen lässt sich als Prüfkriterium kein eigenes Testskript im üblichen Sinne erstellen. Andererseits kann Performance jedoch so kritisch für die Abarbeitung von Prozessschritten sein, dass Fachbereich und Enduser einen Anspruch auf einen minimalen Standard haben. Wenn für die Ausführung bestimmter Funktionen unakzeptable Zeiträume benötigt werden, wird das Geschäft empfindlich gestört, wenn nicht gar unmöglich gemacht. Deshalb kann Performance tatsächlich zu einem Akzeptanzkriterium werden. Dabei erhebt es den gleichen Anspruch bei der endgültigen Beurteilung von einzuführender Software bezogen auf die Abnahme wie klassische Testfälle.

### 7.5.2 Benchmarks

Das eigentliche Problem ist natürlich nicht das Akzeptanzkriterium an sich, sondern die Messlatte. Das subjektive Empfinden eines vor einem stummen Bildschirm sitzenden Nutzers, der auf die Rückkehr seines Cursors wartet – also das reine Antwortzeitverhalten – ist meistens nicht ausreichend, um als messbare Größe zu gelten. In der Regel werden deshalb auch keine online-Funktionen oder GUIs herangezogen, sondern batch-Läufe, deren Exekutionsdauer sich quantifizieren lässt. Die Bestimmung von CPU time (Rechenzeit) oder elapsed time (Laufzeit) ist nicht schwierig, da diese meistens vom System ausgewiesen werden. Schwierig bleibt jedoch deren Interpretation.

Im Idealfall bildet das Testsystem nicht nur die komplette Produktionsumgebung ab, sondern simuliert ebenfalls alle konkurrierenden Prozesse, die normalerweise sonst noch auf den Maschinen laufen – inklusive der wechselnden Userlast während des Tages oder der Nacht. Eine solche Konstellation ist unter wirtschaftlichen Gesichtspunkten kaum herstellbar. Außerdem bestünden immer noch Zweifel an der Aussagekraft, wenn die Lastverteilung Diskussionsgegenstand würde.

Deshalb begnügt man sich normalerweise mit der Herstellung einer Konfiguration, die folgendes ermöglicht:

- Repräsentatives Datenvolumen, das extrapolierbar ist,

- Leistungsauslegung des Systems in Zentraleinheit und Speicher, die eine Skalierbarkeit ermöglichen.

Über einen abgestimmten Algorithmus werden nach den Testläufen sodann die gemessenen Zeiten entsprechend hochgerechnet.

### **7.5.3 Performanceläufe**

Liegen die Kriterien und der Algorithmus fest, sollten innerhalb des Abnahmezyklus Performanceläufe eingeplant werden. Das kann entweder parallel zu dem übrigen Testgeschehen erfolgen oder aber während besonders dafür freigehaltenen Zeitfenstern. Mitunter kann der Test einer batch-Funktionalität neben der inhaltlichen Fehlerprüfung gleichzeitig auch zur Performance-Analyse genutzt werden.

### **7.5.4 Monitoring**

Um dynamische Performancedaten zu ermitteln, gibt es auf dem Markt bzw. von den Herstellern der Betriebssysteme entsprechende Monitore, die diese Daten sammeln und visualisiert am Bildschirm ausgeben. In Abhängigkeit des zu untersuchenden Problems und der Anwendungsumgebung laufen solche Monitore Stunden, Tage oder gar Wochen mit. Die Informationen, die dabei ausgegeben werden, beinhalten unter anderem:

- die Programme, die gerade genutzt werden,
- Prozessor-Auslastung,
- Hauptspeicherbelegung,
- Speichermanagement,
- Anzahl laufender Prozesse,
- Häufigkeit von Datenzugriffen (Tabelle öffnen),
- Overheads und Interrupts,
- Warteschlangen,
- Füllgrad der Systemtabellen,
- I/Os,
- Swap-Rate.

Neben den statischen Systeminformationen und Konfigurationsparametern sind die folgenden Berichte aus dem dynamischen Bereich von Interesse:

- durchschnittliche Prozessorleistung,
- zeitabhängige Prozessorleistung,
- durchschnittliche Speicherauslastung,
- zeitabhängige Speicherauslastung,
- offene Dateien,

- Laufwerksbelegung,
- Kommunikationsverkehr.

Die folgenden Abb. 7.13 bis 7.16 enthalten typische Darstellungen von Performance-Messungen.

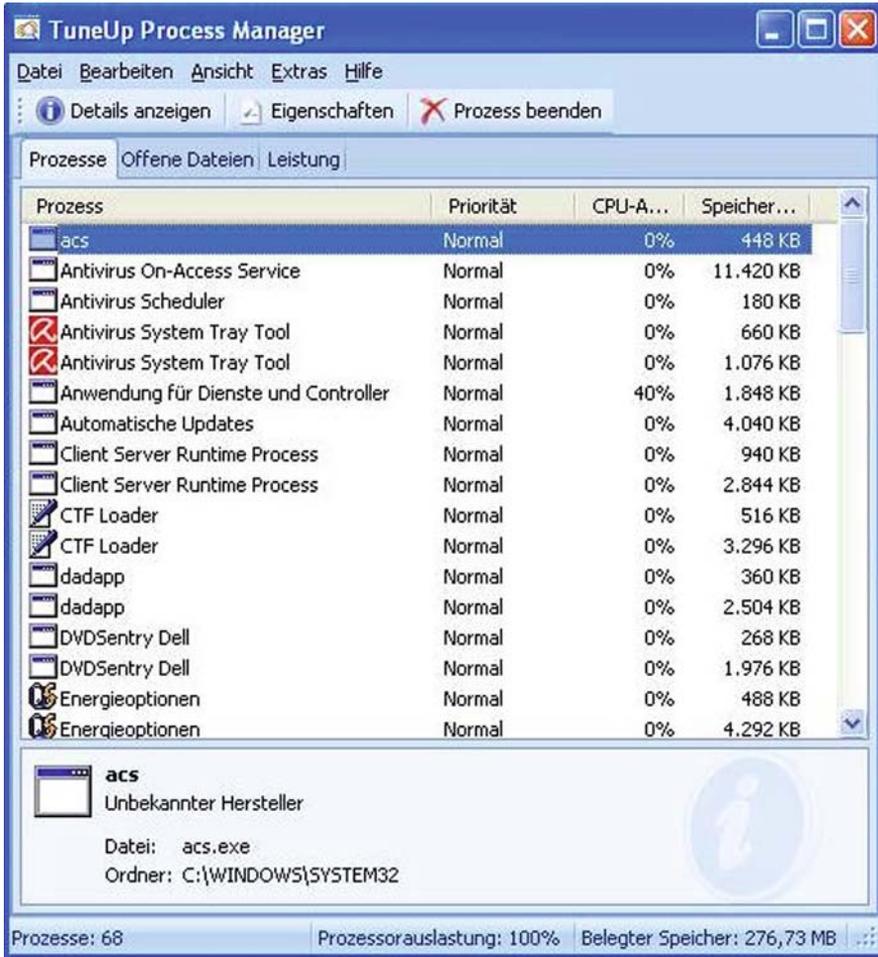


Abb. 7.13 Prozessorleistung insgesamt

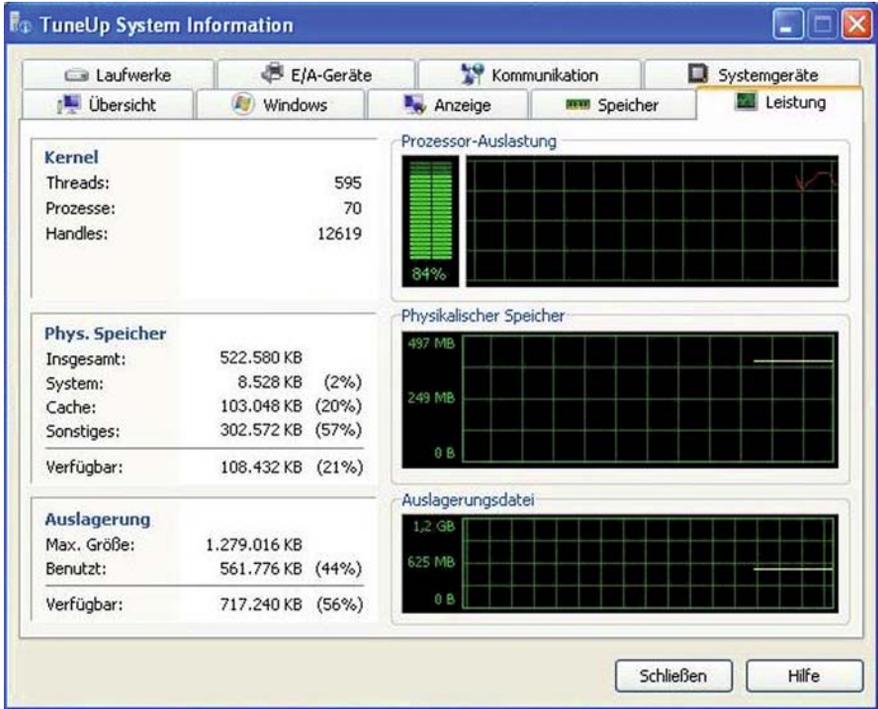


Abb. 7.14 Prozessorleistung differenziert

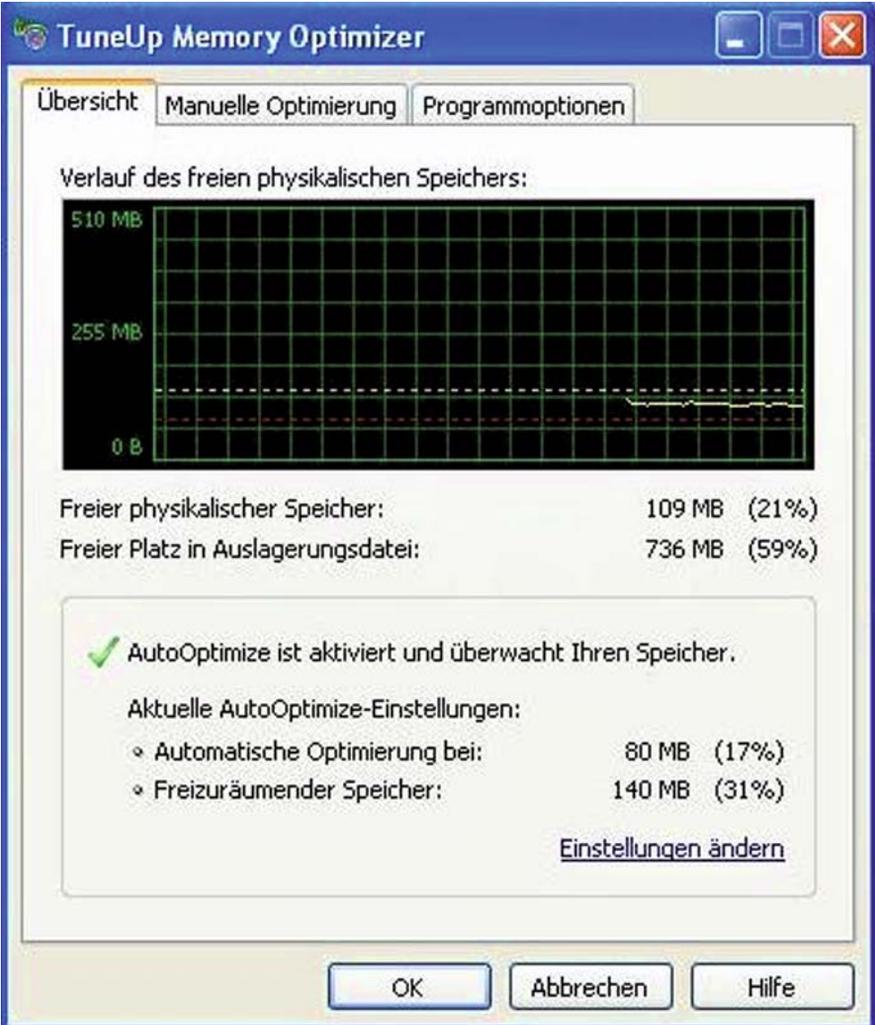


Abb. 7.15 Memory Optimizer

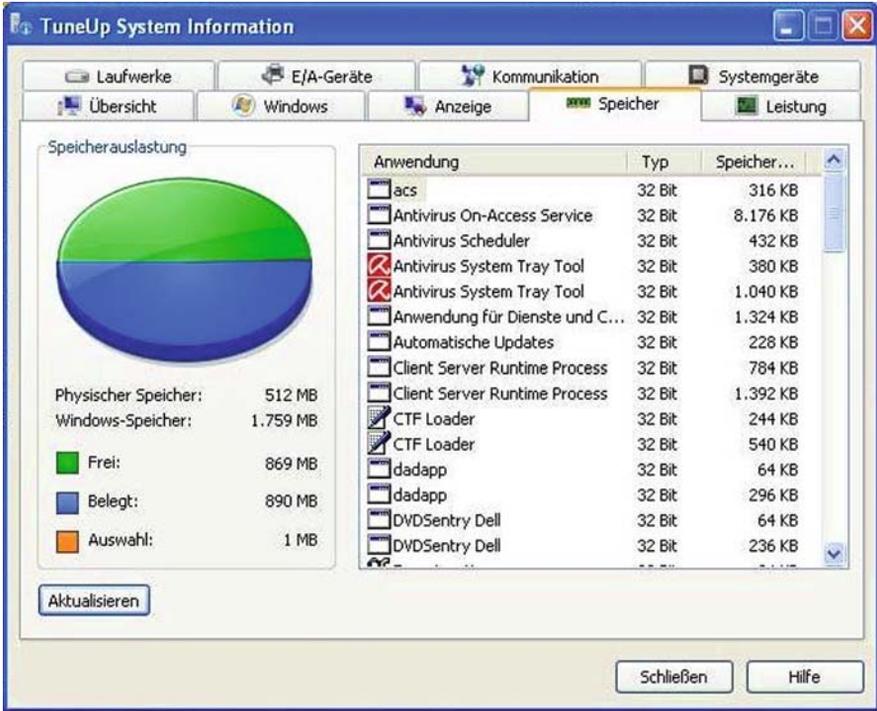


Abb. 7.16 Durchschnittliche Speicherauslastung

### 7.5.5 Templates

Tab. 7.4 Checkliste Performance Optimierung

#### Aktivitäten

erledigt

- Identifizierung des sichtbaren Performance-Problems
- Aufnahme der Systemumgebung
- Aufnahme der Anwendungen
- Aufnahme Datenhaltungssystem
- Aufnahme Produktionsplan
- Aufnahme User-Umfeld
- Detailermittlung Systemparameter
- Auswahl und Installation eines geeigneten Monitors
- Messung des Leistungsverhaltens (Prozessor, Speicher, Kommunikation)
- Identifikation von Spitzen
- Ermittlung der Durchschnittswerte
- Datenanalyse
- Identifikation des Engpasses
- Erstellen eines Maßnahmenkatalogs
- Umsetzung (kurzfristig, mittelfristig, langfristig)
- Regressionsanalyse nach Umsetzung
- Feintuningmaßnahmen
- Regelbeobachtung

### **7.5.6 Lösungsweg**

*Identifikation der Performance-Relevanz für bestimmte Funktionalitäten > Festlegung des Benchmarks > Ermitteln der Methode (Skalierung, Simulation Echtbetrieb etc.) > zeitliche Einplanung > Einsatz eines Monitors > Analyse der Messergebnisse.*

# Kapitel 8

## Dokumentation

### 8.1 Dokumentationsarten

Zu unterscheiden sind Dokumente, die vor, während und nach dem Projekt erstellt und benötigt werden. Dazu gehören sogenannte Referenzdokumente. Diese werden vor und während einer Abnahme benötigt, um ein eindeutiges Abnahmeverständnis zwischen Kunde und Lieferant herbeizuführen. Es handelt sich dabei um eine Mischung aus technischer und kommerzieller Dokumentation, die teilweise über das Vertragswerk, das der Einführung zu Grunde liegt, referenziert wird. In der nachfolgenden Tab. 8.1 befinden sich einige beispielhafte Einträge (s. Felderläuterungen).

Die Felder im Einzelnen:

- Lfd. Nr.: sequentiell
- Dateiname: inkl. Pfad
- Autor: verantwortlicher Verfasser (nicht derjenige User, der die letzte Version gespeichert hat)
- Dokumenttitel: ausgeschriebener Titel, z. B.
  - Abnahmerichtlinie
  - Pflichtenheft ....
  - Testskript für ....
  - Lastenheft für ....
  - Testplan ....etc.
- Version: Nr. der freigegebenen Version (Versionierung nach DIN/ISO)
- Datum: Freigabedatum

Daneben gibt es Dokumente, die während einer Abnahme erstellt werden und Ergebnisse festhalten. Beispielhaft seien genannt:

**Tab. 8.1** Referenzdokumente

Lfd. Nr.	Dateiname	Autor	Titel	Version	Datum
----------	-----------	-------	-------	---------	-------

- Bereitstellungsprotokoll
- Problemspeicher
- Abnahmeprotokoll ....

Und schließlich ist da der gesamt Block der Systemdokumentation. Die mit der Software zu liefernde Dokumentation ist umfänglich im Einführungsvertrag festgeschrieben. Sie kann auf unterschiedlichen Medien ausgeliefert werden:

- Papier (eher selten),
- auf Datenträger,
- elektronisch mit der Software (online),
- elektronisch per Link,
- oder auf mehr als einem von diesen Medien.

Zu dieser Dokumentation gehören normalerweise:

- Leistungsbeschreibung:
  - Darstellung der Geschäftsprozesse, die durch den Softwarebetrieb unterstützt werden,
- Produktionsplan:
  - zeitliche Abfolge von jobs mit Parametern und Konfigurationsbedingungen, die für die Exekution erforderlich sind,
- Bestandsdokument:
  - bzgl. Softkomponenten, Datenbanksystem, jobs, Dienstprogramme etc.,
- Inventardokument:
  - nur bei gleichzeitiger Bereitstellung von Hardware durch den Lieferanten,
- Betriebskonzept:
  - zusätzlich zum Produktionsplan: Datensicherungsstrategie, Logging, Support etc.,
- Benutzerdokumentation:
  - Arbeitsanweisungen, nach Fachgebiet des Sachbearbeiters strukturiert, meistens als online-Hilfe,
- Customizingdokumentation:
  - systemische und fachliche Voreinstellungen und default-Belegungen von Datenfeldern.

Lastenhefte und Pflichtenhefte sollten den letzten Freigabestatus vor Abnahme enthalten.

Wesentlich ist, dass im Vorfeld zwischen allen Beteiligten Einigkeit darüber herrscht, welche Dokumente Abnahmerelevanz haben.

## 8.2 Vollständigkeit und Relevanz

Wesentliche Hinweise zu den benötigten Quellen und Dokumenten sind bereits im Kap. 5 über die Abnahmerichtlinie gegeben worden.

### 8.2.1 *Vollständigkeit*

Es gibt Dokumente, die auf jeden Fall vor Abnahmebeginn vorliegen müssen, und solche, die auch nachgereicht werden können. Zu den ersten gehören unbedingt:

- freigegebene Lastenhefte,
- freigegebene Pflichtenhefte,
- Testskripte,
- Bereitstellungsprotokoll.

Zu den letzteren können z. B. gehören:

- Schulungsunterlagen,
- Prozessbeschreibungen u. a.

Natürlich ist es wünschenswert, dass all diese Dokumente vor Abnahmebeginn vorliegen. Manchmal ist das aus Zeitgründen jedoch nicht möglich. Selbstverständlich können Dokumente, die erst während oder nach der Abnahme entstehen (Problemspeicher, Abnahmeprotokoll etc.) nicht vorher geliefert werden. Die gesamte Software-Dokumentation wird ohnehin erst frühestens nach der Bereitstellung der Software ausgeliefert.

### 8.2.2 *Relevanz*

Zu unterscheiden sind technische gegenüber juristisch relevanten Dokumenten. Zu den juristisch und damit Rechnungslegungs-relevanten Dokumenten gehört sicherlich die Abnahmeerklärung mit ihren Anhängen (Abnahmeprotokoll, Korrekturplan). Alle anderen Unterlagen beinhalten technische Details, die eine Abnahme und später einen Betrieb erst ermöglichen. Ihre Bedeutung zeigt sich allerdings darin, dass z. B. die gesamte Systemdokumentation ein eigener Abnahmegegenstand ist. Dazu weiter unten mehr.

**Tab. 8.2** Dokumente und Verantwortlichkeiten

Lastenhefte	Fachbereiche
Pflichtenhefte	Realisierer
Arbeitsunterlagen zur Abnahme	Abnahmekoordination
Testskripte	Fachbereiche
Systemdokumentation	Lieferant

### 8.3 Verantwortlichkeiten

Je nach Dokumenttyp sind unterschiedliche Verantwortlichkeiten festzulegen (s. dazu Tab. 8.2).

Die Abnahmekoordination hat dafür zu sorgen, dass diese Dokumente rechtzeitig vorliegen und nach Versionierung freigegeben worden sind.

### 8.4 Versionierung und Freigaben

Bei der Erstellung eines Dokumentes, das keinen operativen Charakter hat (wie z. B. Problemspeicher oder Abnahmeerklärung), sondern Inhalte beschreibt, sollte eine Versionierung eingeführt werden. Die zu verwendende Nummernsystematik könnte wie folgt aussehen:

$x.y$ , mit

- $x$ : Versionsnr. und
- $y$ : Hinweis auf den Freigabestatus

Der Freigabestatus kann folgendermaßen kodiert werden:

- 0: Entwurf
- 1: in Abstimmung
- 2: freigegeben.

Zusätzliche Feinabstufungen sind möglich.

Während des Abstimmungsprozesses werden Kommentare und Ergänzungen von verschiedenen Beteiligten eingeholt, die in Folgeversionen einfließen. Diese gesamte Historie wird in der Regel am Schluss eines Dokumentes in einer Tabelle festgehalten (s. Templates), sodass der Entwicklungsprozess einer Dokumentation auch später noch nachvollzogen werden kann. Es kann schon einmal Kontroversen über den letzten gültigen Stand geben, wenn bei den Freigaben und der Verteilung Fehler unterlaufen sind. Die Historie ist ein sicheres Instrument, solche Klärfälle zu lösen.

## 8.5 Templates

Folgende Templates sind nützlich: Freigabehistorie (Tab. 8.3), Liste der Arbeitstabelle (Tab. 8.4).

**Tab. 8.3** Freigabehistorie

Version	Status	Datum	Verantwortlich	Hinweis
0.1	Zwischenstand			Ersterstellung
0.2	Zwischenstand			Überarbeitung
0.3	Zwischenstand			Überarbeitung
0.4	Zwischenstand			Überarbeitung nach Hinweisen durch ...
1.0	Freigabe		s. o.	

**Tab. 8.4** Liste der Arbeitstabellen

Vollständigkeit der Arbeitstabellen

Freigabehistorien

Freigabe Abnahmerichtlinie

Budgetplan

Zuständigkeiten

Tasklisten (diverse)

Problemspeicher

Ideenspeicher

Bereitstellungsprotokoll

Ablaufplan

Checklisten (diverse)

Abnahmegegenstände

Abnahmeobjekte

Referenzdokumente

# Kapitel 9

## Sicherheitsaspekte

### 9.1 IT-Sicherheit als Teil des Qualitätsmanagements

IT-Sicherheit spielt auch bei Einführungsprojekten eine wichtige Rolle. Sie kann als eigenständiges Gebiet eingeordnet werden oder im Zusammenhang mit dem übergeordneten Thema IT-Qualitätsmanagement gesehen werden. Selbst, wenn man es eigenständig neben IT-Qualitätssicherung stellt, sind die gegenseitigen Verflechtungen und Abhängigkeiten so stark, dass eine Betrachtung des einen ohne Rückgriff auf das andere nicht möglich ist. IT-Qualitätsmanagement ist Voraussetzung für eine saubere Umsetzung der Sicherheitsaspekte. Ohne Berücksichtigung der Sicherheitsaspekte gibt es keine Nachhaltigkeit im Qualitätswesen.

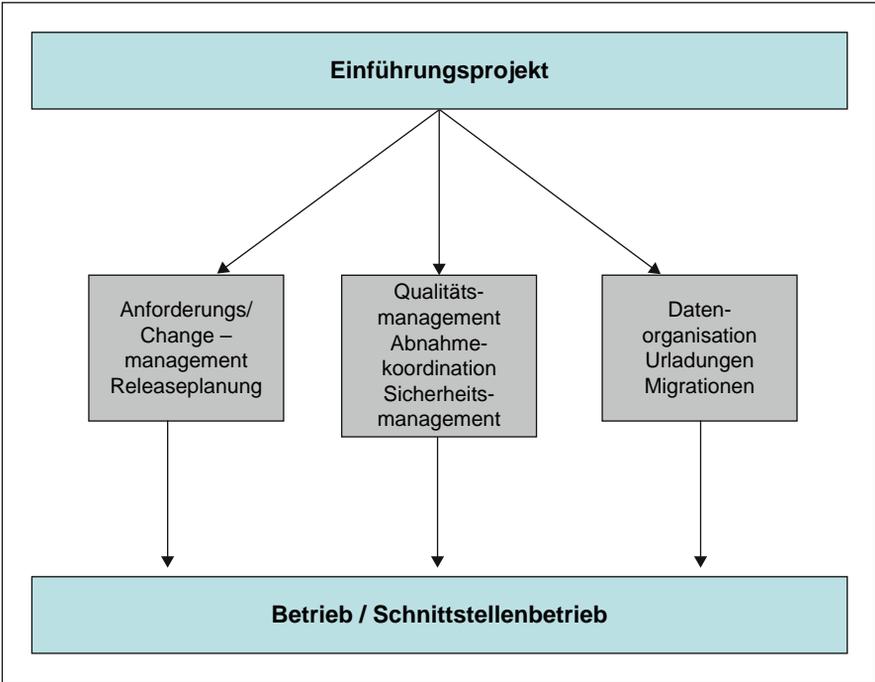
Im Folgenden wird deshalb IT-Sicherheit in einen Zusammenhang mit IT-Qualitätsmanagement gestellt. Es wird zunächst der Gültigkeitsbereich für die Forderungen der IT-Sicherheit im Einführungsprojekt definiert. Der Themenkomplex IT-Sicherheit unterliegt – wie andere Managementbereiche auch – gewissen Normen, auf die bei seiner Umsetzung zurückgegriffen werden kann. Diese werden kurz vorgestellt. Die Abb. 9.1 zeigt die Einordnung von IT-Qualitätsmanagement und IT-Sicherheit in den Rahmen des Gesamtprojekts.

Die Erfüllung der Sicherheitsanforderungen ist abhängig von der strategischen Einordnung dieser Aufgabenstellung insgesamt im Unternehmen. Es wird dargelegt, welche Voraussetzungen dafür geschaffen werden müssen. Dazu gehört auch eine effiziente Einbindung der Projektmitarbeiter.

#### 9.1.1 Gültigkeitsbereich

Der Gültigkeitsbereich des Managementsystems IT-Sicherheit wird durch zwei Begrenzungen charakterisiert:

- organisatorisch,
- zeitlich.



**Abb. 9.1** Einordnung Sicherheitsmanagement

Die organisatorische Gültigkeit bezieht sich auf die Organisationseinheiten im Unternehmen, die von diesem System und der zugehörigen Dokumentation betroffen sind. In der Regel sind das alle. Bei Ausnahmen kann es sich z. B. um ausgelagerte Einheiten, Tochterunternehmen oder Beteiligungsgesellschaften handeln. In Übergangszeiten nach Fusionen bestehen ebenfalls die Möglichkeiten, dass bestimmte Abteilungen, die mit separater Systemstützung gefahren werden, anders gesteuert werden. In den einschlägigen Dokumenten sind diese Gültigkeitsbereiche festzuschreiben.

Bei zeitlich begrenzter Gültigkeit handelt es sich normalerweise um Versionsstände. Jedes Dokument hat eine Versionsnummer, die auf das Hauptdokument verweist. Die Gültigkeitsaussage bezieht sich dann auf die aktuelle Version, in Ausnahmefällen auf Abschnitte von Vorgängerversionen. Auf jeden Fall gilt das letzte Update. Hierzu gehören auch Aussagen, wie mit Änderungen zu den einzelnen Dokumenten zu verfahren ist. Die Änderungen sind in einer Versionshistorie bis zur endgültigen Freigabe festzuhalten.

## 9.2 Normative Verweisungen

Der gesamte Themenkomplex der IT-Sicherheit ist Gegenstand von nationalen Normen und Richtlinien, die an dieser Stelle kurz vorgestellt werden sollen. Detaillierte Informationen sind den Originaldokumenten zu entnehmen.

### **9.2.1 Gesetzliche Vorschriften**

Zunächst ein Hinweis auf Gesetzeswerke, die unterschiedliche Aspekte der IT-Sicherheit berühren. Dazu gehören:

- Bundesdatenschutzgesetz (BDSG),
- Informations- und Kommunikationsdienstegesetz (IuKDG),
- Fernmeldeverkehr-Überwachungs-Verordnung (FÜV),
- Signaturgesetz (SigG),
- Signaturverordnung (SigV),
- Teledienstgesetz (TDK),
- Teledienste-Datenschutzverordnung (TDSV),
- Telekommunikationsgesetz (TKG).

### **9.2.2 Richtlinien und Normen**

Das Bundesamt für Sicherheit in der Informationstechnik (BSI) hat Richtlinien entwickelt, die auf internationalen Normen beruhen. Der wichtigste Standard für Managementsysteme für Informationssicherheit (ISMS) findet sich unter BSI-Standard 100-1. Dem stehen ergänzend gegenüber die sogenannten IT-Grundschutzkataloge der BSI, in denen die unterschiedlichen Sicherheitsanforderungen fachlich ausgebreitet werden. Die Standardreihe des BSI 100-1 bis 100-3 liefert Lösungsanleitungen zur Umsetzung der im Folgenden kurz beschriebenen allgemeinen Normen und bietet auf diesem Wege eine Grundlage zur Zertifizierung z. B. nach ISO 27001.

#### **9.2.2.1 Standard ISO/IEC 13335**

Dieser Standard wie auch die folgenden wurde in Zusammenarbeit mit der International Electrotechnical Commission in Genf entwickelt. Es handelt sich bei diesem Dokument um eine Darstellung allgemeiner Grundsätze, die die Referenzbasis für weitere spezifische Standards darstellen. Dieser Standard beinhaltet im Wesentlichen:

- Konzepte und Modelle für die Sicherheit in der Informations- und Kommunikationstechnologie,
- Technologische Voraussetzungen für das Management von Sicherheitsrisiken,
- Richtlinien für die Netzwerksicherheit.

#### **9.2.2.2 Standard ISO/IEC 17799**

Dieser Standard unterbreitet Lösungswege und Schrittfolgen für die strategische Einführung von IT-Sicherheitssystemen. Detaillierte technische Anleitungen sind

nicht Gegenstand dieses Dokuments. Es hat definitiv Vorschlagscharakter ohne Verbindlichkeit.

### 9.2.2.3 Standard 27001

Der Standard lautet: „Information Technology – Security Techniques – Information Security Management Systems Requirements Specifications“. Auch dieser Standard hat nur Empfehlungscharakter. Technische Anleitungen zur Umsetzung werden nicht gegeben.

## 9.3 Strategische Voraussetzungen

War bis jetzt die Rede von der Einbindung der IT-Sicherheit in Projekte, so folgt nun deren hierarchische Einordnung in einem Unternehmen. Die Sicherheitsaspekte aus dem Projekt heraus lassen sich nicht unabhängig von den übergeordneten Sicherheitsregularien, die sonst im Unternehmen zur Anwendung kommen, betrachten.

IT-Sicherheitsmanagement ist ein Teil des übergeordneten Sicherheitsmanagements, welches das gesamte Unternehmen umfasst – also auch alle anderen materiellen und immateriellen Güter sowie alle Mitarbeiter. Insofern sollte das IT-Sicherheitsmanagement methodisch und prozessual in diese übergeordneten Aspekte eingebunden sein. Wenn von IT-Sicherheitsmanagement die Rede ist, ist damit selbstverständlich der gesamte Komplex der IT- und Kommunikationssicherheit gemeint.

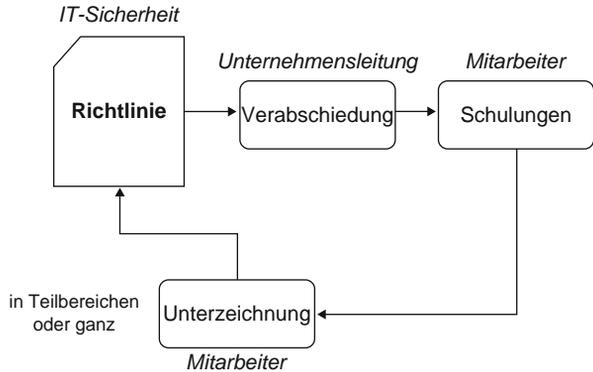
Die IT-Sicherheit wird gewährleistet durch eine Reihe von konzeptionellen und organisatorischen Maßnahmen sowie die notwendigen technischen Voraussetzungen, die erforderlich sind, die Sicherheitsziele zu erreichen. Betroffen sind:

- IT-Prozesse,
- Computersysteme,
- Hardware,
- Software,
- Kommunikationseinrichtungen,
- Daten,
- Dokumentation.

Um die technischen Voraussetzungen für die IT-Sicherheit zu schaffen, spricht man auch von Security and Safety Engineering. Die Voraussetzungen dafür leiten sich von unternehmensspezifischen Sicherheitskriterien ab, die von der Hierarchie unter Beratung der Sicherheitsfachleute vorgegeben werden. Zu den klassischen Kriterien wie

- Datenintegrität,
- Vertraulichkeit u. a.

**Abb. 9.2** Umsetzung der Sicherheitsrichtlinien



können noch weitere wie zum Beispiel Verfügbarkeit und Authentizität hinzukommen. Verfügt ein Unternehmen z.B. über drahtlose Kommunikationsnetze, sehen die Kriterien anders aus als bei reinen LAN-Anwendungen. Dem zugrunde liegt auf oberster Ebene eine vereinbarte und kommunizierte Sicherheitspolitik. Die Sicherheitspolitik sollte als Teil der Unternehmensleitsätze verankert und mit entsprechenden Zuständigkeiten in der Unternehmensleitung versehen sein.

Aus diesen grundsätzlichen Festlegungen folgen auf der Durchführungsebene in hierarchisch strukturierter Form Dokumente, die diese Leitsätze in Richtlinien umsetzen und mit Leben erfüllen.

Richtlinien sind allgemeiner Art oder beziehen sich auf spezifische Technologiefelder. Sie gliedern sich in die eigentliche Richtlinie und dann in entsprechende Durchführungsbestimmungen (s. Abb. 9.2).

In der folgenden Abb. 9.3 ist noch einmal die hierarchische Einbettung der IT-Sicherheitsorganisation beispielhaft dargestellt.



**Abb. 9.3** IT-Sicherheitsorganisation

## 9.4 Einbindung der Mitarbeiterschaft

Grundsätzlich sind alle Mitarbeiter und damit auch die Projektbeteiligten über die geltenden Sicherheitsrichtlinien im Unternehmen zu unterrichten. Das kann geschehen bei der Vergabe eines Accounts durch Zusendung entsprechender Dokumente. In spezifischen Fällen, z. B. bei der Nutzung von WLAN, sollte eine Schulung erfolgen. Eine Schulung für Administratoren sollte immer obligatorisch sein, da diese auf sensible Unternehmensdaten und Konfigurationen zugreifen. Die Sicherheitsaspekte, die bei Administratoren eine Rolle spielen, gehen naturgemäß über diejenigen für eine größere Allgemeinheit hinaus.

Nach erfolgter Schulung bzw. Kenntnisnahme der entsprechenden Sicherheitsdokumentation sollte jeder Mitarbeiter durch seine Unterschrift auf einem eigens dafür vorgesehenen Formblatt bestätigen, dass er informiert worden ist, und dass er mit der Richtlinie einverstanden ist und diese respektieren wird. Die unterschriebenen Bestätigungen sind durch die IT-Sicherheitsorganisation zu archivieren.

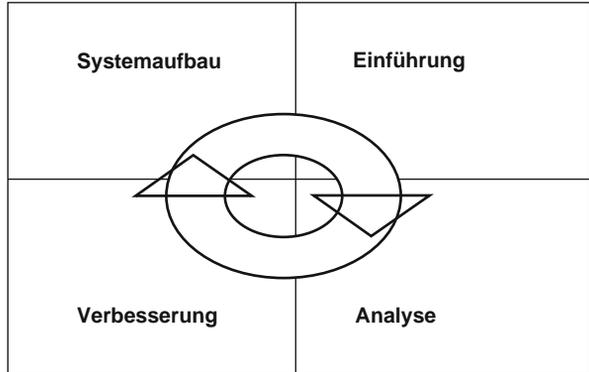
Die folgende Tab. 9.1 fasst noch einmal zusammen, welche strategischen Voraussetzungen für den Aufbau eines IT-Sicherheitsmanagements innerhalb des Einführungsprojektes erforderlich sind.

### 9.4.1 Nachhaltigkeit

Ähnlich wie bei anderen Aspekten des Qualitätsmanagements auch, spielt bei der Überprüfung, Einhaltung und Fortentwicklung der nach dem amerikanischen

**Tab. 9.1** Checkliste IT-Sicherheit

Existiert ein IT-Sicherheitsmanagement relevant für das Einführungsprojekt?	Das IT-Sicherheitsmanagement befasst sich mit allen Sicherheitsaspekten beim Aufbau und dem Betrieb von IT-Installationen.
Sind die Belange des IT-Sicherheitsmanagements dokumentiert?	Voraussetzung für ein wirkungsvolles IT-Sicherheitsmanagement ist eine entsprechende Dokumentation.
Werden im Zuge des Sicherheitsmanagements die gängigen IT-Normen berücksichtigt?	ISO/IEC 13335, 17799, 27001 BSI 100-1 bis -3
Sind die IT-Sicherheitskriterien dokumentiert?	Sicherheit wird nach Kriterien wie Vertraulichkeit, Verfügbarkeit, Integrität u. a. spezifiziert.
Erfolgt nach IT-Sicherheitsschulungen eine unterschriebene Erklärung der Beteiligten?	Die Teilnahme an einer Sicherheitsschulung sollte im Interesse aller Beteiligten dokumentiert werden.
Wird die Einhaltung der Sicherheitsvorschriften regelmäßig kontrolliert?	Zur Kontrolle der Einhaltung der Sicherheitsrichtlinien sollte ein Maßnahmenplan erstellt werden.

**Abb. 9.4** Deming Prozess

Qualitätsguru Deming benannte Prozess auch in der IT-Sicherheitsphilosophie eine wichtige Rolle. Die Abb. 9.4 zeigt diesen Prozess schematisch.

Es geht immer wieder um den gleichen Zyklus:

*Systemaufbau > Einführung > Analyse > Verbesserung.*

Das, was oben angeregt wurde, muss organisatorisch und technisch vorbereitet werden. Dann erfolgt die Einführung unter Einbeziehung aller Beteiligten. Nach einer gewissen Laufzeit werden Erfahrungen gesammelt, die dann schließlich wieder in neue Vorschläge und Verbesserungen in das System einfließen. Und der Prozess beginnt von vorn. Dabei ist zu bedenken, dass die Laufzeitphase nicht mit einer Erprobungsphase gleich zu setzen ist. Vielmehr handelt es sich um einen kontinuierlichen Prozess mit festgelegten Review-Intervallen. Auch geht es nicht ausschließlich um Verbesserungen aufgrund von anfänglichen Design-Fehlern. Vielmehr soll durch den Gesamtprozess sichergestellt werden, dass gerade im IT-Bereich Schritt gehalten wird mit den neuesten technologischen Entwicklungen in Bezug auf Sicherheitsaspekte.

## 9.5 Templates

### Sicherheitsrichtlinie (Struktur)

- Gegenstand der Richtlinie (z. B. Hardware: Laptop; Software: Intranet ....)
- Beantragungsverfahren zur Nutzung
- Verantwortlichkeit für die Vergabe von Rechten
- Begrenzung der Nutzung und Kosten
- Verbote
- Haftung
- Schäden
- Vertraulichkeit

## **Zuwiderhandlungen**

- Anzeige an IT-Sicherheitsbeauftragten

## **Bestätigung der Belehrung**

- Diese Richtlinie sollte Teil der umfassenden Sicherheitsbelehrung von Mitarbeitern sein. Im Anschluss kann dann folgende Vereinbarung unterzeichnet werden:....
- Kenntnisnahme der Richtlinie.
- Bitte, lesen Sie die vorliegende Richtlinie und zeichnen Sie unten auf dem Dokument gegen. Eine Kopie mit Ihrer Unterschrift geht an den IT-Sicherheitsbeauftragten.
- Ihre Unterschrift bestätigt:
  1. Ich habe die Richtlinie erhalten, verstanden und stimme ihr zu.
  2. Bestätigung im Einzelnen der Vorgaben aus der Richtlinie.
  3. Bestätigung der Vertraulichkeitsklausel.
  4. Bestätigung der Haftung und Verantwortlichkeit.
  5. Ich habe verstanden, dass Zuwiderhandlungen zur Richtlinie rechtliche Konsequenzen haben können.
    - Name des Beschäftigten,
    - Unterschrift des Beschäftigten,
    - Abteilung,
    - Datum.

## **9.6 Lösungsweg**

Definition von IT-Sicherheit als Teilprojekt > Integration in den Gesamtzusammenhang der IT-Sicherheitsstrategie > Sicherstellung der strategischen Rahmenbedingungen > Überprüfung der Einhaltung der Vorschriften > kontinuierliche Verbesserung.

# Kapitel 10

## Projektmanagement

### 10.1 Technische und organisatorische Hilfsmittel

Die bisher behandelten Methoden des Projektmanagement bei der Einführung von komplexen Softwaresystemen haben sich auf die erforderlichen organisatorische Maßnahmen und deren Dynamik während des Projektfortschritts gestützt. Dieses sind die Grundlagen für erfolgreiches Projektmanagement überhaupt. Ein rein systemtechnischer Ansatz wird seine Ziele nicht erreichen, wenn diese Grundlagen vorher nicht sauber herausgearbeitet worden sind. Dennoch ist eine systemtechnische Flankierung häufig sinnvoll, um die gesamte Komplexität von solchen Projekten besser zu beherrschen. Manchmal ist sie unabdingbar. Die Systeme, die das Projektmanagement unterstützen, laufen unter dem Kürzel PMS: Projektmanagementsystem. Auf diese soll in diesem Kapitel näher eingegangen werden.

Folgende Aspekte werden dabei vertieft:

- Zielsetzung und Aufgaben eines PMS,
- technische Möglichkeiten und funktionale Ausprägungen,
- die wichtigsten Grundfunktionalitäten,
- spezifische Funktionalitäten im Einzelnen, als da sind:
  - Vorgangsplanung,
  - Meilensteinplan,
  - Kapazitätsmanagement,
  - Gantt-Diagramm,
  - Netzplantechnik,
  - kritische Pfade,
  - Kostenüberwachung.

## 10.2 Zielsetzung und Aufgaben

PMS sollen Projekte ab einer gewissen Komplexität so unterstützen, dass eine erleichterte Abwicklung möglich wird. Dabei soll Transparenz entstehen über:

- den Projektfortschritt,
- zeitliche Engpässe,
- personelle Engpässen,
- und die Kosten.

All das dient dazu, letztendlich die Projektziele zu erreichen. Der Aufwand zur Pflege der PMS-Daten sollte dabei in einem wirtschaftlich vernünftigen Verhältnis zum sonstigen Projektaufwand stehen. Es gibt kleinere Projekte, bei denen man auf aufwendige Datenerfassung und -aufbereitung verzichten kann, oder wo ein EXCEL-Sheet alle notwendigen Anforderungen abdeckt. Auf der anderen Seite gibt es aber auch hochkomplexe Projekte – und über die reden wir in unserem Zusammenhang –, die eine dementsprechende Systemstützung erfordern. Zur Bedienung einer solchen PMS-Anwendung ist dann ein entsprechender Mitarbeiterstab erforderlich. Je nach Komplexitätsgrad kommen also PMS mit unterschiedlichen Mächtigkeiten zum Einsatz.

Neben der planerischen Aufgabe soll ein PMS auch in der Lage sein, rechtzeitig im Projektverlauf Auskunft darüber zu geben, ob es Störungen, Zeitüberschreitungen, Kapazitätsengpässe oder Kostenüberschreitungen gibt. Es soll also auch als Steuerungsinstrument dienen.

Um all diese Anforderungen zu erfüllen, sind neben den zu ermittelnden Basisdaten, die den Projektplänen zugrunde liegen, auch laufend Statusdaten aus dem Projektverlauf selbst zu erfassen, die zum Abgleich der ursprünglichen Planung mit dem tatsächlichen Geschehen herangezogen werden müssen. Dieser Aufwand ist nicht zu unterschätzen. Für PMS gilt wie für alle anderen betriebswirtschaftlichen Systeme auch: die Qualität der Systemstützung hängt von der Datenqualität, des Inputs seiner Bediener ab. Hat man sich einmal für den Einsatz eines PMS entschieden, sollte die Datenpflege auch ernsthaft betrieben werden. Halbherzigkeit führt zu unnötigem Aufwand und dabei noch zu falscher Projektdokumentation.

Für das Berichtswesen an die Projektleitung bieten PMS normalerweise eine ganze Anzahl von Möglichkeiten, die direkt in Berichte übernommen werden können, ohne dass aufwendige Neuverfassungen erforderlich sind.

## 10.3 Möglichkeiten und Umfang

Die auf dem Markt befindlichen PMS haben unterschiedliche Ausprägungen an Funktionalität. Dementsprechend bedürfen sie jeweils entsprechender technischer und organisatorischer Voraussetzungen für ihre effiziente Nutzung. Die meisten Systeme laufen auf gängigen PCs und können im Rahmen der Abwicklung von Pro-

jektassistenten im Zusammenhang mit der sonstigen Projektarbeit bedient werden. Eine gewisse Schulung ist dabei erforderlich. Daneben gibt es große Serveranwendungen. Solche PMS bieten nicht notwendigerweise viele zusätzliche Funktionen, sondern zeichnen sich durch erhöhte Leistungsfähigkeit im Datendurchsatz und durch erweiterte Simulationsmöglichkeiten aus. Wegen des hohen Datenvolumens und der durch viele Aktivitäten bedingten Komplexität und Abhängigkeiten werden diese Systeme meistens durch spezielle Teams betreut.

Daneben gilt es, Abgrenzungen zu beachten. Im Bereich der Kapazitätswirtschaft und insbesondere im Kapazitätsabgleich finden sich ähnliche Möglichkeiten wie in gängigen PPS-Systemen. Analog gilt das für die Kostenüberwachung zu bereits in Organisationen vorhandenen Controlling-Instrumenten. Es kommt häufig vor, dass Kostenverfolgung sowohl im PMS als auch in Controlling-Systemen parallel erfolgt. Da es keine standardmäßigen Schnittstellen von PMS zu großen betriebswirtschaftlichen Anwendungen gibt, sollte auf organisatorischem Wege dafür gesorgt werden, dass Doppelarbeit unterbleibt. Einige betriebswirtschaftliche Systeme, wie z. B. SAP mit PM haben allerdings auch PMS-Funktionen integriert.

## 10.4 PMS-Funktionalitäten

Die ganze Palette von angebotenen Möglichkeiten darzustellen, würde an dieser Stelle den Rahmen sprengen. Deshalb sollen im Folgenden nur die wesentlichen PMS-Funktionalitäten vorgestellt werden:

- Verwaltung von Vorgängen,
- Gantt-Diagramm,
- Meilensteinplan,
- Kapazitätsmanagement,
- Netzplan.

### 10.4.1 *Verwaltung von Vorgängen*

Abbildung 10.1 zeigt eine typische Vorgangsliste, wie sie z. B. in Microsoft Project © angelegt wird. Hierin müssen alle für das Projekt relevanten Einzelaktivitäten erfasst werden. Schon an dieser Stelle zeigt sich, dass eine entsprechende Vorstrukturierung des Gesamtprojekts notwendig ist. Die Reihenfolge der Vorgangsbearbeitung wird über die Anfangs- und Enddaten bzw. die Vorgangsdauer festgelegt. Es macht Sinn, schon bei der Erfassung möglichst auch die zeitliche Abfolge zu berücksichtigen. Das hilft später bei den diversen grafischen Darstellungen, um diese übersichtlich zu gestalten.

Auf der linken Seite haben wir eine laufende Nummer, die zunächst vom System vergeben wird. In der nächsten Spalte erscheint dann der Vorgangsname als spre-

	Vorgangsname	Dauer	Anfang	Ende
1	Kick-off	1 Tag	Don 06.12.07	Don 06.12.07
2	Datenerhebung	15 Tage	Die 18.12.07	Die 08.01.08
3	Migrationsvorbereitung	6 Tage	Mon 10.12.07	Mon 17.12.07
4	Systembereitstellung	2 Tage	Don 06.12.07	Fre 07.12.07
5	Migration	3 Tage	Mit 09.01.08	Fre 11.01.08

Abb. 10.1 Vorgangsliste aus MS-Project ©

chende Langtextbezeichnung. Ihm folgt das Startdatum. Bei der Datumsvergabe sorgt ein interner Kalender für Warnungen, wenn ein solches Datum nicht existiert (z. B. 31.11.) oder auf einen Sonn- oder Feiertag fällt. Danach kann dann die Vorgangsdauer eingegeben werden, aufgrund derer das Enddatum berechnet wird. Man kann auch das Enddatum festlegen, und das System rechnet dann die Vorgangsdauer aus. Es lässt sich allerdings nicht über bestimmen, indem z. B. eine Dauer festgelegt wird, und man gleichzeitig versucht, dieser ein unplausibles Enddatum zuzuordnen.

Den Vorgang kann man dann im Detail betrachten (s. Abb. 10.2) und zusätzliche Informationen einpflegen.

Hier soll der Einfachheit halber die Betrachtung der anderen Reiteroptionen zunächst zurückgestellt werden. Diese werden an anderer Stelle erläutert. Im Wesentlichen werden in der Detaildarstellung die Daten aus der Vorgangsliste wiederholt. Zusätzlich erscheint ein Feld mit der prozentualen Fertigstellungsangabe.

Informationen zum Vorgang

Allgemein | Vorgänger | Ressourcen | Spezial | Notizen | Felder (benutzerdef.)

Name:  Dauer:   Geschätzt

% Abgeschlossen:  Priorität:

Termine

Anfang:  Ende:

Vorgangsbalken ausblenden  
 Vorgangsbalkencollup im

Hilfe OK Abbrechen

Abb. 10.2 Einzelvorgang



Abb. 10.3 Gantt-Diagramm

Dieses Feld kann manuell gepflegt werden. Seine Verwendung wird weiter unten erläutert.

### 10.4.2 Gantt-Diagramm

Im Gantt-Diagramm werden übersichtlich wesentliche Projektplanungs- und Fortschrittsdaten auf einen Blick grafisch dargestellt (s. Abb. 10.3):

- der Vorgang selbst mit seiner Bezeichnung,
- die Vorgangsdauer,
- der Beginn des Vorgangs und
- das Ende des Vorgangs.

Gleichzeitig erscheint auf der x-Achse oben ein Kalender, der sich in unterschiedlichen Feinheitsebenen konfigurieren lässt. Bei Bedarf kann man auch eine zweite Kalenderachse einblenden, die eine Verfeinerung der obersten ist. So können oben z. B. Monate und darunter Wochen, oder oben Wochen und darunter Tage dargestellt werden. Unterhalb dieses Kalenders sind die Einzelvorgänge als Zeitbalken abgebildet.

Man kann Vorgänge hierarchisch beordnen, in dem man bestimmte Vorgangsguppen zu sogenannten Sammelvorgängen zusammenfasst wie in Abb. 10.4:

In der Abb. 10.4 gehören zum Sammelvorgang „Datenerhebung“, der als schwarzer Balken erscheint, die Einzelvorgänge „Migrationsvorbereitung“ und „Systembereitstellung“. Der Sammelvorgang umschließt das früheste Startdatum und das späteste Enddatum aller zugeordneten Einzelvorgänge.

An dieser Stelle lässt sich auch die Verwendung des Erfüllungsgrades („% Abgeschlossen“ aus der Vorgangsbearbeitungsmaske) wie in Abb. 10.5 darstellen.



Abb. 10.4 Sammelvorgänge



Abb. 10.5 Erfüllungsgrad

Der Erfüllungsgrad wird als kleinerer schwarzer Balken innerhalb eines Vorgangsbalkens sichtbar, um den Projektfortschritt zu dokumentieren. Lässt man sich auf die Pflege des entsprechenden Feldes ein, ist ein entsprechender Aufwand zu berücksichtigen.

Die sehr vereinfachten Gantt-Darstellungen lassen eine ganze Anzahl von Möglichkeiten aus. Dazu gehören:

- Verknüpfung mit Vorgängervorgang,
- Dokumentation des Vorgangsverantwortlichen,
- Kapazitätsdarstellungen.

Insgesamt aber ist das Gantt-Diagramm die wohl am Häufigsten verwandte Darstellung im Projektmanagement.

### 10.4.3 Meilensteinplan

Der Meilensteinplan in Abb. 10.6 ist eine Art Subset des Gantt-Diagramms.

Auch hier sieht man wieder alle Vorgänge. Meilensteine sind durch schwarze Rauten dargestellt. Ein Meilenstein kann sich am Ende eines Vorgangs befinden. Der Vorgang ist damit abgeschlossen. Er kann aber auch allein stehen für eine Einzelaktivität von kurzer Dauer – z. B. ein Entscheidungsvorgang. Der Meilenstein ist immer an einen konkreten Termin gebunden und meistens als Ziel oder Teilziel vorgegeben. In diesem Falle sollte die Vorgangsterminierung, was die Dauer betrifft, nach rückwärts erfolgen.

Das System schreibt für einen Meilenstein den Termin bzw. den Endtermin daneben. In dieser Abbildung sind auch beispielhaft Verknüpfungen zwischen Vorgänger- und Nachfolgervorgängen dargestellt.

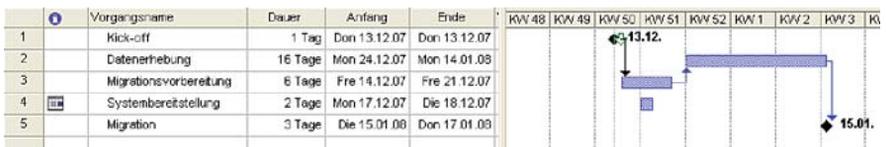


Abb. 10.6 Meilensteinplan



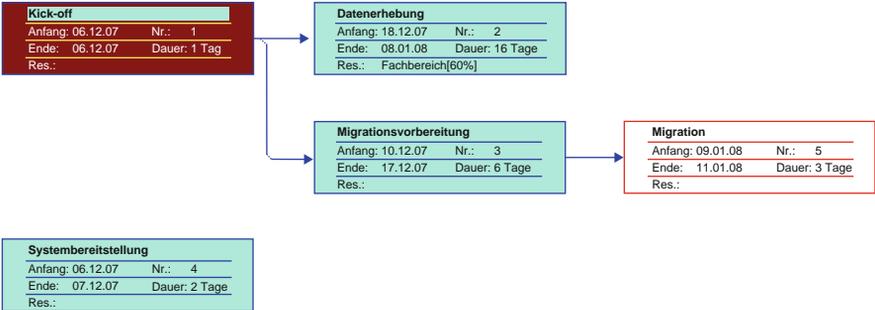


Abb. 10.8 Netzplan

off beginnen. Datenerhebung und Migrationsvorbereitung laufen überlappend. Die Migration selbst kann natürlich erst nach Abschluss der vorbereitenden Arbeiten erfolgen.

#### 10.4.5.1 Kritischer Pfad

Aus dieser einfachen Darstellung wird bereits der kritische Pfad ersichtlich: sind Vorgänge 2 und 3 nicht abgeschlossen, kommt das Projekt entweder zum Stillstand, oder es muss eine Neuterminierung vorgenommen werden. Hier wird auch deutlich, dass für Vorgänge 3 und 5 keine Ressourcen benannt worden sind, und dass für die Datenerhebung der Fachbereich nur zu 60% beteiligt ist. Wer macht den Rest? D. h. der kritische Pfad stellt sich nicht nur als ein zeitliches und logisches, sondern auch als ein Ressourcenproblem dar.

### 10.4.6 Erweiterte Funktionalitäten

Die bisher geschilderten Funktionen sind die wesentlichen Schwerpunkte eines PMS in seinen Grundzügen. Jede einzelne Funktion lässt sich nun anreichern durch zusätzliche Details. Daneben gibt es je nach System weitere Features, die das Projektmanagement erleichtern können. Hier seien beispielhaft nur einige wichtige kurz erwähnt:

#### 10.4.6.1 Vorgangsoptionen

Die Einzelvorgänge lassen sich mit bestimmten Informationen bzw. mit weiteren Steuerungsdaten versehen. Dazu gehören:

- dem Vorgang eine bestimmte Priorität geben,
  - einen Vorgänger definieren, der einen gewissen Zeitabstand zum eigentlichen Vorgang hat, bevor letzterer beginnt,
  - dem Vorgang Einschränkungen auferlegen, z. B.
    - Start so früh wie möglich
    - Start so spät wie möglich
    - Start nicht früher als Datum ttmjjj
    - ....
  - Vorgang unterbrechen,
  - Vorgang als periodisch wiederkehrend definieren;
- außerdem kann jeder Vorgang mit Freitextkommentaren versehen werden.

### 10.4.6.2 Der Basisplan

Beim erstmaligen Anlegen eines Projektes in einem PMS kann diese Version als sogenannter Basisplan definiert werden. Der Basisplan bleibt immer bestehen und wird im Zuge der Projektabwicklung nicht den sich verändernden Umständen angepasst. Ihm gegenüber gibt es dann einen „lebenden“ Projektplan mit zunächst den identischen Plandaten, die sich aber im Laufe des Projektes zeitlich und auch inhaltlich ändern können. Das System liefert nun die Möglichkeit, zu jedem Zeitpunkt einen Abgleich mit dem Basisplan zu fahren, um auf diese Weise die Veränderungen zur ursprünglichen Zielsetzung deutlich zu machen.

### 10.4.6.3 Kapazitätsabgleich

Neben einer einfachen Ressourcenzuordnung, wie oben beschrieben, besteht die Möglichkeit, diese noch einmal separat außerhalb der reinen Vorgangszuordnung zu verwalten. In diesem Zusammenhang können noch folgende Einstellungen vorgenommen werden:

- Verknüpfung einer Ressource mit dem Projektplan (der Name der Ressource erscheint dann in der grafischen Darstellung am Ende des entsprechenden Vorgangsbalkens) – wie oben dargelegt,
- Verknüpfung mit Arbeitszeiten, Schichtkalender etc.,
- Zuordnung von Kostensätzen,
- Zuordnung von Buchungselementen.

Diese zusätzlichen Informationen ermöglichen zweierlei:

- einen realistischen Kapazitätsabgleich SOLL-IST und
- einen vorgangsbezogenen Kostenbericht.

#### 10.4.6.4 Checkliste beim Einsatz eines PMS

Ein PMS kann sehr hilfreich sein, um einer Projektleitung den Rücken von administrativen Aufgaben frei zu halten. Es ist aber eine wirtschaftliche Balance zu halten zwischen diesem Gewinn und dem zusätzlichen Overhead, das durch den Einsatz eines solchen Systems unweigerlich entsteht. Die wichtigsten Fragen dazu können bei der Ausschöpfung bzw. Einschränkung der angebotenen Funktionalitäten und einer ersten Konfiguration eines Projekts durch ein PMS behilflich sein. Sie sind in der Checkliste unter Templates (Tab. 10.1) zusammengefasst.

### 10.5 Templates

**Tab. 10.1** Checkliste PMS

---

In wie viele Einzelvorgänge kann ich mein Projekt zerlegen?
Welche Vorgänge kann man als Sammelvorgänge gruppieren?
Wie viele Projektmitarbeiter sind eingeplant? Rechtfertigt die Anzahl den Einsatz des Kapazitätsmoduls?
Welches wird die Hauptprojektvorlage für die Projektfortschrittsberichte: Meilensteinplan oder Gantt-Diagramm?
Um eine Vorstellung für die Projektkomplexität zu erhalten, sollte das Verhältnis von tatsächlichen Vorgangsverknüpfungen zu allen Einzelvorgängen grob bestimmt werden. Rechtfertigt die Komplexität den Einsatz der Netzplantechnik?
Gibt es bereits ein Projektkosten-Controlling im Unternehmen oder sollen dessen Kosten direkt über das PMS verfolgt werden?
Kann man das Projekt einfach nach vorne oder nach hinten planen, oder sind die Abhängigkeiten so groß, dass zeitliche Einschränkungen im Einzelnen berücksichtigt werden müssen?
Soll mit einem Basisplan gearbeitet werden, weil viele oder starke Abweichungen zu erwarten sind, oder reicht ein stand-alone-Projektplan?
Welche Berichte werden von der Projektleitung erwartet? Davon hängen unter Umständen weitere Detailsinstellungen und Datenerfassungen ab, damit diese Berichte standardmäßig produziert werden können.

---

### 10.6 Lösungsweg

Identifizierung der wesentlichen Kriterien für ein PMS aus dem Projekt heraus > Auswahl eines geeigneten Systems > Schulung der einzusetzenden Mitarbeiter > Auswahl der zu nutzenden Funktionen > Aufbereitung der Basisdaten > Einpflegen der Daten > Aktualisieren mit Projektfortschritt > Steuern nach Projektfortschritt > Aufbau und Nutzung des Berichtswesens.

# Kapitel 11

## Projektcontrolling

### 11.1 Budgetvorgaben

In diesem Abschnitt erfolgt der Brückenschlag zwischen den Budgetvorgaben, die einem Einführungsprojekt gemacht werden, und der Umsetzung dieser Vorgaben innerhalb des Projekts.

In diesem Teil werden insbesondere behandelt:

- Besonderheiten des Projektcontrolling,
- Gegenstände des Controlling,
- Instrumente,
- Steuerungsmöglichkeiten und
- Übergabe an die Linie.

### 11.2 Besonderheiten

Das linienmäßige IT-Controlling läuft schwerpunktmäßig mit den laufenden Ausgaben der im Kostenstellenplan festgehaltenen relevanten Organisationseinheiten. Diese wiederum ventilieren ihre Ausgabenblöcke nach

- laufenden, wiederkehrenden technologischen Kosten (Plattenspeicher, Energie, Peripheriegeräte etc.),
- Personaleinsatz,
- Verbrauchsmaterialien,
- Rechenzentrumsbetrieb,
- Wartung und Pflege,
- Miete und Leasing,
- ASP-Betrieb,
- Kommunikation
- und ähnlichen Kategorien.

Das Controlling gleicht die laufenden Ausgaben mit den am Jahresanfang festgestellten Planwerten ab und steuert eventuell gegen, sollten diese Werte in Gefahr geraten. In der Regel sind Großinvestitionen und Projekte mit zeitlicher Begrenztheit von diesen Verfahren ausgenommen. Dafür werden eigene finanzielle Ressourcen bereit gestellt und gesondert geplant. Diese unterliegen somit auch einem eigenen Controlling, dem IT-Projekt-Controlling.

Beim IT-Projekt-Controlling können insbesondere folgende Gesichtspunkte eine Rolle spielen, die über das klassische Liniencontrolling hinausgehen:

- Aufteilung der Kosten zwischen unterschiedlichen Organisationseinheiten wie
  - IT,
  - Fachbereiche,
  - Projektorganisation selbst,
  - externer Entwicklung,
  - Overheads (Qualitätsmanagement, Abnahmen usw.).
- Dezierte Instrumentarien zur Abwicklung des Controlling, wie
  - Freigabeverfahren,
  - Leistungsnachweise,
  - Projektberichte u. a.
- Anders geartete Steuerungsmechanismen: das IT-Projekt-Controlling selbst kann zu einem eigenen Projektsteuerungsinstrument werden mit allem Einfluss auf den letztendlichen Projekterfolg.
- Und schließlich: die Übergabe aus dem Projekt heraus in die Linie, nach der dann das klassische Liniencontrolling wieder greift.

In den folgenden Abschnitten werden diese Besonderheiten näher erläutert.

## 11.3 Objekte

### 11.3.1 Direkte IT-Kosten

Hiermit sind gemeint:

- Mietkosten für Serverlandschaften,
- Plattenspeicher,
- Peripheriegeräte,
- Betrieb und technische Unterstützung von Testsystemen.

### 11.3.2 Entwicklungskosten

Hierzu gehören:

- Pflichtenhefterstellung,
- internes Qualitätsmanagement,

- Fehlermanagement,
- administrative Overheads (Abnahmen etc.),
- direkte Entwicklungskosten (auf Basis von Tages-/Stundensätzen), intern und extern,
- Dokumentation.

### ***11.3.3 Kosten, die bei den Fachbereichen anfallen***

Hierzu gehören:

- Lastenhefterstellung,
- Abstimmungsoverheads,
- Schulung,
- Abnahmebeteiligung,
- Customizing.

### ***11.3.4 Projektmanagement***

Hierzu gehören:

- Leitung und Administration,
- Support,
- Kommunikation und Projektorganisation,
- Berichtswesen.

All diese oben aufgeführten Kosten fallen sowohl für interne als auch externe Dienstleistungen an. Sie sind – soweit absehbar – vor Projektbeginn zu budgetieren. Außerdem ist ein Budgetposten für Kosten vorzusehen, die nach Aufwand anfallen.

## **11.4 Prozessuale Ansätze**

Prozessuale Ansätze an dieser Stelle bedeuten Maßnahmen zur Steuerung von Ausgaben. Insofern gehen diese über ein reines Berichtswesen mit Soll-Ist-Vergleichen hinaus. Es lassen sich drei Klassen von Steuerungsinstrumenten unterscheiden:

1. Strategische Ansätze
2. Prozessuale Ansätze
3. Berichtswesen

### **11.4.1 Strategische Ansätze**

Hierzu gehören:

- life cycle management,
- Standardisierung,
- Rahmenverträge,
- Anforderungsbudgetierung.

#### **11.4.1.1 Life Cycle Management**

Um die rasanten technologischen Veränderungen in den Griff zu bekommen, ohne in Abhängigkeit von den Wartungs- und Pflegebedingungen der Lieferanten zu gelangen, kann ein life cycle management eingeführt werden. Dieses Instrument verhindert auch ungewollten Wildwuchs in der Systemlandschaft durch unabhgestimmte Beschaffungsaktivitäten anderer Organisationseinheiten.

Beim life cycle management werden unternehmensweit Basiskonfigurationen für z. B. die Endgerätausstattung eines typischen Büroarbeitsplatzes vorgegeben, die beinhalten können:

- PC-Hersteller und Typ,
- Laufwerke,
- USB-Anschlüsse,
- Maus,
- Standardsoftware usw.

Die Gesamtheit aller Arbeitsplätze unterliegt einem Rahmenvertrag mit entweder einem Lieferanten oder einem Dienstleister, der innerhalb vereinbarter Zyklen dafür sorgt, dass die Technologie immer auf dem neusten Stand gehalten wird mit allen erforderlichen Garantie- und Wartungsleistungen. Ähnliche Strategien lassen sich in Großunternehmen für die Serverlandschaft des Netzwerkes fahren. Die Kostenersparnis gegenüber Individuallösungen ist offensichtlich.

#### **11.4.1.2 Standardisierungen**

Neben den Erfordernissen des life cycle managements gibt es Standardisierungserfordernisse im Rahmen von Dienstleistungsbestellungen für Beratungs- und Entwicklungsleistungen. Ein großes Unternehmen kann im Zuge seiner allgemeinen Geschäftsbedingungen über den Einkauf klassifizierte Tages- bzw. Stundensätze durchsetzen, auf die sich ein Anbieter einlassen muss. Gleiches gilt für den Einkauf von Rechenzentrumskapazitäten.

### 11.4.1.3 Rahmenverträge und Abrufe

Bei der Umsetzung standardisierter Vorgaben sollten mit den Lieferanten Rahmenverträge über ein Gesamtpaket von Dienstleistungen abgeschlossen werden, ohne eine Abnahmegarantie zu präjudizieren. Die eigentliche Beauftragung kann in Abhängigkeit vom tatsächlichen Projektaufwand mit einem vereinbarten Vorlauf abgerufen werden, sodass für ein Großprojekt die Vorhaltung von Ressourcenkapazität flexibel gehandhabt werden kann.

### 11.4.1.4 Anforderungsbudgetierung

Bereits in einem sehr frühen Stadium der Projektplanung sollte über das Anforderungsmanagement ein erster grober Kostenrahmen erstellt werden, der eine eigene Identifikation erhält. Gegen dieses Kostenelement kann dann im weiteren Projektverlauf abgerechnet werden. Im Laufe des Projektes können dazu möglicherweise Nachträge eingestellt werden.

## 11.4.2 Prozessuale Ansätze

Hierzu gehören:

- Kontierungselemente,
- Verrechnungssätze,
- das Bestellwertverfahren,
- das Freigabeverfahren.

### 11.4.2.1 Kontierungselemente

Unter den strategischen Ansätzen wurde bereits auf die Notwendigkeit eines frühzeitig festzulegenden Projektkostenrahmens hingewiesen. Dieser erste Kostenrahmen hat zunächst globalen Charakter. Innerhalb des Gesamtrahmens müssen alle Objekte des IT-Controllings untergebracht werden (s. o.: Entwicklung, Fachbereich, Overheads etc.). Um das zu gewährleisten, bietet sich eine hierarchische Gliederung an, wie sie im nachfolgenden Beispiel in Abb. 11.1 dargestellt ist.

Eine solche Struktur erlaubt eindeutige Zuordnungen der Kostenarten und gleichzeitig die Verdichtung angefallener Kosten bis auf höchste Aggregationsstufen, die für das Management Reporting unerlässlich ist.

<b>Abb. 11.1</b> PSP (Projektkostenplanelement)-Hierarchie	PSP	3102	Erweiterung Lagerrechner
		3102 – 1000	Hardware
		3102 – 1000 – 4108	Plattenspeicher
		3102 – 2000	Softwareanpassung
		3102 – 2000 – 4322	Suchfunktion
		3102 – 2000 – 4323	Listenbild
		3102 – 3000	Prozeßberatung
		3102 – 3000 – 4530	Projektleitung
	....		

### 11.4.2.2 Verrechnungssätze

Zuzüglich zu den vertraglich festgelegten Stunden- bzw. Tagessätzen mit externen Dienstleistern existieren in vielen Unternehmen interne Verrechnungssätze zwischen den Organisationseinheiten, die berücksichtigt werden müssen. Bei IT-Dienstleistungen ist das leicht nachvollziehbar. Werden Fachbereichsressourcen einbezogen, müssen diese Leistungen häufig verhandelt oder separat budgetiert werden. Dazu gehören z. B.

- Unterstützung bei Schulungsmaßnahmen,
- Beteiligung bei der Erstellung von Lastenheften,
- Beteiligung bei Abnahmen etc.

### 11.4.2.3 Bestellwertverfahren

Um frühzeitige Budgetüberschreitungen beim Einkauf externer Dienstleistungen, die möglicherweise nicht durch Rahmenvereinbarungen abgedeckt sind, einzugrenzen, lassen sich in den meisten Einkaufssystemen Obergrenzen für den Wert einer einzelnen Bestellung einstellen. Diese Restriktionen können auf bestimmte Hierarchiestufen beschränkt werden. So erfolgt schon bei der Einstellung einer Bedarfsanforderung durch den Verursacher ein entsprechender Hinweis, wenn eine Wertüberschreitung vorgenommen werden soll.

### 11.4.2.4 Freigabeverfahren

Die Freigaben von Budgets oder Teilbudgets (s. Kontierungselemente) erfolgt über eine entsprechende Strategie, in welche Anforderer, Projektleitung und Controlling eingebunden sind. Marktgängige Systeme geben die erforderliche Unterstützung dafür her. In der Regel stellt der Anforderer seinen geplanten Bedarf unter Bezug auf die Kontierungsstruktur im System ein, der Projektverantwortliche prüft, und das Controlling gibt frei. Solche Abhängigkeitspfade lassen sich in den meisten Systemen konfigurieren. Wichtig ist dabei, dass eine automatische Vorprüfung von Budgetgrenzen, Kostenarten und Zeitvorgaben stattfindet, bevor durch den manuellen Eingriff die Freigaben erfolgen.

### 11.4.3 Berichtswesen

Das Berichtswesen für ein IT-Projekt kann sehr umfangreich sein. Entsprechende Systemunterstützung wird von vielen PMS (s. Kap. 10) geliefert, sodass an dieser Stelle eine vertiefte Betrachtung nicht angestellt werden soll. Lediglich auf einige wenige Elemente soll hingewiesen werden:

- Leistungsnachweise,
- Projektberichte,
- Budgetberichte.

#### 11.4.3.1 Leistungsnachweise

Ein typisches Template für Leistungsnachweise zeigt die Abb. 11.2.

Neben den eigentlichen Zeitnachweisen mit den zugehörigen Tätigkeiten sind besonders wichtig das Kontierungselement und evtl. die Bestell-Nr. bei externen Lieferanten, um eine eindeutige Zuordnung zum Budget zu gewährleisten.

Leistungsnachweis		Projekt-Nr.	Monat	Name
Datum	Stunden			
1				
2				
3				
4				
5				
6				
7				
8				
9				
10				
11				
12				
13				
14				
15				
16				
17				
18				
19				
20				
21				
22				
23				
24				
25				
26				
27				
28				
29				
30				
31				
<b>Summe</b>				<b>Unterschrift</b>

Abb. 11.2 Leistungsnachweise

### 11.4.3.2 Projektberichte

Projektberichte werden in unterschiedlichen Formaten geliefert, ein universeller Standard ist nicht vorhanden. Meistens genügt ein Auszug aus dem verwendeten PMS nicht. Der Bericht sollte enthalten:

- inhaltlicher Fortschritt,
- Risiken,
- zeitliche Erfüllung,
- Kostenstand.

Nur das letztere ist hier von Belang. Es ist darauf zu achten, dass der Kosten relevante Teil des Projektberichtes möglichst so formatiert ist, dass eine einfache Übernahme in den Budgetbericht erfolgen kann. Durch entsprechende Verdichtungen kann daraus unmittelbar ein Vorstands- bzw. Geschäftsführungsbericht entstehen.

### 11.4.4 Steuerung

Das Thema Projektsteuerung von komplexen Einführungsprojekten selbst wird an anderer Stelle ausführlich behandelt. Deshalb soll hier lediglich auf die finanziellen Aspekte hingewiesen werden. Dabei sind insbesondere folgende Gesichtspunkte zu berücksichtigen:

- Projektfortschritt und -freigaben,
- Risiken,
- Vorteils-/Nachteilsabwägungen.

Der abwicklungstechnische Vorgang von Freigaben ist bereits weiter oben beschrieben worden. Daneben bleibt festzuhalten, dass die dazu notwendige Systemstützung keinen Automatismus präjudizieren soll, der ein Abstimmungsverfahren aller Beteiligten ersetzt. Der Projektfortschritt ist neben den technischen Erfordernissen natürlich abhängig von den noch verfügbaren finanziellen Ressourcen. Das stufige Freigabeverfahren dient dazu, overruns zu vermeiden. Dabei kann es zu Interessenkonflikten und Neupriorisierungen kommen, wenn mehrere Ressourcen auf den gleichen Geldtopf zugreifen. Hier zeigt sich die Wichtigkeit einer eindeutigen Budgetstrukturierung.

Schon lange im voraus zu einer akuten Freigabeentscheidung sind durch Plan-Ist-Vergleiche die finanziellen Risiken für den Projektfortschritt ersichtlich. Es ist Aufgabe des Controllers, hier vorausschauend und rechtzeitig einzugreifen und auf sich abzeichnende Engpässe hinzuweisen. Dann bleibt noch Zeit, neue Mittel zu beschaffen, bzw. eine Umwidmung vorzunehmen. Vor diesem Hintergrund besteht auch die Möglichkeit einer vernünftigen Bewertung des gesamten finanziellen Projektrisikos. Erfolgt die Warnung zu spät, muss unter Entscheidungsdruck gehandelt werden, sodass häufig eine rationale Abwägung zu kurz kommt.

Auch eine vorzeitige Beendigung eines Projektes ist ein Ergebnis, wenn auch kein gewünschtes. In den Vorteils-/Nachteilsabwägungen müssen neben den noch ausstehenden Kosten und den unternehmerischen Risiken durch Nichterreichung von Vorgaben insbesondere auch die Rückabwicklungskosten berücksichtigt werden. Dazu gehören:

- Verpflichtungen aus bestehenden Beschäftigungsverträgen für externe Dienstleister,
- Rückbau und Neuverwendung gekaufter Hardware,
- Abschreibung nicht verwendeter Investitionen,
- fallback auf vorherige Systemzustände inkl. Schnittstellen und angepasster Partnersysteme,
- Wiederherstellung vorheriger Datenbestände und re-Aktualisierung aus „verlorenen“ Transaktionen,
- evtl. Datenbereinigungsmaßnahmen,
- workarounds für Funktionalitäten, die den Fachbereichen versprochen wurden, jetzt aber nicht mehr geliefert werden.

All diese Gesichtspunkte machen es häufig schwierig, einen Projektabbruch aus rein finanziellen Erwägungen heraus zu rechtfertigen. Das ist mit ein Grund, warum so viele IT-Projekte trotz massiver Kostenüberschreitungen dennoch weitergeführt werden.

### ***11.4.5 Übergabe an die Linie***

Nach Projektende, dessen Definition in der ursprünglichen Projektbeschreibung bzw. -vereinbarung festgelegt wurde (und dem meistens ein Abnahmeprozess vorgeschaltet ist), erfolgt die Übergabe der Leistung in die Linienverantwortung. Damit fallen auch bestimmte Kostenzuordnungen an. Von den eingangs aufgelisteten Linienkosten sind das insbesondere Wartung und Pflege.

Gegen Ende der Projektphase werden während des eigentlichen Projektgeschäfts noch Nachbesserungen im Rahmen der ursprünglichen Budgetierung geleistet, solange eine Endabnahme noch nicht erfolgt ist. Danach müssten die üblichen Pflegeaufwände des Dienstleisters – intern oder extern – greifen. Diese sind meistens vertraglich geregelt. Dieser Gefahrenübergang in eine andere Verantwortung ist allerdings nicht immer so klar abgrenzbar, wie man sich das wünscht.

Auch nach einer Abnahme des Projektgegenstandes können Mängel weiter bestehen, die im Zuge einer Abnahmeerteilung als Auflagen aus der Projektphase in eine Betriebsphase hinüberreichen. Die Nachbesserungskosten liegen natürlich in beiden Fällen beim Dienstleister. Er hat jetzt allerdings das Problem, zu entscheiden, gegen welchen Budgettopf bei ihm diese Nachbesserungen intern abgerechnet werden. Manchmal lassen sich Fehler, die erst in der Nachbesserungsphase, z. T. als Folgefehler der Nachbesserung selber, auftreten, nicht eindeutig zwischen Projekt und Linienbetrieb zuordnen. Aus diesem Grunde ist es wichtig, schon bei der ursprünglichen Liefervereinbarung solche Eventualitäten vertraglich eindeutig festzulegen.

## 11.5 Templates

Folgendes Template ist nützlich: Checkliste IT-Projekt-Controlling (Tab. 11.1)

**Tab. 11.1** Checkliste IT-Projekt-Controlling

Kostenart	budgetiert j/n	PSP-Element	extern(e)/ intern(i)	Dienstleister
<i>Direkte IT-Kosten</i>				
Miete/Leasing Server				
Plattenspeicher				
Peripheriegeräte				
Betriebskosten				
<i>Entwicklungskosten</i>				
Systemanalyse				
Qualitätsmanagement				
Fehlermanagement				
Abnahmen				
Software				
Dokumentation				
<i>Fachbereich</i>				
Systemanalyse				
Schulung				
Abnahme				
Koordination				
Customizing				
<i>Projektmanagement</i>				
Leitung				
Administration				
Berichtswesen				

## 11.6 Lösungsweg

Frühzeitige Bereitstellung des Gesamtbudgets > Definition der Kontierungselemente > Strukturierung des Budgets > Entscheidung über Steuerungsinstrumente > Budgetverfolgung über Rückmeldungen > rechtzeitige Eskalation bei Überschreitungen > Konsolidierung nach Projektende.

## Quellenangaben

Teile des Buches sind in Beiträgen des Autors zum Fachbuch „IT-Management“ beim WEKA-Fachverlag in Kissing erschienen.

Die Abbildungen zur Performance-Messung in Kapitel 7, Teil 7.5 sind screen shots des Performance-Tools „TuneUP Utilities 2008“ der Fa. TuneUp Software GmbH.

Die Abbildungen zum PMS in Kapitel 10 sind screen shots von MSProject © der Fa. Microsoft.

# Sachverzeichnis

## A

Abarbeitungsgrad, 45  
Abnahme, 9, 11, 14, 20, 30, 55, 96  
Abnahmebeginn, 23, 32  
Abnahmebereitschaft, 58  
Abnahmeerklärung, 82, 111  
Abnahmefortschritt, 30  
Abnahmegegenstand, 22, 62  
Abnahmekoordination, 7, 23, 56  
Abnahmeleitung, 44  
Abnahmeort, 63  
Abnahmeplan, 26  
Abnahmeprotokoll, 27, 33, 111  
Abnahmeprozess, 22  
Abnahmereview, 29  
Abnahmerichtlinie, 46, 55  
Abnahmespezifikation, 22  
Abnahmestatus, 39  
Abnahmesystem, 63  
Abnahmeteam, 59  
Abnahmetests, 22  
Abnahmeumgebung, 33  
Abnahmeverfahren, 14, 28  
Abschlussbewertung, 22, 27  
Abschlussreview, 30  
Abweisung, 71  
Abweisungsquote, 74  
Administration, 50  
Akzeptanzkriterium, 35, 102  
Altbestand, 73  
Altdaten, 71  
Altsystem, 21, 71  
Anforderungen, 8, 9, 13  
Anforderungsmanagement, 6–9, 11, 55, 91  
Antwortzeitverhalten, 102  
Anwenderfunktion, 64  
Arbeitsanweisung, 25, 35, 61, 65, 98  
Arbeitszeit, 131  
Auflage, 141

Auftraggeber, 44  
Ausschreibungsverfahren, 10  
Austauschverzeichnis, 100

## B

Basiskonfiguration, 136  
Basisplan, 131  
Bearbeitungsstatus, 14  
Bedienungsfehler, 11  
Beladung, 71, 89, 90, 92  
Benutzerdokumentation, 110  
Berechtigungen, 95  
Berechtigungskonzept, 34  
Bereitstellung, 26, 58, 81  
Bereitstellungsprotokoll, 28, 40, 59  
Bereitstellungstermin, 28  
Berichtswesen, 15, 19, 124  
Bestandsdokument, 110  
Bestellung, 27  
Bestellwertverfahren, 138  
Betrieb, 7, 21  
Betriebsbereitschaftserklärung, 96  
Betriebsbereitschaftsprüfung, 7, 33, 41  
Betriebskonzept, 34, 110  
BSI, 2, 117  
Buchungselement, 131  
Buchungsschlüssel, 86  
Budget, 15, 91  
Budgetplanung, 51  
Budgetüberschreitung, 138  
Bundesdatenschutzgesetz, 117

## C

CAD, 4  
Call Center, 11  
Change Request, 11, 13, 22  
Change-Management, 6, 7, 10, 55  
CMMI, 3  
Controlling, 15, 125

CRM, 4  
 Customizing, 58, 63, 74, 75, 84, 90, 94,  
 101, 110

**D**

Datenbank, 71, 90  
 Datenbeladung, 85  
 Datenbereinigung, 7, 13, 15, 71, 75,  
 95, 97  
 Daten-Controlling, 73  
 Datenfehler, 13  
 Dateninkonsistenz, 21  
 Datenkonsistenz, 73  
 Datenmigration, 22  
 Datenmodell, 86  
 Datenmodell, 21, 71  
 Datenqualitätssicherung, 91, 97  
 Datenquelle, 82  
 Datenstruktur, 74  
 Deming, 121  
 DIN 69 900, 129  
 Dokument, 109  
 Dokumentation, 7, 43, 58, 95  
 Dump, 71, 73, 88

**E**

Echtdaten, 20  
 Einführung, 19  
 Einführungsprojekt, 6, 58  
 Einkauf, 10  
 Einzelaktivität, 125  
 Einzelanforderung, 9, 11  
 Empfehlung, 30  
 Entwicklertest, 20  
 Entwicklungssystem, 14  
 Entwicklungstest, 25  
 ERP, 4  
 Eskalation, 24, 32, 91, 94  
 Eskalationspfad, 43  
 EXCEL, 124  
 Export, 80

**F**

Fachbereich, 10, 23  
 Fehler, 11, 39, 64, 80  
 Fehlerbehebung, 32  
 Fehlerbereinigung, 13  
 Fehlercode, 85, 87  
 Fehlerdokumentation, 27  
 Fehlererkenntnis, 11  
 Fehlerklasse, 32

Fehlerkorrektur, 22  
 Fehlermanagement, 6, 7, 11, 20, 59, 95  
 Fehlertool, 59, 100  
 Fehlertracking, 99  
 Freigabe, 50, 56, 96  
 Freigabestatus, 112  
 Freigabeverfahren, 137

**G**

Gantt-Diagramm, 123, 127  
 Gefahrenübergang, 141  
 Gesamtmeilensteinplan, 15  
 Gesamtprojekt, 15  
 Gesamtprojektleitung, 29, 45  
 Geschäftsprozess, 24  
 Gesetzeswerk, 117  
 Grundschutzkatalog, 117  
 GUI, 85, 102  
 Gültigkeit, 116  
 Gültigkeitsbereich, 2

**H**

Hauptprozess, 61  
 Hauptspeicherbelegung, 103  
 Historie, 56, 112  
 Hotfix, 21, 24  
 Hotline, 11, 13, 34

**I**

Ideenspeicher, 29  
 Import, 80  
 Inbetriebnahme, 20  
 Industrienormen, 19  
 Integrationstest, 99  
 Inventardokument, 110  
 ISO 27001, 117  
 ISO/IEC 13335, 117  
 ISO/IEC 15504, 3  
 ISO/IEC 17799, 117  
 ITIL, 3  
 IT-Qualitätsmanagement, 5

**K**

Kapazitätsabgleich, 125, 131  
 Kapazitätsmanagement, 123, 129  
 Kick-off, 29–31, 44, 73, 92, 100  
 Kommunikationsweg, 43  
 Konfiguration, 23, 86, 95, 101  
 Kontierungselement, 137, 139  
 Korrektur, 12, 20  
 Korrekturplan, 111

Kosten, 123  
 Kostenart, 137  
 Kostenelement, 137  
 Kostenrahmen, 45  
 Kostenstellenplan, 133  
 Kostenverfolgung, 125  
 kritischer Pfad, 130

**L**

Lastenheft, 9, 23, 24, 90, 110  
 Leistungsbeschreibung, 110  
 Leistungsnachweis, 139  
 Life Cycle, 136  
 Liniencontrolling, 134  
 Locking, 26  
 Logdatei, 80, 87  
 Logfile, 65

**M**

Mandantenaufbau, 89  
 Mapping, 86, 95  
 Massentest, 79  
 Meilenstein, 9, 19, 26, 123  
 Meilensteinplan, 92, 128  
 Methodologien, 3  
 Microsoft Project, 125  
 Migration, 15, 71, 95, 96  
 Migrationsmanagement, 6  
 Migrationsschritt, 73  
 Monitor, 103

**N**

Nachabnahme, 21, 30, 31  
 Netzplan, 123, 129  
 Norm, 116  
 Nummernkreis, 86

**O**

Ordnerstruktur, 50  
 Organisation, 43

**P**

Partnersystem, 79, 99  
 Patch, 14, 58  
 Patchprozess, 21  
 Patchzyklus, 30  
 Performance, 23, 26, 34, 102  
 Performancedaten, 103  
 Performancetest, 80  
 Personalkosten, 51  
 Pflichtenheft, 9, 13, 23, 24, 57, 110

Pilotbetrieb, 19  
 Planwert, 134  
 Plausibilisierung, 98  
 Plausibilitätsprüfung, 83  
 PM, 125  
 PMS, 123  
 PPS, 4, 125  
 Priorisierung, 33  
 Priorität, 38, 64, 131  
 Probebetrieb, 7, 33, 34, 81, 96  
 Problemspeicher, 29, 32  
 Produktionsausfälle, 20  
 Produktionsplan, 110  
 Produktivdatenbank, 25  
 Produktivsystem, 34  
 Projektabbruch, 141  
 Projektbericht, 134, 140  
 Projektcontrolling, 133  
 Projektfortschritt, 51  
 Projektkostenplanelement, 138  
 Projektmanagement, 50, 123  
 Projektmanagementsystem, 123  
 Projektorganisation, 5  
 Projektrisiko, 140  
 Projektsteuerung, 140  
 Protokoll, 27, 36  
 Prozess, 98, 101  
 Prozessdokumentation, 91, 95  
 Prozesshierarchie, 26  
 Prozesskette, 98  
 Prozessor, 103  
 Prozessorleistung, 103  
 PSP, 138

**Q**

Qualität, 19  
 Qualitätsmanagement, 6, 19  
 Qualitätsnormen, 2  
 Quellsystem, 71, 85, 95

**R**

Rechenzentrum, 92  
 Rechenzentrumskapazität, 136  
 Rechnung, 6  
 Redundanz, 74  
 Referenz, 109  
 Referenzdokument, 60, 62  
 Regelkommunikation, 43  
 Release, 9, 22  
 Releaseplan, 9  
 Releaseplanung, 7

Ressource, 129  
 Ressourcenplanung, 59  
 Review, 33, 44, 58, 92, 121  
 Reviewprozess, 30  
 Revision, 27  
 Richtlinie, 56, 57, 116  
 Rückabwicklung, 141

## S

Sammelvorgang, 127  
 Schichtkalender, 131  
 Schnittstelle, 23, 63, 65, 79, 85, 86, 94  
 Schnittstellenabnahme, 82  
 Schnittstellenbetrieb, 7  
 Schnittstellentest, 21  
 Schulung, 59, 91, 94, 96  
 Service Center, 7  
 Service Level Agreement, 15, 73  
 Sicherheit, 115  
 Sicherheitsmanagement, 118  
 Simulation, 80  
 Softwarefehler, 11, 13  
 Sonderfälle, 79  
 Speicherauslastung, 103  
 Speichermanagement, 103  
 Spezifikation, 57  
 SPICE, 3  
 Standard, 117  
 Status, 45  
 Steuerung, 135  
 Steuerungsgruppe, 58  
 Steuerungsinstrument, 124  
 Stichprobe, 74  
 Subprozess, 35, 61  
 Support, 11, 13, 23, 45, 58, 92  
 Supportschleife, 11  
 Systemdokumentation, 110  
 Systemsupport, 99  
 Systemtabelle, 103  
 Systemverfügbarkeit, 45

## T

Taskliste, 59  
 Teilprojekt, 7, 14  
 Teilprojektbericht, 45  
 Teilprojektleitung, 23  
 Teilprojektstrukturen, 7  
 Teilprozess, 35, 61, 98  
 Test, 55  
 Testanforderungen, 22

Testbett, 82  
 Testdatei, 81  
 Testdaten, 22–24, 31, 58, 63, 91  
 Testdatenbett, 20  
 Testfallkatalog, 57, 61  
 Testlandschaft, 79  
 Testlokalität, 23  
 Testmandant, 74, 75, 88, 95  
 Testpersonal, 23  
 Testprotokoll, 59, 80  
 Testprozessschritt, 57  
 Testskript, 20, 22–24, 100  
 Testsystem, 14, 25, 59, 73  
 Testteam, 29  
 Testumgebung, 84, 91  
 Tracker, 58, 64  
 Tracking Tool, 14  
 Transaktion, 83  
 Transaktionscode, 86, 88

## U

Übergabe, 141  
 Updates, 21, 28  
 Uladung, 71, 74, 95

## V

Variante, 20  
 Verantwortlichkeiten, 19  
 Verrichtungsebene, 25  
 Versionierung, 112  
 Vorgang, 123  
 Vorgangsdauer, 125, 127  
 Vorgangsname, 125  
 Vorlaufstrecke, 19  
 Vorschrift, 117

## W

Wareneingangsprüfung, 22  
 Workaround, 33

## Z

Zeitnachweis, 51  
 Zertifizierung, 117  
 Ziele, 19  
 Zielsetzung, 19  
 Zielsystem, 88  
 Zugriffsrecht, 86  
 Zurückweisung, 22