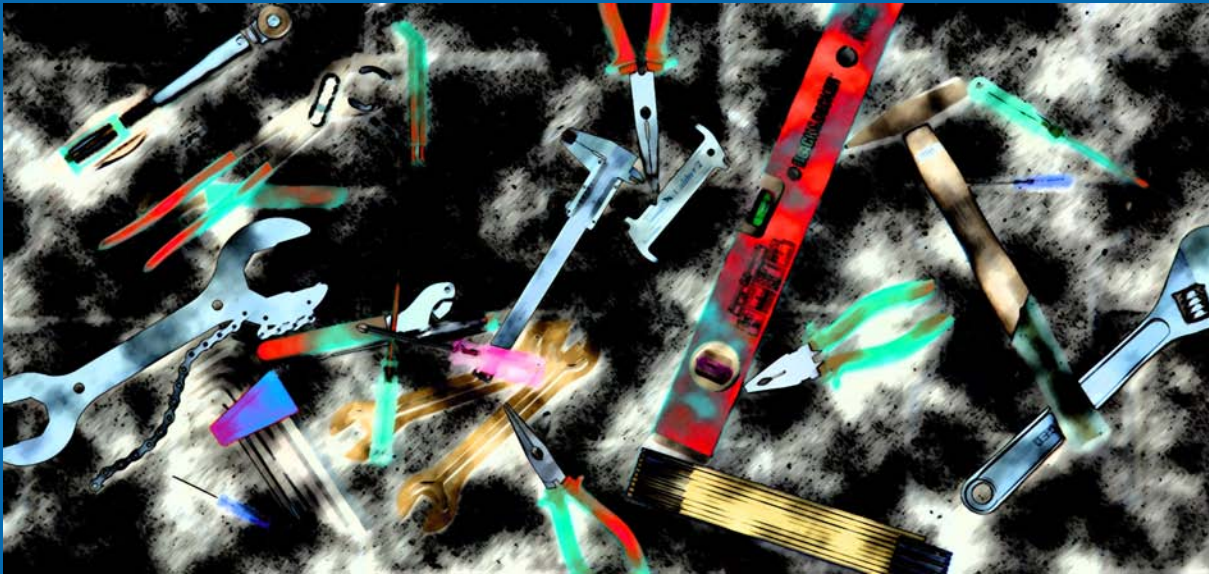


Michael Brenner

# Werkzeugunterstützung für ITIL-orientiertes Dienstmanagement

Ein modellbasierter Ansatz





**Werkzeugunterstützung für ITIL-orientiertes  
Dienstmanagement**  
– ein modellbasierter Ansatz

### **Bibliografische Information der Deutschen Nationalbibliothek**

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

Brenner, Michael:

Werkzeugunterstützung für ITIL-orientiertes Dienstmanagement

– ein modellbasierter Ansatz

Dissertation an der Fakultät für Mathematik, Informatik und Statistik der Ludwig Maximilians Universität München zur Erlangung des Grades Doctor rerum naturalium (Dr. rer. nat.), München, 2007

1. Berichterstatter: Prof. Dr. Heinz-Gerd Hegering

2. Berichterstatter: Prof. Dr. Helmut Krcmar

Die Informationen in diesem Buch wurden mit größter Sorgfalt erarbeitet. Dennoch können Fehler nicht vollständig ausgeschlossen werden. Verlag, Autoren und ggf. Übersetzer übernehmen keine juristische Verantwortung oder jedwede Haftung für eventuell verbliebene fehlerhafte Angaben oder deren Folgen.

ISBN: 9783837002010

© Michael Brenner

<http://www.mnm-team.org/~brennera/>

Herstellung und Verlag:

Books on Demand GmbH, Norderstedt, <http://www.bod.de>

## Danksagung

Mein erster und besonderer Dank gilt natürlich Prof. Dr. Heinz-Gerd Hegering, der als Chef einen zeitlichen und organisatorischen Rahmen schuf, der die Erstellung dieser Arbeit überhaupt erst möglich machte, und als Doktorvater mit einem unglaublichen, unermüdlichen Einsatz in vielfältigster Weise zum Gelingen dieser Arbeit beitrug. Großer Dank gebührt auch bei meinem Zweitgutachter, Prof. Dr. Helmut Krcmar, für seine kritischen Anmerkungen und Anregungen zur Arbeit sowie die überaus zügige Bearbeitung ihrer Vorversionen. Ohne die im MNM-Team vorzufindende Diskussionskultur hätte auch diese Arbeit nicht so entstehen können - vielen Dank an alle Kollegen für ihre Kritik, ihre Anregungen und ihre Unterstützung! Auch außerhalb der Reihen des MNM-Teams haben viele durch kritische Auseinandersetzung mit Teilergebnissen, Einbringung praktischer Erfahrungen, hilfreiche Anregungen und auf andere Arten zum Erfolg der Arbeit beigetragen. Stellvertretend seien hier Marcus Giese, Robert Kuhlig, Volker Leitzgen und Sven Walter genannt. Vielen Dank aber auch an die vielen hier nicht genannten, aber nicht vergessenen! Herzlichen Dank natürlich auch an meinen Vater, ohne dessen Hilfe beim Durchstehen mancher nicht so guten Zeit vor meinem Eintritt in das MNM-Team ich die Promotion nie hätte in Angriff nehmen können.

München, Juli 2007



# Zusammenfassung

In Unternehmen sind immer mehr, auch für den Geschäftserfolg kritische, Geschäftsprozesse von IT-Diensten abhängig. Bei der Vereinbarung dieser steht daher meist das einzuhaltende Qualitätsniveau bezüglich Parametern wie Verfügbarkeit oder Ausfallhäufigkeit im Mittelpunkt. Die Steuerung der IT-Dienstleistungserbringung mit dem Ziel eines effizienten Erreichens dieser Qualitätsvorgaben wird als (IT-)Dienstmanagement bezeichnet. Es erfordert nicht nur neue Methoden im Management der technischen IT-Infrastruktur, sondern auch eine Steuerung und Optimierung der Abläufe in der IT-Betriebsorganisation, der IT-Dienstmanagementprozesse.

Für die Gestaltung dieser Dienstmanagementprozesse werden, vor allem im europäischen Raum, immer häufiger die Richtlinien der IT Infrastructure Library (ITIL) herangezogen. Anders als im infrastrukturorientierten Dienstmanagement, steht die Forschung an Tools für Dienstmanagementprozesse erst am Anfang – für die Werkzeugunterstützung von ITIL-Prozessen sind bislang keine methodischen Ansätze entwickelt worden.

Basis für das hier vorgeschlagene Vorgehen in dieser Frage sind Klassifikationen von Dienstmanagement-Rahmenwerken, Dienstmanagementprozessen und IT-Managementwerkzeugen, welche überhaupt erst erlauben, für jeden Dienstmanagementprozess den geeigneten Typ von Werkzeug bestimmen zu können. Der im Anschluss vorgestellte modellbasierte Ansatz verfolgt das Ziel einer fundierten und nachvollziehbaren Konkretisierung von ITIL-Prozessen und der Entwicklung unterstützender Werkzeuge. Er ordnet die Referenzmodelle ausgewählter ITIL-Prozesse sowie einer grundlegenden Werkzeugarchitektur in einem auf dem Zachman-Framework basierenden Rahmen ein. Nach der Diskussion dieser Modelle schließt ein Ausblick auf die zahlreichen, wichtigen Forschungsfragenstellungen in diesem Umfeld die Arbeit ab.





# Abstract

In enterprises, ever more and ever more critical business processes depend on the delivery of IT services. Consequently, a main concern in their agreement is usually the definition of (quality of) service levels for parameters such as frequency of outages or overall availability. The management of IT provisioning with the focus on efficiently securing compliance with these service levels, is called IT Service Management. IT Service Management requires not only new methods for controlling the technical IT infrastructure, but also for managing the workflows in the IT service provider organization, the IT Service Management processes.

When designing these IT Service Management processes, IT organizations, particularly in Europe, frequently consult the guidelines of the IT Infrastructure Library (ITIL). However, unlike for infrastructure-oriented service management, so far very little research has been done regarding tool support for IT Service Management processes and no methodical approach for ITIL tool support has been developed yet.

Basis for the here presented approach to address this issue are classifications of service management frameworks, service management processes and IT management tools – in combination, they permit identifying the most suitable types of tools for service management processes. The subsequently presented, model-based approach for ITIL tool support pursues the goal of operationalizing ITIL processes and developing tools in a well-founded and reproducible way. It arranges the reference models of selected ITIL processes (and of a fundamental tool architecture) in a suitably adapted version of the Zachman framework. After the discussion of these models, an outlook on the numerous, important research questions in this field concludes this work.

## *Abstract*

# Inhaltsverzeichnis

<b>Zusammenfassung</b>	<b>vii</b>
<b>Abstract</b>	<b>ix</b>
<b>1 Einleitung</b>	<b>1</b>
1.1 Motivation . . . . .	1
1.1.1 Warum Dienstmanagement? . . . . .	1
1.1.2 Warum prozessorientiertes Dienstmanagement? . . . . .	4
1.1.3 Warum ITIL-orientiertes Dienstmanagement? . . . . .	8
1.1.4 Warum Werkzeugunterstützung für ITIL-orientiertes Dienstmanagement? . . . . .	9
1.2 Einordnung der Arbeit und Fragestellungen . . . . .	10
1.3 Aufbau der Arbeit . . . . .	12
<b>2 Begriffe und Konzepte</b>	<b>15</b>
2.1 IT-Service . . . . .	15
2.2 IT-Service-Provider . . . . .	17
2.3 Prozess . . . . .	18
2.4 Kontinuierliche Verbesserung . . . . .	20
2.5 Prozessmodellierung . . . . .	22
2.6 Benchmarking . . . . .	23
2.7 Best Practice . . . . .	24
2.8 Prozesskontrolle . . . . .	26
2.9 Rollen . . . . .	28
<b>3 ITIL im Kontext der ITSM-Rahmenwerke</b>	<b>31</b>
3.1 ITIL . . . . .	32
3.1.1 Herkunft und Zielsetzung . . . . .	32
3.1.2 Struktur . . . . .	34
3.1.3 Inhalte – Die Dienstmanagementprozesse . . . . .	36
3.1.4 Werkzeuge für ITIL . . . . .	47
3.1.5 Diskussion . . . . .	48
3.2 eTOM . . . . .	50
3.2.1 Herkunft und Zielsetzung . . . . .	50
3.2.2 Struktur und Inhalte . . . . .	53
3.2.3 Abbildung auf ITIL . . . . .	67

3.2.4	Diskussion . . . . .	68
3.3	CMM und CMMI . . . . .	69
3.3.1	Herkunft und Zielsetzung . . . . .	70
3.3.2	Struktur und Inhalte . . . . .	71
3.3.3	Abbildung auf ITIL . . . . .	72
3.3.4	Diskussion . . . . .	72
3.4	ITS-CMM . . . . .	72
3.4.1	Herkunft und Zielsetzung . . . . .	72
3.4.2	Struktur und Inhalte . . . . .	72
3.4.3	Abbildung auf ITIL . . . . .	73
3.4.4	Diskussion . . . . .	73
3.5	CobiT . . . . .	73
3.5.1	Herkunft und Zielsetzung . . . . .	73
3.5.2	Struktur und Inhalte . . . . .	74
3.5.3	Abbildung auf ITIL . . . . .	75
3.5.4	Diskussion . . . . .	76
3.6	eSCM . . . . .	76
3.6.1	Herkunft und Zielsetzung . . . . .	76
3.6.2	Struktur und Inhalte . . . . .	77
3.6.3	Abbildung auf ITIL . . . . .	78
3.6.4	Diskussion . . . . .	78
3.7	MOF . . . . .	79
3.7.1	Herkunft und Zielsetzung . . . . .	79
3.7.2	Struktur und Inhalte . . . . .	79
3.7.3	Abbildung auf ITIL . . . . .	80
3.7.4	Diskussion . . . . .	81
3.8	ITUP . . . . .	82
3.8.1	Herkunft und Zielsetzung . . . . .	82
3.8.2	Struktur und Inhalte . . . . .	82
3.8.3	Abbildung auf ITIL . . . . .	82
3.8.4	Diskussion . . . . .	83
3.9	ISO/IEC 20000 . . . . .	83
3.9.1	Herkunft und Zielsetzung . . . . .	83
3.9.2	Struktur und Inhalte . . . . .	84
3.9.3	Abbildung auf ITIL . . . . .	86
3.9.4	Diskussion . . . . .	86
3.10	Weitere Frameworks . . . . .	87
3.11	Einordnung und Ausblick . . . . .	89
3.11.1	Beziehungen zwischen ITSM-relevanten Rahmenwerken . . . . .	90
3.11.2	Einordnungsschema für Dienstmanagement-Rahmenwerke . . . . .	92
3.11.3	Zusammenfassung und Ausblick . . . . .	96
<b>4</b>	<b>Möglichkeiten und Grenzen der Werkzeug-Unterstützung</b> . . . . .	<b>99</b>
4.1	Motivation für spezialisierte ITIL-Werkzeuge . . . . .	100

4.2	Klassifikation von Werkzeugen . . . . .	102
4.2.1	Bestehende Klassifikationen . . . . .	102
4.2.2	Einordnung anhand Lebenszyklus- und technischer Orientierung . . . . .	104
4.2.3	Verfeinerung der Einordnung für prozessorientierte ITSM-Betriebswerkzeuge . . . . .	106
4.3	Klassifikation von ITIL-Prozessen . . . . .	108
4.3.1	Prozessklassifikation in der Literatur . . . . .	109
4.3.2	Kriterien für die Klassifikation von Dienstmanagementprozesse . . . . .	110
4.3.3	Annahmen zum Einsatzszenario . . . . .	114
4.3.4	Bestimmung der Kriterienausprägung für ITIL-Prozesse . . . . .	115
4.3.5	Zusammenfassung der Ergebnisse . . . . .	118
4.3.6	Einteilung in Quadranten der Werkzeugeignung . . . . .	121
4.4	Zusammenfassung und Fazit . . . . .	124
<b>5</b>	<b>Modellbasierter Ansatz</b>	<b>127</b>
5.1	Motivation . . . . .	127
5.2	Abgrenzung . . . . .	129
5.3	Ziele . . . . .	130
5.4	Methode . . . . .	132
5.4.1	Ordnungsrahmen . . . . .	133
5.4.2	Vorgehen . . . . .	139
<b>6</b>	<b>Umsetzung</b>	<b>141</b>
6.1	Owner View . . . . .	141
6.1.1	Abgrenzung und Zielsetzung . . . . .	142
6.1.2	Erfahrungen bei der Erstellung von ITIL-Modellen . . . . .	142
6.1.3	Allgemeine Darstellung der Modelle . . . . .	145
6.1.4	Data Model . . . . .	145
6.1.5	Function Model . . . . .	150
6.1.6	People Model . . . . .	153
6.2	Designer View . . . . .	154
6.2.1	Grundlegender Architekturansatz . . . . .	154
6.2.2	Modelle . . . . .	155
6.3	Überlegungen zum Builder View . . . . .	156
6.4	Erfahrungen . . . . .	157
<b>7</b>	<b>Zusammenfassung und Ausblick</b>	<b>159</b>
7.1	Ergebnisse und Fazit . . . . .	159
7.2	Offene Forschungsfragestellungen . . . . .	161
<b>A</b>	<b>Modelle</b>	<b>163</b>

*Inhaltsverzeichnis*

<b>Literaturverzeichnis</b>	<b>193</b>
<b>Abkürzungsverzeichnis</b>	<b>209</b>
<b>Abbildungsverzeichnis</b>	<b>215</b>
<b>Index</b>	<b>219</b>

# 1 Einleitung

## 1.1 Motivation

Die Anforderungen, die Unternehmen an ihre IT stellen, sind im Wandel begriffen und verlangen zunehmend nach einer kontrollierteren und stärker an den Anwenderbedürfnissen ausgerichteten IT-Leistungserbringung. Die Erfüllung dieser geänderten Bedürfnisse verlangt nach einer neuen Ausrichtung des IT-Managements mit entsprechenden Methoden, wofür sich der Begriff Dienstmanagement beziehungsweise *IT Service Management* (ITSM) etabliert hat. Die Erreichung der mit dem Dienstmanagement verknüpften Visionen setzt allerdings die Beantwortung zahlreicher, bislang offener Fragen voraus.

### 1.1.1 Warum Dienstmanagement?

Die überwiegende Mehrheit der Geschäftsprozesse in einem Unternehmen ist heute vom Funktionieren der IT-Systeme und der auf ihnen realisierten Anwendungen abhängig. In größeren Unternehmen hat sich, neben einer oder mehrerer Abteilungen für IT-Entwicklung, für den Betrieb der Anwendungen und der ihnen zugrunde liegenden Infrastruktur meist eine interne IT-Betriebsorganisation etabliert. Diese, in der Regel von der Software- und IT-Systementwicklung getrennte Organisation ermöglicht als Dienstleister ihren „Kunden“ (den Fachabteilungen des Unternehmens) IT-Anwendungen zur Durchführung ihrer Geschäftsprozesse zu nutzen.

*Kontext der Arbeit*

Als zentrale Aufgabe der internen IT-Betriebsorganisation wurde in der Vergangenheit primär die Aufrechterhaltung des Betriebs mittels eines effektiven Netz- und Systemmanagements gesehen. Dieses technische IT-Management befasst sich vorwiegend mit den Fragestellungen des effizienten Einsatzes von IT-Komponenten. Die Herausforderungen in diesem Bereich ergeben sich vorwiegend aus den Problemen der Komplexität, Heterogenität und Verteiltheit der zu managenden Infrastruktur. Große Erleichterungen bei der Bewältigung dieser Probleme wurden Administratoren und Systembetreuern durch die Entwicklung von Werkzeugen des Netz- und Systemmanagements, wie z.B. Management-Plattformen, möglich [HAN99].

*Netz- und Systemmanagement*

Allerdings ziehen Fortschritte in der Steuerung des Einsatzes von IT-Ressourcen nicht immer im gleichen Maße Fortschritte für die Anwender und Kunden nach

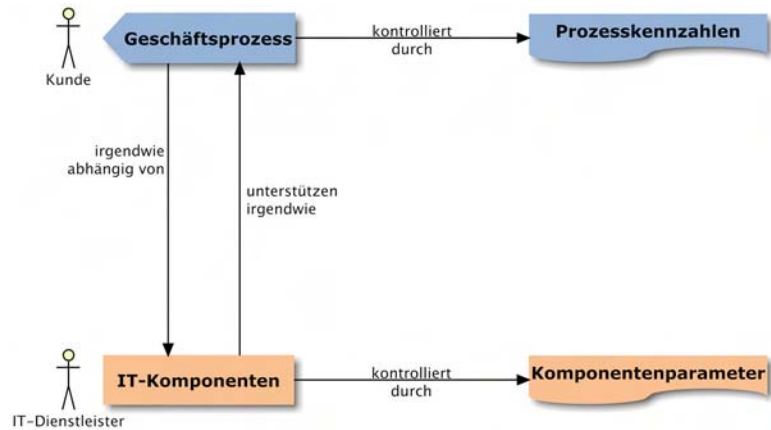


Abbildung 1.1: Problematik des komponentenorientierten IT-Managements

*Defizite des  
komponenten-  
orientierten  
Managements*

sich. Auch eine deutliche Verbesserung einiger im Netz- und Systemmanagement überwachten technischen Parameter hat nicht immer zur Folge, dass die Anwender das Verhalten der von ihnen genutzten Anwendungen als besser wahrnehmen oder der unterstützte Geschäftsprozess besser funktioniert. Dieses Phänomen hat seinen Ursprung in den stark unterschiedlichen Sichtweisen auf die IT und der damit verbundenen Problematik, wie sie in Abbildung 1.1 dargestellt ist.

*Kluft zwischen  
Geschäftsprozes-  
sen und  
IT*

Die Prozesse des IT-Kunden werden meist gemanagt und ihre Leistung mittels Prozesskennzahlen bestimmt. Ebenso besteht in der Regel auch ein Management der IT-Infrastruktur im Sinne einer Kontrolle und Steuerung der IT-Komponenten mittels Komponentenparameter. Aber die technische Infrastruktur ist meist über viele Jahre gewachsen, auch die Nutzung der Systeme durch die Anwender hat sich im Laufe der Zeit verändert – der aktuelle Zusammenhang zwischen der Leistung der Kundenprozesse und dem Status der IT-Komponenten ist meistens unklar. Der Kunde weiß im Normalfall wenig über die Infrastruktur hinter den für ihn sichtbaren IT-Leistungen (z.B. Betrieb seiner Applikationen) – umgekehrt hat auch die IT-Organisation selten mehr als eine vage Vorstellung von den durch die IT-Infrastruktur unterstützten Prozessen. Fragen wie „Verursacht ein 24-stündiger Ausfall der Komponente A einen höheren Geschäftsverlust als ein Ausfall der Komponente B?“ können in vielen Fällen weder vom Kunden noch vom IT-Dienstleister befriedigend beantwortet werden. Auch wenn in der Gesamtorganisation viel Teilwissen vorhanden ist, über die Gesamtzusammenhänge können meist nur vage Aussagen getroffen werden. Dementsprechend kann das Management der IT-Infrastruktur auch nur schwer auf einen maximalen Anwender- und Kundennutzen hin optimiert werden.



Das Dienstmanagement oder IT-Dienstmanagement<sup>1</sup> ist eine eigene Teildisziplin des IT-Managements, die diese Lücke schließen soll. Es erweitert die Aufgaben des IT-Managements, welches sich bis dahin vorwiegend mit der effizienten Verwaltung der IT-Ressourcen nach technischen Gesichtspunkten befasste, um das Ziel diesen Ressourcen-Einsatz hinsichtlich der vom Nutzer wahrgenommenen Qualität des IT-Dienstes (*IT-Service*) zu optimieren.

*IT-Dienstmanagement*

Das wohl zentralste Konzept des IT-Dienstmanagements ist dabei das *Service Level Management* (SLM), in welchem wiederum die Vereinbarung eines *Service Level Agreement* (SLA) im Mittelpunkt steht. In einem SLA wird der zu erbringende Dienst spezifiziert. Wo es, wie oben dargelegt, um die Unterstützung der Mitarbeiter einer Fachabteilung bei der Durchführung von Geschäftsprozessen geht, also im Kontext der so genannter *Enterprise Services*, ist ein Dienst dementsprechend eine Entität, die durch seine Festlegung in einem SLA geschaffen wird<sup>2</sup>. Der Schwerpunkt der Dienstspezifikation liegt dabei in der Regel nicht in der Beschreibung der Funktionalität des Dienstes, sondern in der Festlegung von Maßen für nicht-funktionale Qualitätsaspekte sowie eines während der Dienstleistung einzuhaltenen Qualitätsniveaus. Für die Qualität des Dienstes hat sich der Begriff *Quality-of-Service* (QoS) etabliert, als Begriff für ein Maß, das zur Bestimmung dieser Qualität dient, dementsprechend *Service-Parameter* oder *Quality-of-Service-Parameter* (QoS-Parameter), und das Niveau der Dienstqualität, das bei der Dienstleistung einzuhalten ist, wird als *Service Level* bezeichnet.

*SLM als zentrales Konzept des Dienstmanagements*

Diese Zusammenhänge sind, anlehnend an [Lew99], in Abbildung 1.2 zusammenfassend dargestellt: Zwischen Kunde und IT-Dienstleister wird ein SLA für einen Service (z.B. Zugriff auf ein Buchungssystem) vereinbart und festgelegt, mittels welcher Service-Parameter (z.B. Verfügbarkeit oder Antwortzeit) die Service-Qualität kontrolliert werden soll. Im Rahmen dieser Definition der Service-Parameter sollte in einem SLA auch festgelegt werden, wie diese Parameter zu messen und in welchen Intervallen sie zu bestimmen sind. Schließlich wird im SLA festgehalten, welcher *Service Level*, also welches Qualitätsniveau bei der Dienstleistung einzuhalten ist. *Service Levels* sind hierbei festgelegte Wertebereiche für Tupel von QoS-Parametern. Beispielsweise kann als „Service Level Gold“ eine Verfügbarkeit größer 99.9% und Antwortzeit kleiner 0,5 Sekunden, als „Service Level Silver“ eine Verfügbarkeit größer 99% und Antwortzeit kleiner 1 Sekunde festgelegt werden.

*Kernbegriffe des Service Level Management*

Allerdings bestehen meist vielfältige Abhängigkeiten zwischen Diensten und den Komponenten der IT-Infrastruktur, welche die Bereitstellung von Diensten, die Kontrolle der Service-Parameter und die Einhaltung von *Service Levels* zu anspruchsvollen Aufgaben für das Infrastruktur-Management machen. Das hierfür

*Technisches Dienstmanagement*

<sup>1</sup>Die Begriffe „IT-Dienstmanagement“, „Dienstmanagement“, „Service-Management“, „IT-Service-Management“ sowie die gebräuchliche Abkürzung „ITSM“ für letzteres sind synonym.

<sup>2</sup>Für eine Diskussion des Begriffes „Dienst“, siehe auch Abschnitt 2.1 auf Seite 15.

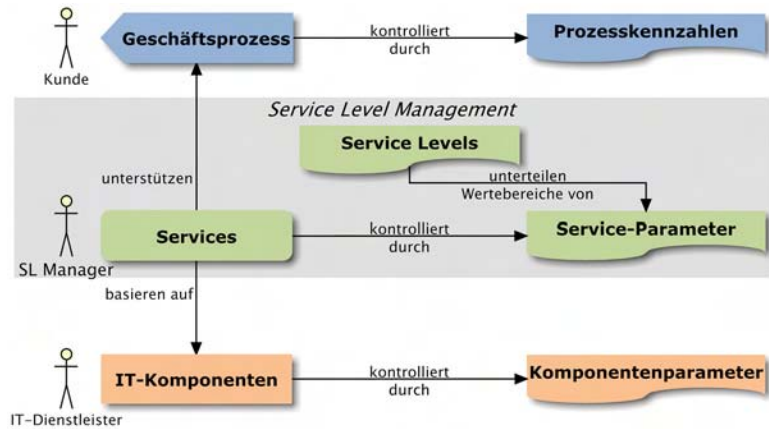


Abbildung 1.2: Service Level Management im komponentenorientierten Netz- und Systemmanagement

notwendige technisch orientierte Dienstmanagement baut auf dem komponentenorientierten Netz- und Systemmanagement auf.

*Service-  
Management-  
Plattform*

Die Forschung zu Werkzeugen der dienstorientierten Steuerung der Infrastruktur-Komponenten, wie beispielsweise Tools zur dienstorientierten Event-Korrelation oder der Modellierung von Diensten in einer Service-MIB, ist derzeit noch in vollem Gange. Die Vision ist diese, zum Teil noch zu entwickelnden Werkzeuge in eine Service-Management-Plattform [DR02] (siehe Abbildung 1.3) zu integrieren, die als „Master-Tool“ eine dienstorientierte Kontrolle und Steuerung der IT-Infrastruktur von einer Konsole aus erlaubt.

### 1.1.2 Warum prozessorientiertes Dienstmanagement?

*vorläufige  
Grenzen der  
Automatisierung*

Im technischen Dienstmanagement können mittlerweile standardisierte, kommunikationsorientierte Dienste, wie Sprach- oder Videoübertragung, in homogenen Infrastrukturen weitgehend automatisch aktiviert, überwacht und gesteuert werden. Die bei der IT-Leistungserbringung für Geschäftsprozesse im Mittelpunkt stehenden *Enterprise Services* haben allerdings größtenteils andere Eigenschaften. Bei ihnen ist in der Regel eine sehr viel größere Zahl verschiedener Dienste auf einer sehr heterogeneren IT-Infrastruktur zu managen. Insbesondere die Heterogenität hat bislang verhindert, dass sich ein technischer Standard zur Management-Kommunikation mit allen relevanten Komponenten einer IT-Infrastruktur durchsetzen konnte – meistens sind also nicht alle Komponenten für die Steuerung mittels einer einheitlichen Management-Architektur instru-

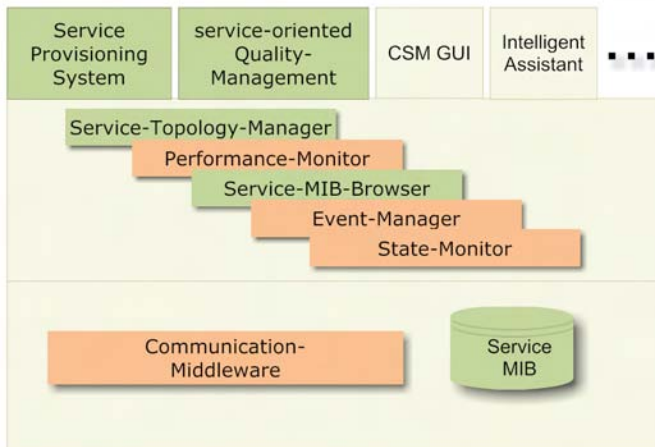


Abbildung 1.3: Service Management Platform [DR02]

mentiert. Dies hat zur Folge, dass man bereits auf der Ebene des Netz- und Systemmanagements von der Steuerung der kompletten Infrastruktur durch eine zentrale Management-Plattform-Anwendung heute oft noch weit entfernt ist. Die Vision des umfassenden Dienstmanagements von einer einzelnen Konsole der Service-Management-Plattform aus wird also erst in einiger Zeit realisiert werden können.

Dies lässt die Frage berechtigt erscheinen, wie man mit den vorhandenen Dienstmanagement-Aufgaben umgeht, für die heutige Werkzeuge noch keine umfassende Automatisierung bieten. Für viele Problembereiche, beispielsweise die Erstellung und stete Aktualisierung einer grundlegenden Dokumentation der Infrastruktur, gibt es prinzipiell nicht nur technische, sondern auch organisatorische Ansätze. Zwar bieten diese oft keine den technischen Ansätzen ebenbürtige Effizienz, andererseits sind sie Technologie-unabhängig und können zuverlässig und pragmatisch Unterstützung bieten, wo automatisierte technische Lösungen auf absehbare Zeit noch nicht verfügbar sein werden. Beispielsweise lassen sich durch Auto-Discovery-Werkzeuge große Mengen an Infrastruktur-Daten gewinnen, die in dieser Menge und diesem Detailgrad kaum manuell erhebbar wären. Andererseits sind für das Management aber häufig auch Daten von Interesse, die sich aus den verschiedensten Gründen nicht oder nur mit unverhältnismäßig hohem Aufwand automatisch sammeln lassen – hinzu kommt, dass für seltenere oder neuere Komponenten Autodiscovery-Werkzeuge ständig angepasst werden müssen, was die Flexibilität bei der Infrastruktur-Gestaltung einschränkt. Auch die automatische Dokumentation von Diensten sowie das Erkennen von Abhängig-

*organisatorische  
Maßnahmen  
können  
pragmatische  
Lösungen bieten*

keiten zwischen Diensten und Komponenten ist in der Praxis noch nicht gewährleistet. Hierbei können organisatorische Richtlinien und Verfahrensweisen (zur Dokumentation der Infrastruktur und Regelung von Infrastruktur-Änderungen) den Aufbau und die Aktualisierung eines Infrastruktur-Modells ermöglichen, das nicht von einer technologie-spezifischen, automatischen Erfassbarkeit aller gewünschter Elemente abhängt.

„*Availability of Service Matters Above All Else*“

Die Notwendigkeit eines prozessorientiertes Dienstmanagements lässt auch direkter aus den Anforderungen des *Service Level Management* ableiten – genauer gesagt aus der effizienten Einhaltung von *Service Levels*, speziell bezüglich einer Mindestverfügbarkeit. Zwar bleibt die Steuerung Performanz-bezogener QoS-Parameter (wie Antwortzeit) eine beinahe rein technische Fragestellung, doch der wohl grundlegendste Service-Parameter ist die *Availability* – die Verfügbarkeit des Dienstes schließlich die Voraussetzung um ihm überhaupt eine Performanz beimesen zu können. Performanz-bezogene QoS-Parameter spielen bei *Enterprise Services* meist eine weniger wichtige Rolle als bei Kommunikations- und Multimediaanwendungen. Umfragen bestätigen, dass bei IT-Services die Verfügbarkeit der insgesamt wichtigste Einflussfaktor auf die Kunden- und Anwenderzufriedenheit ist [Nex00] – oder prägnant ausgedrückt: „Availability of service matters above all else“<sup>3</sup> [LBN02].

*Bestimmung der Verfügbarkeit*

Die Verfügbarkeit wird jedoch nicht alleine von der technischen Realisierung (z.B. redundante Auslegung von Systemkomponenten) des Dienstes beeinflusst, sondern in einem hohen Maß auch durch die organisatorische Gestaltung des IT-Betriebs. Definiert wird Availability im Allgemeinen als das Verhältnis der Zeit, die eine Element (System, Komponente, Dienst, ...) funktionstüchtig ist (*Uptime*) zur gesamten Zeit der Beobachtung oder Messung. Die Differenz zwischen diesen beiden Werten, also die Zeit, in der das Element nicht funktionstüchtig ist, wird als *Downtime* bezeichnet:

$$Availability = \frac{Uptime}{TotalTime} = \frac{Uptime}{Uptime + Downtime}$$

*MTBF und MTTR*

Eine durchschnittliche oder zu erwartende Verfügbarkeit wird in der Regel basierend auf zwei etablierten Parametern bestimmt, der *Mean Time Between Failures* (MTBF)<sup>4</sup> und der *Mean Time To Repair* (MTTR)<sup>5</sup>. Wie in Abbildung 1.4 anschaulich dargestellt ist, entspricht die *Time To Repair* der *Downtime* und die *Uptime* der Differenz zwischen dieser und der *Time Between Failures*. Damit erhält man, bei Betrachtung der jeweiligen Mittelwerte, die Netzmanagement häufig angetroffene Availability-Definition (vgl. z.B. [ITU97]):

<sup>3</sup>Frei übersetzt: „Nichts ist wichtiger als die Verfügbarkeit des Dienstes“.

<sup>4</sup>Durchschnittliche Zeitdauer zwischen zwei Ausfällen.

<sup>5</sup>Durchschnittliche Zeitdauer bis zur Reparatur – genau genommen ist hiermit die Zeit bis zur vollständigen Wiederherstellung der Funktion gemeint, also ggf. inklusive Wiederanlaufzeit.

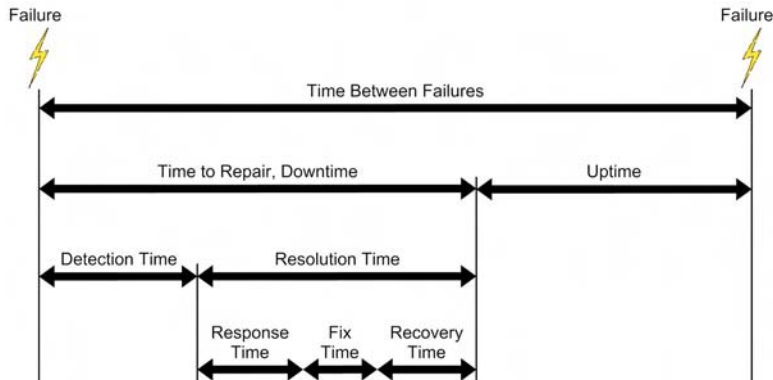


Abbildung 1.4: Availability (in Anlehnung an [vB05])

$$Availability = \frac{MTBF - MTTR}{MTBF}$$

Dies lässt sich auch ausdrücken als (vgl. [Nat96]):

$$Availability = 1 - Unavailability = 1 - \frac{MTTR}{MTBF}$$

Dies bedeutet, dass die *Unavailability* (Nichtverfügbarkeit) im gleichen Maße von MTBF wie MTTR abhängt, d.h. eine Verdoppelung der MTBF hat den gleichen Einfluss auf die Gesamtverfügbarkeit wie eine Halbierung der MTTR. Ein effizientes dienstorientiertes Management der Verfügbarkeit muss also die Faktoren, die zur Wiederherstellungszeit beitragen im selben Maß berücksichtigen wie jene, die die Ausfallhäufigkeit eines Dienstes beeinflussen. Auf beide Parameter haben nicht nur technische, sondern auch organisatorische Gegebenheiten einen maßgeblichen Einfluss.

*auch Wiederherstellungszeit wichtig für Verfügbarkeit*

Eine Schätzung von **Gartner** geht davon aus, dass 80% der unerwünschten IT-Ausfälle nicht auf technisches Versagen von Komponenten zurückzuführen sind, sondern auf menschliche Fehler und mangelhafte Verfahrensweisen („people and process issues“) in der IT-Entwicklung und dem IT-Betrieb [Sco99]. Die MTBF für einen Dienst hängt also nicht unwesentlich von der organisatorischen Gestaltung des IT-Betriebes ab. Noch unmittelbarer ist aber der Einfluss der „people and process issues“ bei der der MTTR.

*80% der Ausfälle durch „people and process issues“*

## 1 Einleitung

*Wiederherstellungszeit stark von organisatorischen Abläufen abhängig*

In der Praxis sind die in Abbildung 1.4 dargestellten Zeitintervalle für *Detection Time*, *Response Time*, *Fix Time* und *Recovery Time* alle stark von der Effektivität der Abläufe in der IT-Organisation abhängig. Zwar können eine optimierte Gestaltung der IT-Infrastruktur und der Einsatz von Werkzeugen des Netz- und Systemmanagements diese Zeiten bis zu einem gewissen Grad beeinflussen, z.B. kann die durchschnittliche *Detection Time* durch Einsatz von Monitoring-Werkzeugen verringert werden, im Kern bleiben die Aufgaben dahinter aber zu einem großen Teil manueller Natur. Die Verfügbarkeit ist also abhängig davon, wie schnell und effizient der Entstörungsprozess läuft. Wie lange es dauert, bis die von einem Anwender eingereichte Störungsmeldung bearbeitet wird, wann sie (falls die Störung durch einen Systemausfall verursacht ist) an die geeigneten technischen Abteilung weitergeleitet wird, ab wann sich daraufhin ein Spezialist der Störung annimmt, wie lange die Störungsbehebung braucht und wie schnell der Dienst wieder reaktiviert werden kann – alles wird wesentlich durch das Management des Entstörungsprozesses, das heißt auch den Einsatz personeller Ressourcen der IT-Organisation, beeinflusst. Somit ist gerade die MTTR eines Dienstes von organisatorischen Abläufen abhängig.

*IT-Services basieren auf IT-Komponenten und IT-Prozessen*

Die Prozesse in der IT-Organisation wirken sich also auch auf die Verfügbarkeit und damit auf den wohl gängigsten Qualitätsparameter eines Dienstes aus. Die Darstellung aus Abbildung 1.2 muss daher erweitert werden: IT-Dienste basieren nicht nur auf IT-Komponenten, sondern auch auf Prozessen in der IT-Organisation, den IT-Prozessen bzw. Dienstmanagementprozesse (siehe Abbildung 1.5 auf der nächsten Seite). In der Praxis wird diesem Umstand bei der Gestaltung von SLAs heute schon in Teilen Rechnung getragen, indem beispielsweise auch Erreichbarkeitszeiten der Support-Hotline, maximale Lösungszeiten für Störungen usw. festgelegt werden.

### 1.1.3 Warum ITIL-orientiertes Dienstmanagement?

*Frameworks für Dienstmanagementprozesse*

Richtlinien, wie diese Prozesse des Dienstmanagements zu gestalten sind, existieren in der Form verschiedener ITSM-Frameworks. Die *IT Infrastructure Library* (ITIL) beschreibt in einer Reihe von Büchern ein Rahmenwerk von Dienstmanagementprozesse und liefert zu jedem dieser Prozesse Leitlinien und Empfehlungen. Das Rahmenwerk und die einzelnen Prozessbeschreibungen werden dabei nicht formal dargestellt. Prozessbeschreibungen erfolgen in Buchkapiteln, die nur ansatzweise über eine gemeinsame, definierte Struktur verfügen und abgesehen von einzelnen Tabellen, Listen und Abbildungen (welche meist keiner definierten Notation entsprechen) keine formalen Beschreibungselemente enthalten.

*ITIL hat mit Abstand größte Verbreitung*

Trotzdem kann die ITIL, zumindest im europäischen Raum, als de-facto Standard für Dienstmanagementprozesse angesehen werden. Auch wenn entsprechende Untersuchungen in Details zu unterschiedlichen Ergebnissen kommen (vgl. [Blu06] und [Sch04]), so steht doch außer Frage, dass unter den Rahmenwerken

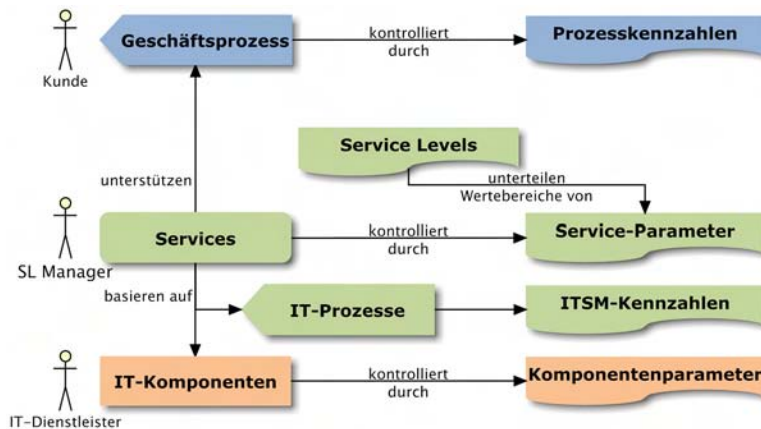


Abbildung 1.5: Vervollständigte Sicht auf das Service Management

für prozessorientiertes IT-Dienstmanagement die ITIL die mit Abstand größte Verbreitung genießt. Zumindest mittelfristig dürfte sich hieran nicht viel ändern, ist doch auch der neue internationale Standard für IT-Dienstmanagement, ISO/IEC 20000, weitgehend an den in der ITIL definierten Prozessen ausgerichtet. Auf ITIL baut zudem eine zunehmende Zahl ergänzender ITSM-Rahmenwerke von IT- und Beratungsfirmen auf, z.B. das *Microsoft Operations Framework* (MOF).

#### 1.1.4 Warum Werkzeugunterstützung für ITIL-orientiertes Dienstmanagement?

Wie oben dargelegt, hängen also IT-Dienste nicht nur von der Funktion von IT-Komponenten, sondern auch von der Effektivität der Dienstmanagementprozesse ab. Um eine Einhaltung von *Service Levels* garantieren zu können, müssen also in aller Regel nicht nur IT-Komponenten, sondern auch Dienstmanagementprozesse quantitativ gemanagt werden. Um beim Beispiel der Verfügbarkeit und der Darstellung in Abbildung 1.4 zu bleiben: Im Entstörungsprozess (in ITIL: *Incident Management*) müssen gegebenenfalls bei einer größeren Störung die Ressourcen so eingeteilt werden, dass die Länge der Reaktionszeit (*Response Time*), und der Lösungszeit (*Fix Time*) nicht zur Verletzung einer im SLA getroffenen Vereinbarung zur Gesamtverfügbarkeit führen. Auch sollten rechtzeitig vor der Verletzung von SLAs oder intern festgelegten Schwellwerten möglichst automatisch Eskalationen ausgelöst werden. Dies kann zwar prinzipiell auch mit manuellen Mitteln (Karteikarten, Zeittafeln, Telefon, Email usw.) erreicht werden, es ist aber leicht

*auch der Einsatz personeller Ressourcen muss gemanagt werden*

einzusehen, dass in größeren Szenarien, in denen es vielleicht dutzende oder hunderte offene Störungen zur selben Zeit bearbeitet werden müssen, eine effiziente Steuerung des Entstörungsprozesses nur mit Hilfe geeigneter IT-Tools geschehen kann.

### *Workflow- Management- Tool für ITIL-Prozesse*

Eine zukünftige Service-Management-Plattform würde also auch ein solches Werkzeug mit integrieren müssen. Die Funktionalität dieses Tools sollte eine Kontrolle und Steuerung wichtiger Dienstmanagement-Prozesse erleichtern, beispielsweise durch eine Statusüberwachung bei der Bearbeitung von Störungen, Automatisierung zeitgesteuerter Eskalationen, Kommunikation von Arbeitsaufträgen et cetera – kurz, das Management des Workflows.

### *ITIL-Tools bisher kaum wissenschaftlich untersucht*

Im Vergleich zu Werkzeugen des Komponentenmanagements sind Werkzeuge für das Management von Dienstmanagementprozesse aber bislang kaum Gegenstand wissenschaftlicher Untersuchungen gewesen. Es existieren einige wissenschaftliche Ansätze zu Werkzeugen für die Steuerung oder (Teil-)Automatisierung von Abläufen die als Teile von ITIL-Prozessen gesehen werden können (z.B. [Dre95], [BS04] oder [Kel05]). Keines dieser Konzepte adressiert aber die Koordination und Steuerung komplexerer Abläufe unter Beteiligung mehrerer Personen (eventuell auch aus unterschiedlichen Organisationen), wie sie ITIL-Prozesse darstellen.

## 1.2 Einordnung der Arbeit und Fragestellungen

### *grundlegende Forschungsfelder im ITSM*

Im doch recht weiten Forschungsfeld des IT-Dienstmanagements kann sich die vorliegende Arbeit nur mit einem spezifischen Ausschnitt befassen. Eine allgemeine Einordnung der Fragestellungen des Dienstmanagements ist in Abbildung 1.6 illustriert: Wie oben argumentiert, müssen für das IT-Dienstmanagement sowohl Elemente der IT-Infrastruktur wie auch der Einsatz von Personal gemanagt werden. Hierzu können unterschiedliche Mittel zum Einsatz kommen: organisatorische Maßnahmen (z.B. Prozessmanagement) oder auch konkretere Werkzeuge wie eben IT-Tools<sup>6</sup>. Prinzipiell können organisatorische Maßnahmen zum Management der Infrastruktur-Elemente beitragen (Pfeil 2 in Abbildung 1.6), wie z.B. Richtlinien des ITIL *Configuration Management*, meistens dienen sie aber mehr der Steuerung des Personaleinsatzes (Pfeil 1), wie es beispielsweise beim *Incident Management* der Fall ist. Auch IT-Werkzeuge können beim Management des Personaleinsatzes unterstützend wirken (Pfeil 3), wie z.B. Workflow-Management-Tools, oder, wie klassische Werkzeuge Netz- und Systemmanagements, die Steuerung der Infrastruktur-Elemente erleichtern (Pfeil 4).

---

<sup>6</sup>Auch wenn Elemente des Prozessmanagements, Kennzahlensysteme usw. häufig auch als Management-Werkzeuge bezeichnet werden: Der Schwerpunkt dieser Arbeit liegt auf IT-Werkzeugen, daher werden im Folgenden die Begriffe Tool und Werkzeug synonym verwendet und stehen für IT-Tools, also IT-Systeme, die Management-Aufgaben unterstützen.



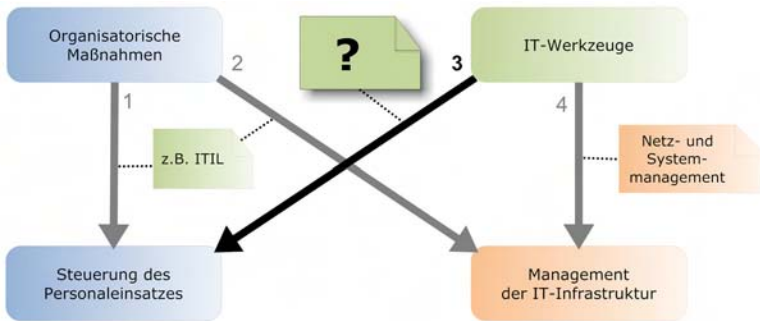


Abbildung 1.6: Einordnung der zentralen Fragestellung

Auch wenn letztlich alle Bereiche teilweise berührt werden, so ist die vorliegende Arbeit doch schwerpunktmäßig bei Pfeil 3, den IT-Werkzeugen, die Aktivitäten der am Dienstmanagement beteiligten Personen unterstützen, einzuordnen. Vereinfacht gesagt befassen sich Frameworks für das prozessorientierte Dienstmanagement (wie die ITIL) mit der Gestaltung organisatorischer Managementmaßnahmen, um sowohl den Personaleinsatz wie auch den Ressourceneinsatz zu optimieren – also mit den Beziehungen, die in Abbildung 1.6 durch die Pfeile 1 und 2 repräsentiert sind. Dafür, wie dies gestaltet sein soll, wird im Rahmen dieser Arbeit auf die ITIL zurückgegriffen. Eine Weiterentwicklung der ITIL ist dabei nicht primärer Gegenstand dieser Arbeit – d.h. auch wenn im Laufe der Arbeit ITIL-Richtlinien konkretisiert werden, so dient dies vorrangig dem Ziel der ITIL-Werkzeugunterstützung und nicht einer Überarbeitung der ITIL an sich. Der Einsatz von IT-gestützten Management-Tools zur Kontrolle der IT-Infrastruktur, repräsentiert durch Pfeil 4, ist das Thema der Forschung im Netz- und Systemmanagement. Das Forschungsgebiet, das sich mit dem Einsatz von IT-Werkzeugen zur Steuerung des Personaleinsatzes (Pfeil 3) befasst, wird für den allgemeinen Fall häufig als *Computer Supported Cooperative Work* (CSCW) bezeichnet, ist aber für die Domäne des IT-Dienstmanagements noch weitgehend unerforscht. Hier liegt der Schwerpunkt dieser Arbeit, auch wenn natürlich weitergehende Fragen bezüglich der Werkzeug-Integration (also zwischen Pfeil 3 und Pfeil 4) nicht ganz ausgeklammert werden können.

*Einordnung und Abgrenzung der Arbeit*

Anders als bei Werkzeugen des infrastrukturorientierten IT-Managements steht die Forschung bezüglich einer Tool-Unterstützung für die organisatorischen (bzw. prozessorientierten) Aspekte des Dienstmanagements erst am Anfang. Zur Werkzeugunterstützung für ITIL-orientiertes Dienstmanagement ergeben sich eine Reihe von grundlegenden Fragestellungen, die bislang nur partiell oder noch gar nicht wissenschaftlich untersucht worden sind:

## 1 Einleitung

### *offene Fragen zu IT-Werkzeugen für ITIL-basiertes Dienstmanage- ment*

- Wie ordnet sich die ITIL in die Landschaft der Frameworks und Standards für die Gestaltung von Prozessen für den IT-Betrieb ein? Lassen sich für andere Frameworks entwickelte Werkzeugkonzepte für ITIL-orientiertes Dienstmanagement übernehmen?
- Was für eine Art von Unterstützung sollte ein Tool für prozessorientiertes Dienstmanagement bieten? Welche Typen von Werkzeugen für IT-Dienstmanagement lassen sich festlegen? Wie ordnen sich klassische Typen von IT-Management-Werkzeugen im Vergleich hierzu ein?
- Welcher Typ von Werkzeug ist für die Unterstützung eines ITIL-Prozesses am geeignetsten? Fällt die Antwort auf diese Frage für alle ITIL-Prozesse gleich aus?
- Wie können auf Basis der ITIL-Prozessbeschreibungen Werkzeuge entwickelt werden? Kann dies modellbasiert angegangen werden und wie könnte ein entsprechender Ansatz aussehen? Welche Schwierigkeiten können sich in der praktischen Umsetzung ergeben?

## 1.3 Aufbau der Arbeit

Die Beantwortung dieser Fragen erfordert mehrere Schritte, die den weiteren Aufbau der Arbeit vorgeben. Welche grundlegenden Schritte dies sind und in welchem Kapitel der Arbeit die entsprechenden Ergebnisse dokumentiert sind, illustriert Abbildung 1.7.

### *Begriffsbildung*

Da viele zentrale Begriffe des Dienstmanagements in Literatur und Praxis unterschiedlich verwendet werden, folgt zunächst – soweit nicht bereits in Abschnitt 1.1 erfolgt – in Kapitel 2 die Klärung bzw. Festlegung einiger grundlegender Begriffe, die im Kontext dieser Arbeit wichtig sind.

### *Status Quo prozess- orientierter ITSM-Standards*

Ein fundiertes Konzept zur Tool-Unterstützung für die ITIL muss auch berücksichtigen, inwiefern sich Werkzeuge oder Werkzeugkonzepte anderer ITSM-Standards, gegebenenfalls mit Anpassungen, übernehmen ließen. Voraussetzung hierfür wäre aber eine vergleichsweise gute Abbildbarkeit dieser Standards auf die ITIL. Kapitel 3 stellt daher neben der ITIL auch verschiedene konkurrierende oder ergänzende Rahmenwerke für prozessorientiertes IT-Management vor (3.1 - 3.10) und untersucht, wie sich diese in den Kontext der grundlegenden Aufgabenbereiche im Dienstmanagement einordnen lassen (Abschnitt 3.11).

### *Klassifikation von Werkzeugen und Prozessen*

In Kapitel 4 folgt, nach einer Darlegung der Gründe, warum bisherigen Arbeiten zu ITIL-Werkzeuge noch einige Fragen offen lassen (Abschnitt 4.1), in Abschnitt 4.2 eine einfache Einteilung von IT-Management-Werkzeugen allgemein (Abschnitt 4.2.2) und Werkzeugen für die Unterstützung der Ausführung von

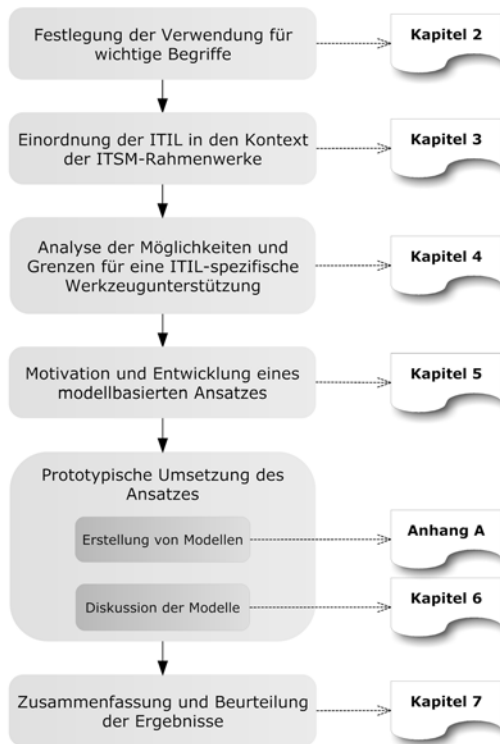


Abbildung 1.7: Aufbau der Arbeit

## 1 Einleitung

Dienstmanagementprozesse im besonderen (Abschnitt 4.2.3). Eine Klassifikation von ITIL-Prozessen unter Aspekten der Werkzeug-Unterstützung (Abschnitt 4.3) erlaubt schließlich die Identifikation der praktikablen, besonders lohnenswert erscheinenden Kombinationen von ITIL-Prozess und Werkzeugtyp.

- Modellbasierter Ansatz* Für eine entsprechend eingeschränkte und präzisierete Anwendungsdomäne wird in Kapitel 5 ein modellbasierter Ansatz vorgestellt. Dieser legt einen, auf dem *Zachman Framework* [Zac87] basierenden Ordnungsrahmen für die zu erstellenden Modelle und ein diesem entsprechendes Vorgehen fest.
- Prototypische Umsetzung* Kapitel 6 befasst sich mit der Diskussion von Herausforderungen, Designentscheidungen und gewonnenen Erkenntnissen bei der prototypischen Umsetzung dieses Ansatzes. Die entstandenen Modelle – für ausgewählte ITIL-Prozesse, eine plattformunabhängige Systemarchitektur sowie Aspekte einer beispielhaften plattformspezifischen Umsetzung der Architektur – sind in Anhang A dokumentiert.
- Abschluss* Kapitel 7 fasst die Ergebnisse der Arbeit zusammen, diskutiert Folgefragestellungen und gibt einen Ausblick auf mögliche zukünftige Forschungsvorhaben und Entwicklungen.

## 2 Begriffe und Konzepte

IT-Dienstmanagement ist ein sich zur Zeit schnell wandelndes und unübersichtliches Gebiet – das soll heißen, unter diesem Begriff werden viele Modelle, Methoden und Produkte publiziert. In den zahlreichen Arbeiten zu dem Thema finden sich dabei viele unterschiedliche Sichtweisen und Ansätze, dennoch kommt meist ein ähnliches Vokabular zur Anwendung. Konzepte und Begriffe sind daher häufig nur unscharf voneinander abgegrenzt und nicht einheitlich definiert. Aufbauend auf das vorherige Kapitel werden daher einige weitere, für die folgende Arbeit wichtige Begriffe und Konzepte erläutert und, wo unterschiedliche Auffassungen existieren, festgelegt, wie sie im Kontext der vorliegenden Arbeit weiter verwendet werden.

### 2.1 IT-Service

Die Frage was ein Dienst (*Service*) ist, wurde bereits in Abschnitt 1.1 angerissen. Im Folgenden werden noch einmal kurz einige unterschiedliche, in Literatur und Praxis zu findende Auffassungen dieses Begriffes dargelegt und die weitere Verwendung bestimmt.

Der Begriff *Service* findet auch in Kernbereichen der Informatik häufig Anwendung. Im Bereich von Netzwerktechnologien wird beispielsweise unter *Service* die Funktionalität verstanden, welche eine bestimmte Schicht (im Sinne des OSI-Referenzmodells) der über ihr liegenden Schicht am so genannten *Service Access Point* (SAP) zur Verfügung stellt – in diesem Kontext also meistens ein Daten-transportdienst [Bla92]. Im Bereich der allgemeineren IT-Architekturen, insbesondere bei der Diskussion um das Konzept *Service-Oriented-Architecture* (SOA) (siehe z.B. [KBS05]), existiert wiederum ein hiervon leicht abweichender Dienstbegriff, der auf der Festlegung von Kommunikationsschnittstellen für technische Dienste basiert. Auch wenn, ebenso wie für andere Begriffe im SOA-Bereich, hier keine einheitliche Definition für *Service* existiert [BD07], ist doch damit in aller Regel nicht die Interaktion zwischen unterschiedlichen Schichten (im Sinne des OSI-Referenzmodells) gemeint, sondern diejenige zwischen verschiedenen *Tiers* (im Sinne einer *Multi-Tier Architecture*, vgl. [Sun01]), also unterschiedlichen Applikationen. Beiden technischen Dienstbegriffen ist aber gemein, dass eine Funktionalität an einer wohldefinierten Schnittstelle zur Verfügung gestellt wird, welche von den Details der technischen Realisierung abstrahiert.

*Dienstbegriff in  
der klassischen  
Informatik*

### *Einschränkung der Betrachtung auf „Enterprise Services“*

In der vorliegenden Arbeit ist aber natürlich, wie bereits in Abschnitt 1.1 diskutiert, von Diensten im Sinne von IT-Services oder *Enterprise Services* die Rede – also komplexen Diensten, die nicht primär von Rechnern, sondern von Menschen im Unternehmensumfeld<sup>12</sup> genutzt werden. Gemessen an der Fülle von Literatur zum Thema IT-Dienstmanagement gibt es bemerkenswert wenige Versuche einer prägnanten Definitionen von *Service*. Die existierenden unterscheiden sich oberflächlich stark, zielen aber bei genauerer Betrachtung doch in eine ähnliche Richtung.

### *unterschiedliche Dienstbegriffe...*

So beschreibt die ITIL einen Dienst als „ein oder mehrere IT-Systeme, welche einen Geschäftsprozess ermöglichen“<sup>3</sup>. Im ITS-CMM (vgl. Abschnitt 3.4) heißt es hingegen: „Ein Service besteht aus einem Satz im Wesentlichen immaterieller Leistungen, die von einer Partei für eine andere Partei erbracht werden“<sup>4</sup>. Lewis bezeichnet, im Kontext einer unternehmensinternen Dienstleistung, einen Dienst als eine Funktion, welche die IT-Infrastruktur des Unternehmens für das Kerngeschäft erbringt<sup>5</sup>. Er fährt fort, dass ein Dienst sowohl als Abstraktion, aber auch als Epiphänomen<sup>6</sup> der IT-Infrastruktur gesehen werden kann. Man kann sagen, dass letztere Umschreibung recht gut eine häufig anzutreffende Sichtweise der IT-Organisation widerspiegelt (welche keine rechte Kenntnis davon hat, wofür die von ihr betriebenen Systeme tatsächlich nun genau verwendet werden) – erstere Umschreibung fasst die Sicht, welche der Kunde bzw. Anwender am liebsten einnehmen würde (in der er sich nicht mit technischen Details auseinander setzen muss), zusammen.

### *...spiegeln unterschiedliche Sichten wider*

Wie man sieht, spiegeln sich im Dienstbegriff oft die unterschiedlichen Sichtweisen der an Dienstleistung und Dienstnutzung beteiligten Parteien (siehe auch [Hup06]). Die Definition der ITIL basiert klar auf einer IT-Sicht, während die Definition im ITS-CMM eher einer Kundensicht entspricht und versucht den Begriff allgemein (nicht IT-spezifisch) festzulegen. Insgesamt erscheint die, wenn auch nicht sonderlich prägnante Begriffsbildung von Lewis am schlüssigsten. Sie lässt jedoch außer Acht, dass eine Erbringung eines IT-Dienstes in der Praxis nicht nur auf einer IT-Infrastruktur, sondern auch auf IT-Prozessen aufbaut (vgl. Diskussion in Abschnitt 1.1.2). Im konkreten Fall bestimmen allerdings letztlich Kunde

<sup>1</sup>Analog hierzu gibt es natürlich auch primär für Privatpersonen bzw. Konsumenten angebotene IT-Services, vgl. „Massendienste“ in [Sch01]. Diese werden aber in dieser Arbeit nicht weiter betrachtet.

<sup>2</sup>*Enterprise Service* wird hier im Sinne der IT-Dienste, die innerhalb eines Unternehmens erbracht werden, verstanden – nicht wie im gelegentlich auch gebrachten Sinne von Dienstes des Unternehmens (für andere Unternehmen oder Endkunden, vgl. z.B. [KBS05])

<sup>3</sup>Im Original: „One or more IT systems which enable a business process“ [OGC01]

<sup>4</sup>Im Original: „A service is an essentially intangible set of benefits provided by one party to another party“ [NCTvV05]

<sup>5</sup>Im Original: „A service is a function that the enterprise network provides for the business“ [Lew99]

<sup>6</sup>„Epiphänomen“ wird in diesem Zusammenhang als sekundäres Phänomen aufgefasst – also hier IT-Service als ein Effekt, der durch das Vorhandensein der IT-Infrastruktur ohne genaue Kenntnis der Wirkungskette entsteht.

und IT-Service-Provider (vgl. Abschnitt 2.2) im SLA, was der vereinbarte Dienst ist. Dabei ist der *Dienst* eigentlich ein immaterielles Hilfskonstrukt, eine Abstraktion, die weder der Domäne des Kunden noch der der IT-Organisation eindeutig angehört. Im Idealfall einigen sich dabei beide Parteien in der Dienstdefinition auf eine Abstraktion, welche der IT-Service-Provider auf seine IT-Betrieb- und Infrastruktur-orientierte Sichtweise abbilden kann – und der Kunde auf seine Sicht des eigenen Geschäftsprozesses (in die er den Dienst als Prozessressource integrieren kann).

## 2.2 IT-Service-Provider

Ein *IT-Service-Provider* (ITSP) ist im Kontext dieser Arbeit als die Organisation<sup>7</sup> zu verstehen, welche einen (in einem SLA definierten) IT-Dienst für einen Kunden erbringt. Die Dienstleistung erfolgt dabei nach einem „alles aus einer Hand“-Prinzip. Das bedeutet der IT-Service-Provider steht gegenüber dem Kunden in der Verantwortung für die Qualität des Dienstes mit allen seinen Komponenten, er hat also gegebenenfalls die Qualität von Sub-Diensten sicher zu stellen und kann dies nicht auf den Kunden abschieben (vgl. Diskussionen zu Geschäftsmodellen im Bereich *Application Service Provider* (ASP) [Apf00], [Fac02]).

„alles aus einer Hand“-Prinzip

Ein IT-Service-Provider und seine Kunden können Teil eines Unternehmens bzw. einer Institution sein (z.B. als unterschiedliche Abteilungen in einer Firma), aber auch organisatorisch in keinerlei Beziehung zueinander stehen (z.B. unterschiedlichen Firmen angehören), das heißt nur über das SLA als Geschäftspartner verbunden sein. Im ersteren Fall spricht man von einem internen Service-Provider, in letzterem von einem externen. Management und Provisioning von *Enterprise Services* obliegen in den meisten großen Unternehmen, trotz Auslagerung von immer mehr Infrastrukturdienstleistungen, meist immer noch unternehmenseigenen IT-Abteilungen [Eco03] [Cap07], also internen IT-Providern.

interne und externe IT-Service-Provider

In größeren Unternehmen werden dabei die IT-Systeme und Applikationen meist von einer spezialisierten IT-Entwicklungsabteilung, gegebenenfalls unter Rückgriff auf externe Entwickler, angefertigt. Diese Applikationen werden dann von einer organisatorisch getrennten Abteilung<sup>8</sup>, dem IT-Service-Provider (häufig IT-Operations genannt) als IT-Service betrieben. In größeren Umgebungen umfasst auch diese Betriebsorganisation wiederum spezialisierte Abteilungen oder Gruppen (siehe beispielhafte Darstellung in Abbildung 2.1).

Abgrenzung zur Entwicklung

<sup>7</sup> genau genommen die Rolle, welche die Organisation einnimmt – vgl. Abschnitt 2.9

<sup>8</sup> Genauer betrachtet existieren oft mehrere Betriebs- oder auch Entwicklungsabteilungen, die jedoch meist wie dargestellt organisatorisch zusammengefasst werden.

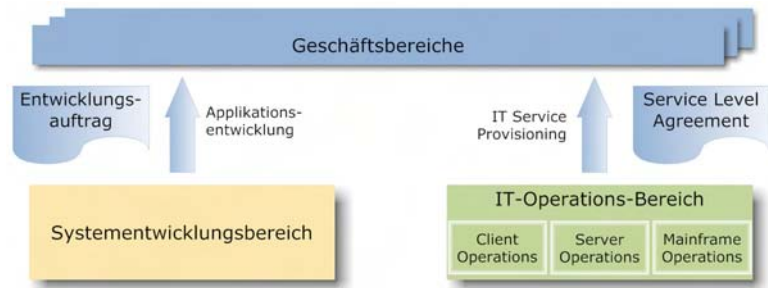


Abbildung 2.1: Klassische organisatorische Strukturierung unternehmensinterner IT

### 2.3 Prozess

#### BPR und TQM

Auch wenn das Konzept des Geschäftsprozesses selbst älter ist, so geht die jüngere Sicht auf Geschäftsprozesse und *Geschäftsprozessmanagement* (GPM) doch hauptsächlich auf die Popularisierung dieser Konzepte durch in den 1980er und frühen 1990er Jahren Verbreitung findenden Management-Techniken zurück. Diese, das *Total Quality Management* (TQM) (siehe z.B. [EDJ99]) und das *Business Process Reengineering* (BPR) (propagiert z.B. von Hammer [HC93, Ham96]) fanden zunächst vor allem in der produzierenden Industrie zur Verbesserung der Koordination von Geschäftsabläufen weite Verbreitung. Beiden liegt die grundsätzliche Idee der möglichst vollständigen Betrachtung und Steuerung von Abläufen in einer Wertschöpfungskette zugrunde. Dieses Konzept wurde um die Behandlung so genannter unterstützender Prozesse, d.h. Prozesse, die nicht der Wertschöpfung unmittelbar zuarbeiten, aber für deren effektive Aufrechterhaltung unerlässlich sind (z.B. Buchhaltung), erweitert. Das aus den USA stammende Konzept des BPR verfolgt bei der Verbesserung der Prozesse ein prinzipiell radikaleres Vorgehen mit der Analyse und dem kompletten Redesign von Unternehmensprozessen, während das in Japan entstandene TQM eher graduelle, aber beständige Verbesserungen anstrebt. Im folgenden soll aber das Prozessmanagement im Allgemeinen betrachtet werden und auf die Unterschiede und Vor- und Nachteile der Umsetzungsparadigmen von TQM und BPR nicht näher eingegangen werden.

#### Defintion Geschäftsprozess

Zwar existieren verschiedene Definitionen zum Begriff *Geschäftsprozess*, die gängigen unterscheiden sich aber nur in Nuancen voneinander - im Großen und Ganzen wird der Begriff des Geschäftsprozesses mittlerweile einheitlich verwendet. Ein Prozess ist nach DIN EN ISO 8402:1995-08<sup>9</sup>:

<sup>9</sup> mittlerweile abgelöst durch DIN EN ISO 9000:2000-12



„Ein Satz von in Wechselbeziehungen stehenden Mitteln und Tätigkeiten, die Eingaben in Ergebnisse umwandeln. Zu den Mitteln können Personal, Einrichtungen und Anlagen, Technologie und Methodologie gehören.“

Das britische *Office of Government Commerce* (OGC) definiert einen Prozess mit

„Eine verbundene Serie von Aktionen, Aktivitäten, Änderungen etc., ausgeführt von Agenten, mit der Absicht einen Zweck zu erfüllen oder ein Ziel zu erreichen.“<sup>10</sup>

ähnlich [OGC00], wenn hier auch, ebenso wie bei der Definition von Hammer<sup>11</sup>, mehr Augenmerk auf die Zielorientierung gelegt wird.

Häufig unterscheidet man zwischen so genannten Kernprozessen und den unterstützenden Supportprozessen. Als Kernprozesse bezeichnet man hierbei Prozesse, die direkt zur Wertschöpfung beitragen (also z.B. Produktion, Verkauf), als Supportprozesse jene, die für die Realisierung der Kernprozesse notwendig sind, jedoch keinen direkten Kunden- oder Produktbezug haben (z.B. Personalwesen, Buchhaltung) [BK03, Ham96]. Oft werden nur Prozesse, die der hier gegebenen Definition der Kernprozesse entsprechen, als Geschäftsprozesse bezeichnet und Supportprozesse schlicht als Prozesse [SS04]. Entsprechend kann man auch im ITSM zwischen Geschäftsprozessen und IT-Prozessen unterscheiden. Dies ist allerdings für allgemeine Diskussionen nicht ganz unproblematisch, hängt doch die Unterscheidung zwischen wertschöpfendem Geschäftsprozess und unterstützendem IT-Prozess dann vom organisatorischen Kontext ab:

*Kern- und Supportprozesse*

Handelt es sich aus Sicht der Kundenorganisation um einen internen IT Service Provider, d.h. wird zum Beispiel die interne IT eines Automobilbauers oder einer Versicherung betrachtet, dann sind im Regelfall<sup>12</sup> alle Dienstmanagementprozesse als Supportprozesse einzuordnen – schließlich tragen sie nur zu einem IT-Dienst bei, der wiederum nur eine Ressource für die eigentlichen Kernprozesse darstellt. Für einen externen IT-Service-Provider, für den die Bereitstellung (und Verrechnung) von IT-Diensten das Kerngeschäft darstellt, sind hingegen auch einige IT-Prozesse direkt produkt- und kundenorientiert, und somit, aus seiner Sicht, die Kern- bzw. Geschäftsprozesse.

*Zuordnung bisweilen von der Perspektive abhängig*

<sup>10</sup>Im Original: „A connected series of actions, activities, changes etc. performed by agents with the intent of satisfying a purpose or achieving a goal.“

<sup>11</sup>„Wir definieren einen Unternehmensprozess als Bündel von Aktivitäten, für das ein oder mehrere unterschiedliche Inputs benötigt werden und das für den Kunden ein Ergebnis von Wert erzeugt.“ [HC93]

<sup>12</sup>Dies gilt wenn die IT, wie bislang allgemein üblich, als so genanntes *Cost Center* betrieben wird. Wird die IT als so genanntes *Profit Center* betrieben, d.h. arbeitet sie unternehmerisch und stellt ihre Leistungen in Rechnung, so kann man sie im Rahmen dieser Diskussion wie einen externen Provider anzusehen.

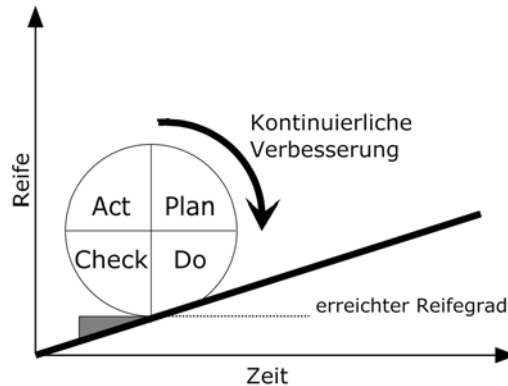


Abbildung 2.2: Qualitätskreis von Deming (Darstellung nach [OGC00])

## 2.4 Kontinuierliche Verbesserung

Eng verknüpft mit dem TQM ist die Idee des *Continuous Improvement* (Kontinuierliche Verbesserung). Im Zusammenhang mit Prozessmanagement besagt sie, dass es nicht ausreichend ist Prozesse einmalig zu definieren und in einer Organisation zu etablieren, sondern dass die Prozesse selbst einer kontinuierlichen Verbesserung zu unterziehen sind. Diese kontinuierliche Verbesserung wird in der Regel anhand eines Zyklus illustriert – die wohl älteste und am häufigsten zitierte Darstellung hiervon ist der in Abbildung 2.2 gezeigte *Deming Cycle*<sup>13,14</sup>.

### Deming Cycle

Die vier Schritte der kontinuierlichen Verbesserung nach dem *Deming Cycle* sind<sup>15</sup>:

**Plan** Verbesserungen am Prozess werden geplant, hierbei werden Anforderungen und Erwartungen an den verbesserten Prozess formuliert.

**Do** Der Prozess wird entsprechend der Planung umgesetzt (vorzugsweise pilotiert).

**Check** Der Prozess wird überprüft und gemessen, inwieweit Ablauf und Ergebnisse den Erwartungen und Anforderungen entsprechen (Soll-Ist-Abgleich).

<sup>13</sup>benannt nach seinem mutmaßlichen Urheber W. Edwards Deming – von diesem selbst meist als *Shewhart Cycle* bezeichnet, wobei offenbar nicht ganz geklärt ist dies aufgrund einer Art Widmung heraus geschah [PNC00] oder er tatsächlich *Shewhart* zugeschrieben werden muss [Wal98] [MD96]

<sup>14</sup>deutsch: „Deming-Kreis“, „Qualitätskreis von Deming“ [ITS04] oder „Deming-Zirkel“ [Hei05]

<sup>15</sup>Die Beschreibungen der einzelnen Schritte sind nicht genau festgelegt und variieren in der Literatur, vgl. z.B. [Wal98], [MD96] und [PNC00]

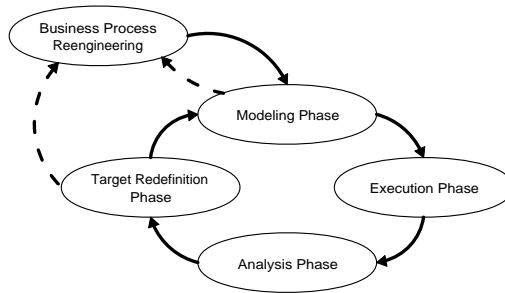


Abbildung 2.3: Phasen des kontinuierlichen Prozessmanagements nach [NPW03]

**Act** Basierend auf Analyse und Bewertung der Ergebnisse, Anpassung des Prozesses, gegebenenfalls Übergang von Pilotierung zu breiterer Implementierung, Dokumentation der Analyseergebnisse. (Danach wieder Beginn bei „Plan“)

Nach diesen vier iterativen Schritten *Plan*, *Do*, *Check*, *Act* (PDCA) wird der Deming-Kreis auch oft *PDCA Cycle*<sup>16</sup> genannt.

*PDCA Cycle*

Dieses grundlegende Konzept der zyklischen Prozessverbesserung hat viele Erweiterungen, Anpassungen und Neuinterpretationen erfahren. Es existieren in der Literatur eine Vielzahl von Modellen zur kontinuierlichen Verbesserung bzw. zu den verschiedenen Phasen des Prozessmanagements, die auf dem *Deming Cycle* basieren: beispielsweise der *Define, Measure, Analyze, Improve, Control* (DMAIC) Zyklus von *Six Sigma* [PNC00], das Modell von *Neumann, Probst, Wernsmann* zum kontinuierlichen Prozessmanagement [NPW03] (siehe Abbildung 2.3) oder der *Workflow Life Cycle* von *Borghoff, Schlichter* [BS00]. Diese Modelle beschreiben in der Regel mehr als vier Phasen – meist weil sie Elemente des *Deming Cycle* verfeinern oder weil sie im Gegensatz diesem eine Ausführungsphase beinhalten (wie die beiden letztgenannten).

*viele zyklische Modelle zum Prozessmanagement*

Hinsichtlich einer möglichst übersichtlichen Kategorisierung von IT-Dienstmanagement-Standards unter Werkzeugaspekten erscheinen aber die Phaseneinteilungen der Prozessmanagement-Modelle schon recht feingranular. Im Folgenden soll daher ein einfacheres Modell Verwendung finden, dargestellt in Abbildung 2.4. Es beschränkt sich auf drei Phasen *Planung*, *Ausführung* und *Analyse*, welche für die in späteren Kapiteln erfolgenden Erklärungen und Einordnungen hinreichend sind. In der Phase *Planung* wird der Prozess definiert, dessen Spezifikation in der Phase *Ausführung* zur Anwendung kommt. Die während der Prozessausführung gesammelten und zu KPIs verfeinerten Daten sind die Basis

*vereinfachtes Phasenmodell*

<sup>16</sup>Spätere Versionen von Deming ersetzen „Check“ mit „Study“, daher sieht man auch bisweilen die Bezeichnung *PDSA Cycle*

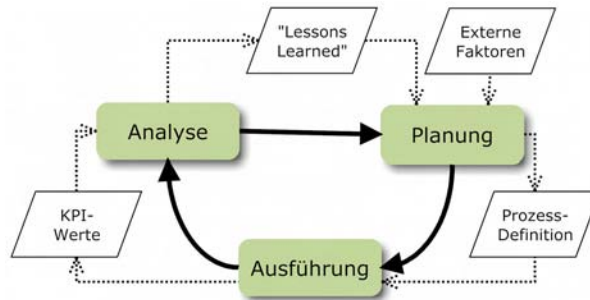


Abbildung 2.4: Vereinfachtes Lebenszyklusmodell des prozessorientierten ITSM

für die Phase *Analyse*. Die Ergebnisse der Analyse, also beispielsweise inwieweit die letzte vorgenommene Änderung am Prozess die erwünschten Resultate gebracht hat, bilden schließlich, gegebenenfalls zusammen mit Änderungen, die sich in Rahmenbedingungen ergeben haben, die Basis für eventuelle weitere Planungen zur Prozessanpassung oder -verbesserung. Das Modell ähnelt also dem oben vorgestellten von Neumann, Probst, Wernsmann, dessen Phasen *Modeling*, *Execution* und *Analysis* weitestgehend *Planung*, *Ausführung* und *Analyse* entsprechen. Ebenso wie dieses verzichtet es im Vergleich zum *Deming Cycle* auf die Umsetzungsphase, die, mit ihren speziellen Herausforderungen im Sinne eines organisatorischen Change-Managements (dass auch tatsächlich „gelebt“ wird, was geplant wurde), in dieser Arbeit nicht näher betrachtet werden soll.

## 2.5 Prozessmodellierung

Auch wenn sie nicht als solche erkannt und gemanagt werden – Geschäftsprozesse sind in einer Organisation praktisch immer zumindest rudimentär vorhanden, z.B. in Form sich wiederholender Aufgaben und Verfahren. Nach der Identifikation und Festlegung der zu managenden Prozesse (vgl. [BM03]) ist meist deren Modellierung der erste Schritt in einem Prozessmanagement-Projekt bzw. der Einstieg in ein Programm zur kontinuierlichen Verbesserung. Eine solche *Istmodellierung* kann beispielsweise der Verständnisbildung, der Identifizierung von Schwachstellen bzw. Verbesserungspotentialen oder als Basis für die spätere *Sollmodellierung* dienen [SL03].

### *Istmodellierung*

Aufbauend auf der durch die Istmodellierung ermöglichten Analyse des Status Quo ist der nächste Schritt die Sollmodellierung [SS03], in der ein Prozessmodell entworfen wird, das die identifizierten Schwachstellen und Verbesserungspotentiale adäquat adressiert.

Prozesse können auf verschiedene Weisen modelliert werden, d.h. es existiert eine Vielzahl von (teils grafischen) Sprachen zur Prozessmodellierung bzw. von Modelltypen (vgl. [RSD03]). Viele mit der Prozessmodellierung häufig in Verbindung gebrachten Modelltypen (z.B. Wertschöpfungsdiagramme) eignen sich aber nur um bestimmte Aspekte eines Prozesses darzustellen oder einen groben Überblick zu geben. Hier zeigen sich erhebliche Unterschiede in Abhängigkeit davon, an wen als Adressat bzw. Nutzer des Modells gedacht wird. Für Modelle, die durch Menschen genutzt werden sollen, kommen meist grafische Notationen zum Einsatz, die je nach genauere Adressatengruppe (externe Partner, höheres Management, Prozessmanager, Programmierer,...) dem Ersteller des Modells mehr oder weniger Detaillierung und formale Gestaltungsfreiräume erlauben. Prozessmodelle, die rechnergestützt weiterverarbeitet werden sollen, müssen hingegen in aller Regel hingegen formal spezifiziert werden, d.h. hier steht meist nicht die grafische Darstellung im Vordergrund, sondern eine textuelle, programmiersprachenähnliche.

*Adressaten des Modells*

Meist nimmt bei der Modellierung eines Prozesses die Darstellung des Kontrollflusses (welche Aktivität wird wann und unter welchen Bedingungen ausgeführt?), d.h. des *Workflow*, den zentralen Platz ein. Aber je nach Zielsetzung des Modells können auch andere Aspekte wie verwendete Ressourcen, benötigte und manipulierte Daten oder die beteiligten Akteure und Rollen (wer führt welche Aktivitäten aus, wer hat welche Verantwortung?) wichtig sein.

*verschiedene zu modellierende Aspekte*

Es existieren verschiedene Modellierungsmethoden, die versuchen Modelltypen für die Darstellung der verschiedenen wichtigen Aspekte eines Prozesses zur Verfügung zu stellen. Im deutschsprachigen Raum hierbei die *Architektur integrierter Informationssysteme* (ARIS) [Sch97] besonders verbreitet. ARIS zerlegt zur Reduzierung der Komplexität des einzelnen Modells das Gesamtmodell eines Prozesses in verschiedene Sichten, für die jeweils ein eigener Modelltyp bestimmt ist. Mittlerweile wird zunehmend auch der Einsatz der ursprünglich zur Beschreibung von Softwaresystemen entwickelten *Unified Modelling Language* (UML) zur Prozessmodellierung propagiert [OWS<sup>+</sup>03] (siehe auch Abschnitt 3.10). Auf Rahmenwerke zur Ordnung der verschiedenen Sichten und Modelltypen, die bei der Prozess- und Systemmodellierung möglich und sinnvoll sind, wird in Abschnitt 5.4.1 noch näher eingegangen.

*verbreitete Modellierungsmethoden*

## 2.6 Benchmarking

Das Erkennen der lohnenswertesten Verbesserungsmöglichkeiten und das Redesign von Geschäftsprozessen ist eine schwierige und aufwändige Aufgabe. Es liegt daher nahe sich, um schneller zu besseren Ergebnissen zu kommen, an bewährten Vorbildern, also beispielsweise gelungenen Umsetzungen ähnlicher Prozesse in anderen Umgebungen, zu orientieren.

- Benchmarking-partner** Ein solcher Vergleich mit anderen Organisationen ist die Kernidee des geschäftsprozessorientierten Benchmarking [Cam89], welches sich als im Zusammenhang mit TQM und BPR stehende Idee in den 1980er Jahren etablierte [BE94]. Man vergleicht eigene Methoden und Abläufe mit denen des *Benchmarking Partner* und verbessert daraufhin seine Prozesse gegebenenfalls nach dessen Vorbild.
- Arten des Benchmarking** Die grundlegenden Arten des Benchmarking werden nach **Camp** hauptsächlich durch die Beziehung zum *Benchmarking Partner* bestimmt [Cam89]. Ein *Benchmarking Partner* kann in der eigenen Firma oder Institution gefunden werden (*Internal Benchmarking*), ein direkter Konkurrent sein (*Competitive Benchmarking*) oder auch aus verwandten oder völlig anderen Branchen stammen (*Functional Benchmarking* und *Generic Benchmarking*). Bei allen drei letztgenannten Arten des externen Benchmarking ist das Ziel immer einen möglichst guten Partner (den *Best Competitor* oder *Functional Industry Leader*) zu finden, von dem man die *Industry Best Practices* lernen kann.
- Finden des besten Benchmarkingpartners häufig ein Stolperstein** Die Suche nach dem bestmöglichen *Benchmarking Partner* gestaltet sich aber häufig als schwierig und gilt als das Hauptproblem des Benchmarking. Beim *Competitive Benchmarking* besteht gerade bei den führenden Firmen meist eine geringe Neigung, die eigenen Geschäftsabläufe für Konkurrenten offen zu legen und damit eventuell hart erarbeitete Wettbewerbsvorteile quasi zu verschenken. Diese Hürde muss beim *Functional Benchmarking* und *Generic Benchmarking* in der Regel nicht überwunden werden, aber die Zahl der potentiellen Kandidaten ist weltweit unüberschaubar groß und es sind oft im eigenen Unternehmen nur wenig Kenntnisse über die spezifischen Stärken branchenfremder Firmen vorhanden. Dazu kommt, dass sich der entsprechende Kontakt oft schwer herstellen lässt, insbesondere dann, wenn zu dem als potentieller Partner erkannten Unternehmen keine Geschäftsbeziehungen bestehen oder es seinen Sitz in einer anderen Nation hat.

## 2.7 Best Practice

Aus Sicht der möglichen Schwierigkeiten bei der Suche nach dem besten *Benchmarking Partner* und des allgemein hohen Aufwandes, mit dem Benchmarkingprojekte verbunden sind, stellt sich die Frage, ob sich entsprechende Erkenntnisse – zumindest teilweise – effizienter gewinnen und einsetzen lassen. Benchmarking geht im Regelfall vom Informationsaustausch zwischen zwei (Teil-) Organisationen aus. Insbesondere beim *Functional Benchmarking* und *Generic Benchmarking* kann man aber davon ausgehen, dass viele der im Rahmen eines entsprechenden Projektes identifizierten *Industry Best Practices* nicht nur für den (einen) Benchmarkingpartner, sondern auch für eine Vielzahl anderer Organisationen von Interesse sind. Für eine weitere Verbreitung der erprobten Verfahrensweisen ist der bilaterale Informationsaustausch zwischen je zwei Organisationen offen-

sichtlich nicht immer der effizienteste Weg.

Praktiken und Verfahrensweisen, die allgemein als vorbildhaft gelten, können prinzipiell auch unabhängig eines bilateralen Benchmarkings dokumentiert werden. Dieser Gedanke liegt dem Konzept der *Best Practices Database* zugrunde [All97] – für den Rückgriff auf das in diesen dokumentierte Wissen ist also nicht die direkte Beteiligung an einem Benchmarkingprojekt erforderlich. Eine frühe Umsetzung dieser Idee ist beispielsweise die so genannte *Global Best Practices Database* [HKK98], in der die Consulting-Firma Arthur Andersen<sup>17</sup> ab Ende 1990 Praktiken und Geschäftsprozesse in bestimmten Aspekten besonders erfolgreicher Firmen zu klassifizieren und dokumentieren begann [All97].

*Best Practices  
Databases*

Der Begriff Best Practice<sup>18</sup> als Kurzform von *Industry Best Practices* stammt also aus dem Benchmarking-Umfeld, wird aber mittlerweile auch außerhalb dieses Kontextes für allgemein als vorbildhaft angesehene Verfahrensweisen verwendet. Die Verwendung des Begriffes Best Practice, also „beste Praxis“ bzw. „beste Verfahrensweise“, hierfür ist natürlich eher optimistisch. Weder bei der Auswahl eines Benchmarkingpartners noch beim Aufbau einer *Best Practices Database* können alle weltweit prinzipiell in Frage kommenden Organisationen untersucht werden. Zudem handelt es sich bei *Industry Best Practices* um bewusst mit einem Benchmarkingpartner geteilte oder in einer *Best Practices Database* publizierte Erkenntnisse – im Zusammenhang mit dem Kerngeschäft stehende und einen Wettbewerbsvorteil sichernde, spezielle Verfahrensweisen werden aber meist zu den Geschäftsgeheimnissen zählen und nur selten ohne Weiteres preisgegeben werden. Auch könnten Innovationen, die über das in der Praxis bekannte hinausgehen, noch zu Verbesserungen führen. Was allgemein als Best Practice bezeichnet wird, ist also unter Umständen nicht einmal das weltweit beste Praxiswissen – und schon gar nicht das theoretische Optimum. Best Practice sind also immer nur die besten – bzw. die allgemein als am besten angesehenen – unter den bekannten und untersuchten Verfahrensweisen. Ein singulärer Wettbewerbsvorteil (gegenüber allen im selben Umfeld agierenden Organisationen) kann durch die Adaption von Best Practice daher offensichtlich nicht erreicht werden.

*allgemein als  
vorbildhaft  
geltende Verfah-  
rensweisen*

Dies führt bisweilen zu abweichenden Definitionen. Beispielsweise sehen Becker et. al. [BADK02] innovative und theoriebasierte Elemente, welche genau so einen Wettbewerbsvorteil sicherstellen, als notwendigen Bestandteil eines *Best-Practice-Modells* an (vgl. die Diskussion um ITIL als Best Practice, siehe Abschnitt 3.1.1). In diesem Sinne könnte es allerdings kaum publizierte Best Practice geben. Für das, was gemeinhin als Best Practice bezeichnet wird, in der Praxis bewährte Verfahrensweisen, wird dort der Begriff *Common Practice* vorgeschlagen. Allerdings trifft dieser Begriff der „gewöhnlichen Praxis“ das Konzept

*Common  
Practice und  
Good Practice*

<sup>17</sup>der Consulting-Arm von Arthur Andersen wurde in Andersen Consulting umbenannt, welches nach einer durch den Enron-Skandal angestoßenen Welle von Umstrukturierungen mittlerweile in Accenture und verschiedenen anderen Consulting-Firmen aufgegangen ist.

<sup>18</sup>Singular und Plural (*Best Practices*) werden in der selben Bedeutung verwendet

wohl auch nicht besser – was im Wortsinne gewöhnlicherweise praktiziert wird, ist häufig vom allgemein als vorbildhaft angesehenen noch weit entfernt – und würde sich per Definition kaum umzusetzen lohnen (es ist ja gewöhnlich schon implementiert). Im englischsprachigen Raum wird bisweilen im Wechsel mit Best Practice der Begriff *Good Practice* (siehe z.B. [Sco03]) verwendet um auf praktischen Erfahrungen basierende Empfehlungen zu bezeichnen – was das Konzept vom Wortsinn her wohl am besten trifft.

*Good Practice  
und Best  
Practice sind  
synonym*

Es scheint allerdings wenig lohnend für den Begriff Best Practice, dessen Bedeutung sich im angelsächsischen Raum seit mehr einem Jahrzehnt etabliert hat und dort weitgehend einheitlich verwendet wird, eine Neudeutung auf Basis der Semantik einzelner Begriffsbestandteile zu versuchen oder umgekehrt zu versuchen, einen neuen Namen für das dahinter liegende Konzept zu etablieren. Die Begriffe Best Practice, *Industry Best Practices* und *Good Practice* stehen alle für *bekannte, als vorbildhaft akzeptierte Verfahrensweisen* und sind als synonym anzusehen.

### 2.8 Prozesskontrolle

*KPIs als  
Datenbasis für  
kontinuierlichen  
Verbesserung*

Beim Managements von Prozessen, also auch ITSM-Prozessen, ist häufig von Parametern zur Prozesskontrolle die Rede. Auch hier existieren keine präzisen und gleichzeitig universell anerkannte Begriffsdefinitionen. Der wohl am häufigste gebrauchte Ausdruck in diesem Zusammenhang ist allerdings wohl der des *Key Performance Indicator* (KPI). Im Großen und Ganzen handelt es bei KPIs im Kontext des Dienstmanagements letztlich um nichts anderes als die in Abschnitt 1.1.2 bereits kurz diskutierten ITSM-Kennzahlen (vgl. Abbildung 1.5 auf Seite 9). Die KPIs werden vor allem im Kontext der kontinuierlichen Verbesserung (Abschnitt 2.4) verwendet um dem *Process Owner* (vgl. Abschnitt 2.9) aussagekräftige Daten zum Prozess an die Hand zu geben, auf deren Grundlage er fundierte Entscheidungen hinsichtlich zukünftiger Schritte zur Prozessoptimierung treffen kann (siehe Abbildung 2.5).

*typische KPIs*

Für jeden definierten Prozess werden in der Regel auch KPIs festgelegt, welche die Qualität und andere wichtige Eigenschaften der Prozessimplementierung widerspiegeln sollen. KPIs sind häufig als Verhältnisse von einfacheren Parametern definiert oder als ein über einen bestimmten Zeitraum gemittelter Durchschnittswert. Eine typische mittels eines KPI kontrollierte Eigenschaft ist zum Beispiel, wie häufig ein Prozess durchgeführt wird, also wie viele Fälle (*Cases*) beziehungsweise Prozessinstanzen in einem Zeitraum auftreten – im Falle des *Incident Management* könnte dieser KPI als „Zahl der neu erfassten *Incidents* pro Monat“ definiert sein. Ebenfalls beinahe immer als KPI definiert ist die durchschnittliche Durchlaufzeit eines Prozesses – im Falle des *Incident Management* könnte dies als die während des Beobachtungszeitraumes (z.B. Monat) im Durchschnitt be-



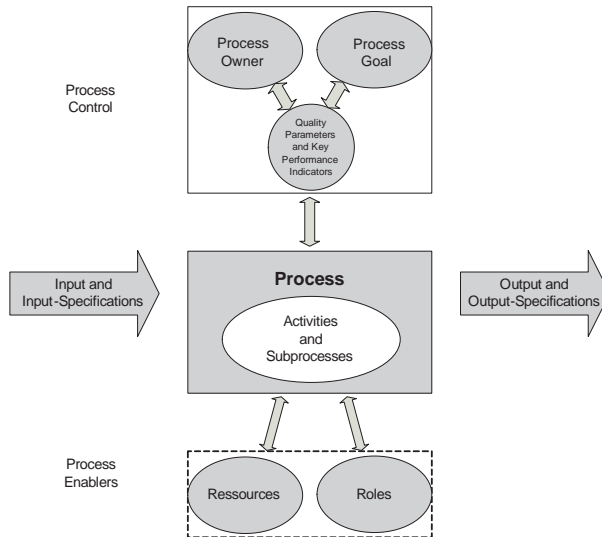


Abbildung 2.5: Allgemeines Modell des Prozessmanagements (nach [OGC00])

nötigte Zeit für die Bearbeitung eines *Incidents* (von der Neuerfassung bis zum Abschluss) festgelegt sein (vgl. auch Abschnitt 4.3.2).

Es ist allerdings wichtig, KPIs nicht, wie häufig geschehen, mit Service-Parametern zu verwechseln. Der Zweck der KPIs liegt primär in ihrer Verwendung im Provider-internen Management, analog zu den Komponentenparametern. Zwar können in einigen Fällen die Definitionen von KPIs mit denen von im SLA festgelegten Service-Parametern sich ähneln oder sogar übereinstimmen – beispielsweise eine im SLA festgelegte maximale durchschnittliche Wiederherstellungszeit mit der Durchlaufzeit des *Incident Management*. Dies gilt jedoch nicht für den allgemeinen Fall. So lässt sich aus der Fallfrequenz des *Incident Management* keine Aussage über die Qualität des Prozesses und schon gar nicht über die Dienstqualität ableiten. Aber auch für die Prozessqualität relevante Parameter haben nicht automatisch Einfluss auf die Qualität des Dienstes. Beispielsweise ist für das *Incident Management* in Fällen, in denen die meisten Anfragen und Störungsmeldungen nicht telefonisch (sondern über E-Mail oder spezielle Support-Systeme) eingehen, die Erstlösungsrate ein gutes Maß für die Effizienz des Prozesses, aber kein gutes Maß für die Dienstqualität (da es dem Nutzer, wenn die Störung schnell behoben wird, letztendlich egal sein kann, ob daran nur ein oder mehrere Support-Mitarbeiter beteiligt waren). Wenn, wie häufig zu beobachten, statt sinnvoller Service-Parameter (z.B. durchschnittliche Wiederherstellungszeiten, „pick-up-times“ und Quoten für sofortige Bearbeitung ggf.

*Service-Parameter*  
≠ KPI

für telefonische Anfragen) solche KPIs unreflektiert in SLAs vereinbart werden, bringt es für Kunden und Nutzer keinen Vorteil, schränkt aber den IT-Service-Provider in seinen Möglichkeiten der Gestaltung des Betriebs unnötig ein<sup>19</sup>.

### 2.9 Rollen

#### *grundlegende Rollen im ITSM*

Wo immer das IT-Dienstmanagement die rein technische Ebene verlässt, also auch über Personen und Organisationen gesprochen werden muss, ist meistens auch von Rollen die Rede. Allgemein werden im Kontext von *Enterprise Services*, neben dem IT-Service-Provider (vgl. Abschnitt 2.2), die grundlegenden Rollen<sup>20</sup> *User* (Nutzer) und *Customer* (Kunde) unterschieden. Der Kunde ist dabei der Vertragspartner des IT-Service-Provider, während der Nutzer derjenige ist, der den Dienst auf einer alltäglichen Basis für seine Arbeit verwendet. Die Abgrenzung der Rollen wird beispielsweise in der ITIL [OGC00] oder im MNM-Dienstmodell [GHK<sup>+</sup>01] (siehe Abbildung 2.6) so vollzogen. Im Bereich der Telekommunikationsdienste wird die Unterscheidung zwischen Kunde und Anwender nicht immer getroffen (z.B. in der eTOM, vgl. Abschnitt 3.2), wohl da hier schwerpunktmäßig eher Massendienste (vgl. [Sch01]) betrachtet werden.

#### *Definition Rolle*

Auch wenn für den Begriff *Rolle* ebenfalls keine griffige und allgemein anerkannte Definition existiert, so hat sich doch ein weitgehender Konsens über seine Bedeutung entwickelt, der in verschiedenen Modellen zum Ausdruck kommt. Abbildung 2.7 stellt ein solches, UML-basiertes Modell dar, welches Elemente von *Fowlers Party Pattern*<sup>21</sup> [Fow97a] und seinen Überlegungen zu Rollen [Fow97b] zusammenfasst und erweitert.

#### *Zuordnung zu Parteien*

Als Oberklasse von Person, Organisation oder einer Gruppe innerhalb einer Organisation (z.B. ein Team), gilt die Klasse *Party* (Partei). Eine Rolle kann von einer Partei ausgefüllt werden, also beispielsweise die Rolle „Kunde“ durch eine Organisation oder eine Person. Das Einnehmen der Rolle geht in der Regel mit bestimmten Rechten und Pflichten einher, beispielsweise bezüglich des Zugriffs auf Management-Funktionalität des Dienstes. Die Zuordnung einer Partei zu einer Rolle ist immer an einen bestimmten Kontext gebunden, d.h. beispielsweise eine Organisation, welche die Rolle *Kunde* im Kontext einer Service-Provisioning-Beziehung einnimmt, kann gleichzeitig die Rolle *Service-Provider* im Kontext einer anderen Service-Provisioning-Beziehung einnehmen, vgl. MNM-Dienstmodell [GHK<sup>+</sup>01] (siehe auch Abschnitt 3.10 und insbesondere Abbildung 2.6 auf der nächsten Seite).

---

<sup>19</sup>Dies gilt natürlich analog auch für im SLA niedergelegte Vereinbarungen zu technischen Randbedingungen, welche keine direkte Relevanz für die Dienstnutzung haben (beispielsweise Festlegung der Nutzung spezifischer, mutmaßlich zuverlässiger Komponenten statt Vereinbarung von Verfügbarkeitszielen).

<sup>20</sup>vgl. Abschnitt 2.9

<sup>21</sup>Ein Modell zu „Party“ und findet sich auch in SID [TMF04e], vgl. Abschnitt 3.2.

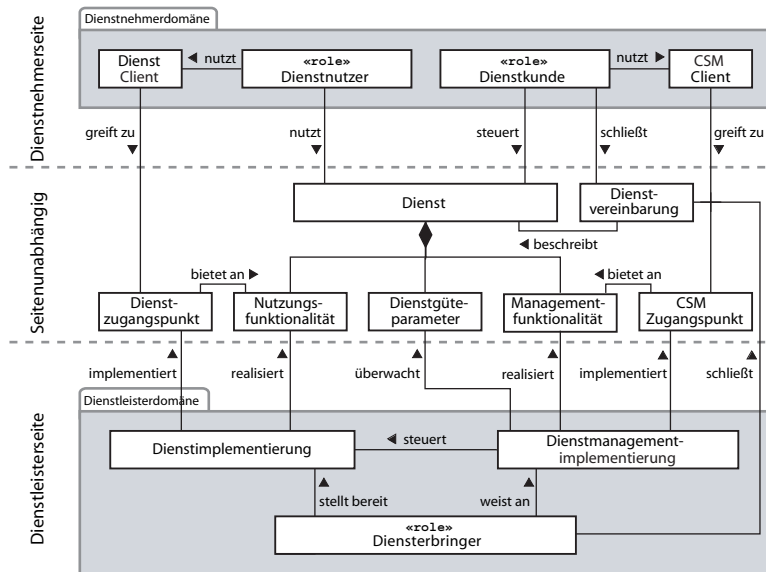


Abbildung 2.6: MNM-Dienstmodell

Auch im Prozessmanagement spielen Rollen häufig eine große Rolle, erlauben sie doch eine Zuteilung von Rechten und Pflichten (im Kontext eines Prozesses) ohne diese notwendigerweise fest auf Stellen in der Aufbauorganisation abbilden zu müssen. Hinsichtlich grundlegender, also für jeden Prozess existierender Rollen herrscht in der Literatur keine einheitliche Meinung. In der Regel werden nur für den Gesamtprozess in irgendeiner Weise verantwortliche Rollen festgelegt, am häufigsten werden hierbei die Begriffe *Process Owner* (Prozesseigentümer) und *Process Manager* (Prozessmanager) genannt (z.B. [SH05]) – es existieren aber zahlreiche weitere englische Begriffe wie *Process Responsible*[NPW03], *Process Executive*[McC01] und so weiter. Am ehesten scheint aber Konsens zu sein,

*Rollen im  
Prozessmanagement*

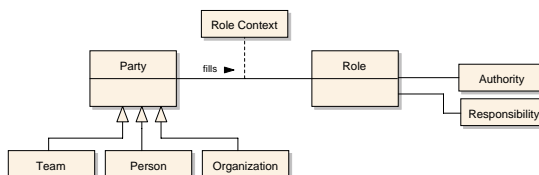


Abbildung 2.7: Rollen und Parteien

dass der Prozesseigentümer dem Prozessmanager vorgesetzt ist (z.B. [EW03] [SH05]). Er ist der Gesamtverantwortliche für den Prozess und ernennt in der Regel den Prozessmanager. Der Prozessmanager ist der Ausführungsverantwortliche, er setzt die Pläne des Prozesseigentümers um. Der Prozessmanager kann dabei unter gewissen Umständen auch aktiv an der Ausführung des Prozess beteiligt sein (z.B. zur Behandlung von wichtigen Ausnahmefällen), was auf den Prozesseigentümer nicht zutrifft. Allerdings kann durchaus argumentiert werden, dass, außer in vielleicht sehr großen Organisationen, diese Rollen zusammenfallen und in der Praxis all diese Begriffe im Großen und Ganzen das gleiche bedeuten (vgl. [Har91]).

### 3 ITIL im Kontext der ITSM-Rahmenwerke

Neben der ITIL existieren zahlreiche weitere Rahmenwerke, die mit prozessorientiertem IT-Dienstmanagement in Zusammenhang gebracht werden. Es ist allerdings nicht immer klar, wo diese in Bezug auf Methodik, berücksichtigte Management-Aspekte oder Inhalt im Vergleich zur ITIL stehen – Eine übergreifende Einordnung von Rahmenwerken des Dienstmanagements gibt es bislang nicht. Bisherige Vergleiche beschränken sich entweder auf die Betrachtung von nur zwei Standards oder erschöpfen sich in einer reinen Aufzählung der Rahmenwerke, Wiedergabe der proklamierten allgemeinen Zielsetzung und Nennung der Herausgeber bzw. Eigentümer.

Im Folgenden wird daher eine entsprechende Einordnung entwickelt, die es erlaubt die Schwerpunkte und Zielsetzungen der verschiedenen Rahmenwerke miteinander in Beziehung zu setzen (die gefundene Einordnung findet sich in Abschnitt 3.11). Die Platzierung der verschiedenen Standards des Dienstmanagements in dieses Schema erlaubt die Bestimmung, in welcher Nähe sie – nicht in Inhalt und Begrifflichkeit, wohl aber in Zielsetzung und grundlegender Methodik – zur ITIL stehen. So kann abgeschätzt werden, wo sich prinzipiell eine Wiederverwendung von Inhalten oder Methodik anderer Standards des Dienstmanagements realisieren lassen könnte.

Die nächsten Abschnitte stellen, nach einer kurzen Einführung in die ITIL (Abschnitt 3.1), zunächst die wichtigsten Rahmenwerke in diesem Feld vor. Hierzu gehört primär die eTOM (Abschnitt 3.2), ein umfang- und einflussreiches Framework, welches unabhängig von der ITIL entstanden ist, ihr aber in der erklärten Zielsetzung recht ähnlich ist. Betrachtet werden darauf folgend die für das *Software Engineering* entwickelten Bewertungsmethoden CMM, CMMI und SPICE (Abschnitt 3.3), welche großen Einfluss auf die Entwicklung anderer prozessorientierter IT-Management-Standards ausgeübt haben. In Abschnitt 3.4 wird das auf ITIL- und CMM-Konzepte verbindende ITS-CMM eingegangen, dann auf die ebenfalls in einigen Aspekten an das CMM angelehnten Standards Co-bit (Abschnitt 3.5) und eSCM (Abschnitt 3.6). Anschließend werden die ITIL-Adaption und -Erweiterungen der Firma Microsoft (Abschnitt 3.7) und IBM (Abschnitt 3.8), sowie der ITIL-basierte Standard ISO/IEC 20000 (Abschnitt 3.9) vorgestellt. Nach einem kurzen Überblick über weitere in diesem Zusammenhang relevanten Modelle und Rahmenwerke (Abschnitt 3.10) folgen in Abschnitt 3.11 schließlich die Vorstellung des Einordnungsschemas, eine Platzierung der vorgenannten Standards und eine dieses Kapitel abrundende Diskussion.

## 3.1 ITIL

Die ITIL ist wohl das Rahmenwerk, welches im Zusammenhang mit Dienstmanagement am häufigsten genannt, bisweilen sogar mit diesem Begriff gleich gesetzt wird. Dabei spielen allerdings nicht alle der mittlerweile recht zahlreichen ITIL-Publikationen eine gleich gewichtige Rolle. Im folgenden werden Herkunft, Zielsetzung (Abschnitt 3.1.1) sowie die Struktur und die grundlegenden Inhalte (Abschnitte 3.1.2 und 3.1.3) der ITIL vorgestellt.

### 3.1.1 Herkunft und Zielsetzung

*Ursprung im  
Management  
von  
Rechenzentren  
Erste Version  
der ITIL*

Die *IT Infrastructure Library* (ITIL) entstand in den 1980er Jahren im Umfeld durch staatlicher Rechenzentren in Großbritannien, als die *Central Computer and Telecommunications Agency* (CCTA) begann Erfahrungen zum IT-Management zu sammeln und in einem *Code of Practices* zu dokumentieren. Zunächst bestand ITIL aus zehn *Core Titles*, von denen jedes ein so genanntes Modul, d.h. die Beschreibung eines IT-Management-Prozesses nebst Umsetzungsempfehlungen enthielt. Diese zehn Module waren in die zwei Sets *Service Support* und *Service Delivery* gegliedert. Bis Ende der 90er Jahre wurde die ITIL immer weiter ergänzt, so dass am Ende über 40 veröffentlichte Bände in neun Sets zusammengefasst waren. Davon befassten sich, neben den beiden bereits erwähnten Kernsets, noch das *Manager's Set*, *Software Support Set*, *Networks Set* und das *Computer Operations Set* mit der Erbringung von IT-Diensten und dem klassischen Netz- und Systemmanagement. Die übrigen drei beschäftigten sich mit unterstützenden physischen Infrastrukturen, wie Gebäuden, Verkabelung und so weiter [HAN99].

*aktuelle Version*

Diese erste Version der ITIL ist zwischen 1999 und 2006 grundlegend überarbeitet worden. Die bisherigen, sich mit IT-Diensten befassenden Module wurden aktualisiert, ergänzt und zusammen mit neuen Modulen in insgesamt sieben Bänden zusammengefasst. Diese Bände haben jeweils, mit Ausnahme des Security Managements, in etwa den Umfang eines bisherigen Sets. Diese Bände sind:

- *Service Support* [OGC00]
- *Service Delivery* [OGC01]
- *Security Management* [OGC99]
- *Planning to Implement Service Management* [OGC02c]
- *ICT Infrastructure Management* [OGC02b]
- *Application Management* [OGC02a]
- *The Business Perspective* [OGC04]

## Kern der ITIL

Diese sieben Bände werden als die *Core Guidance* der ITIL bezeichnet. Im Vergleich zur ersten Version, in welcher ja nur die Sets *Service Support* und *Service Delivery* als der Kern bezeichnet wurden, hat sich also der Fokus erheblich erweitert. Allerdings werden in der Praxis immer noch die Inhalte dieser beiden ersten Sets (d.h. jetzt die Bände *Service Support* [OGC00] und *Service Delivery* [OGC01]) als Kern der ITIL angesehen und ihr Inhalt häufig mit „Service-Management nach ITIL“ gleichgesetzt. So beschränkt sich das vom *Office of Government Commerce* (OGC), dem *IT Service Management Forum* (ITSMF)<sup>1</sup> sowie den Zertifizierern *Information Systems Examination Board* (ISEB)<sup>2</sup> und *Examination Institutein* (EXIN) gemeinsam festgelegte Curriculum für die im ITIL-Umfeld recht populären ITSM-Zertifikate<sup>3</sup> auf Kenntnisse zu *Service Support* und *Service Delivery*.

## ergänzende Publikationen

Es gibt aber auch noch weitere Bücher, die unter dem ITIL-Label erscheinen. Diese Bücher, *Software Asset Management* [OGC03] und *ITIL Small-scale Implementation* [OGC06a], stellen die *Complementary Guidance*, also Ergänzungen zur Kern-ITIL dar. Hierzu gehört mittlerweile auch eine Version eines aus dem Umfeld des ITSMF stammenden einführenden Buches [vB05], das nun als *Introduction to ITIL* [OGC05a] parallel auch von der OGC herausgegeben wird.

Die aktuelle Fassung der ITIL, bisweilen *Version 2.0* genannt, löst die ursprüngliche Version ab, das bedeutet die alten Bücher gelten als obsolet und werden nicht mehr verlegt. Allerdings werden sie noch als *ITIL Back Catalogue* elektronisch vertrieben<sup>4</sup>, wohl auch weil sich für einzelne der von *ITIL Version 1* abgedeckten Themen keine Entsprechung in der aktuellen Version findet. Eigentümer der aktuellen Version der ITIL ist das OGC, in dem die CCTA 2001 aufgegangen ist.

## Zielsetzung

Das Ziel bei der Schöpfung der ITIL war, ein umfassendes Rahmenwerk für ein kundenorientiertes und kosteneffektives Management von IT-Diensten zu schaffen. Dabei will man den Anforderungen aller mit IT-Dienstleistungen befassten Organisationen, gleich ob der öffentlichen Hand oder der Privatwirtschaft zugehörig, gleich ob mit nur einer Handvoll oder tausenden IT-Mitarbeiter, gerecht werden. Die ITIL dokumentiert hierfür, nach eigener Aussage, die *Best Practice* im IT-Dienstmanagement. Sie hat damit den Anspruch nicht theoretisch erarbeitete Normen zu setzen, sondern vielmehr die jeweils fortschrittlichsten, erprobten Verfahrensweisen im IT-Dienstmanagement zu beschreiben. Diese Praxisorientierung soll unter anderem durch die enge Kooperation des ITSMF, eines Zusammenschlusses von ITIL-Anwendern, gewährleistet werden.

Um die von der ITIL reklamierte Eigenschaft *Best Practice* zu dokumentieren

<sup>1</sup>häufig auch „itSMF“ geschrieben, [www.itsmf.com](http://www.itsmf.com)

<sup>2</sup>Teil der *British Computer Society* (BCS)

<sup>3</sup>Es werden die Zertifikate *ITIL Foundation*, verschiedene Varianten des *ITIL Practitioner* sowie das *Manager's Certificate in IT Service Management* angeboten [EXI07]. Letzteres wird oft auch als *Service Manager Certificate* bezeichnet.

<sup>4</sup><http://www.tso.co.uk/ITIL/>

*ITIL als Best Practice* gibt es allerdings auch Missverständnisse, die wohl auf Unterschieden in der Verwendung dieses Begriffes zwischen dem deutschen und dem angelsächsischen Sprachraum zurückzuführen sind. So wird bei **Hochstein, Zarnekow, Brenner** [HZB04, HZB05] ausgesagt, dass es sich bei ITIL nicht um *Best Practice*, sondern vielmehr um *Common Practice* handele. Dies geschieht allerdings auf Basis von entsprechenden Definitionen in **Becker et. al.** [BADK02] die, außerhalb des akademischen Umfeldes im deutschsprachigen Raum, kaum Verwendung finden und dem allgemeinen internationalen Sprachgebrauch nicht entsprechen. Man kann die Inhalte der ITIL also durchaus als *Best Practice*, im Sinne der allgemein etablierten Verwendung dieses Begriffes (vgl. Abschnitt 2.7) verstehen.

*Public Domain Framework* Auch der Begriff *Public Domain Framework* in der Selbstdarstellung der ITIL bedarf einer Klarstellung: Er bezieht sich darauf, dass die in ITIL dokumentierten Praktiken uneingeschränkt (also u.A. lizenzfrei) eingesetzt werden können. Die Bücher selbst, aus denen die ITIL besteht, stehen jedoch unter Copyright<sup>5</sup>.

Die weitere Beschreibung von Struktur und Inhalt der ITIL beschränkt sich auf die Dienstmanagement-relevanten Kernbereiche *Service Support* und *Service Delivery*.

#### 3.1.2 Struktur

*pro Prozess ein Kapitel* Das Dienstmanagement wird innerhalb der ITIL in den beiden Bänden *Service Support* und *Service Delivery* adressiert. Beide enthalten, neben allgemeinen Ausführungen (zu Einführung von IT-Dienstmanagement usw.), die Beschreibungen von jeweils fünf Dienstmanagementprozesse, wobei jedem Prozess ein eigenes Kapitel gewidmet ist. In *Service Support* befasst sich zusätzlich ein Kapitel mit dem *Service Desk*, welches keinen Prozess, sondern eine *Function* darstellt. Der Begriff *Function* ist hierbei im Sinne von „Business Function“ zu verstehen, was im Deutschen in etwa einer Organisationseinheit (in einer funktional gegliederten Organisation) oder einer Stelle in kleineren Organisationen entspricht.

*Aufteilung in Service Support und Service Delivery* Die Aufteilung der Kapitel auf die beiden Bände ist in Abbildung 3.1 dargestellt. Die in *Service Delivery* definierten Prozesse befassen sich dabei mit mittelfristigen Gestaltungsaufgaben, haben also in der Regel einen Zeithorizont von einigen Monaten bis wenigen Jahren, während die Prozesse des *Service Support* sich tendenziell mehr mit der Steuerung des täglichen Betriebes befassen. Man spricht in diesem Zusammenhang auch von den *operativen* Prozessen im *Service Support* und den *taktischen* Prozessen der *Service Delivery* [Vog00].

*Anwender und Kunden* Das *Service Desk* stellt dabei den *Single Point of Contact* (SPOC), das heißt die alleinige Schnittstelle des IT-Service-Provider für die Anwender bereit. Es übernimmt aber auch die Kontrolle und weitgehende Ausführung des Prozesses

<sup>5</sup> *Crown Copyright*, siehe [www.itil.co.uk/faqs.htm](http://www.itil.co.uk/faqs.htm)



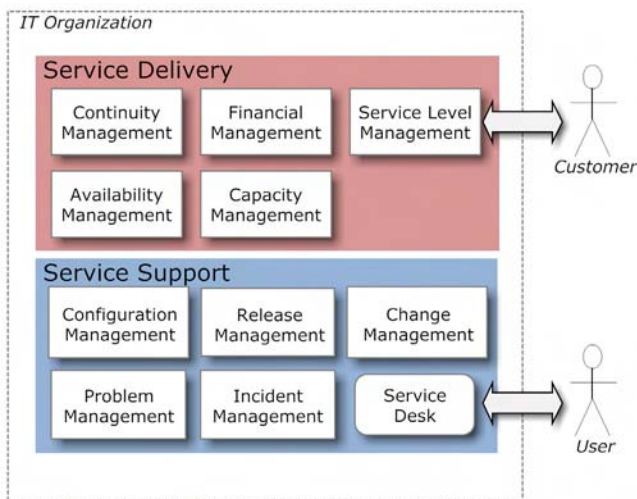


Abbildung 3.1: ITIL Kernprozesse

*Incident Management.* Die Vereinbarung von SLAs, also die Interaktion mit dem Kunden, obliegt in der ITIL hingegen dem *Service Level Management*. Die restlichen Prozesse haben im Regelfall<sup>6</sup> weder mit Anwendern, noch mit Kunden eine direkte Interaktion.

Die Kapitel variieren stark in Umfang (von 24 bis über 81 Seiten) und Struktur. Im *Service Support* ähneln sich immerhin die Prozessbeschreibungen und bestimmte Elemente tauchen in der Gliederung jedes entsprechenden Kapitels auf. So gibt unter anderem zu allen Prozessen jeweils einen Abschnitt zu Zielsetzung, Aktivitäten, Rollen und Prozesskontrolle<sup>7</sup>. In der *Service Delivery* sind die Parallelen in der Strukturierung zwischen den Prozessen allerdings deutlich geringer ausgeprägt. Die Gliederungen der Kapitel unterscheiden sich bis auf wenige Elemente stark und folgen weitgehend nicht dem in *Service Support* vorhandenen Aufbau – so finden sich beispielsweise Beschreibungen der Aktivitäten, außer im Kapitel zu *Capacity Management*, nicht in einem entsprechend benannten Hauptabschnitt.

*Kapitelaufbau unterschiedlich*

<sup>6</sup>Die wichtigste Ausnahme stellt die Beteiligung von Vertretern dieser Gruppen an der Autorisierung und Planung besonders schwer wiegender Veränderungen im Rahmen des *Change Management* dar, vgl. Abschnitt 3.1.3

<sup>7</sup>Dieser Abschnitt hat allerdings unterschiedliche Namen: „Key Performance Indicators“ (*Incident Management*), „Metrics“ (*Problem Management*), „Metrics and management reporting“ (*Change Management*), „Process Control“ (*Configuration Management* und *Release Management*)

### 3.1.3 Inhalte – Die Dienstmanagementprozesse

Im folgenden werden die wichtigsten Aspekte der zehn in der ITIL definierten Dienstmanagementprozesse erläutert. Hierbei werden, soweit möglich, die originalen Begriffe und Zuordnungen aus der ITIL verwendet um eine korrekte und realistische Zusammenfassung zu bieten, auch wenn dies der Klarheit der Darstellung und dem Verständnis der übergreifenden Konzepte nicht immer zuträglich ist. Für eine verständlichere Einführung sei auf die umfangreiche Sekundärliteratur zu ITIL (z.B. [vB05] oder [Köh05]) verwiesen, welche sich allerdings nicht auf eine Erläuterung der originalen ITIL-Prozesse beschränkt, sondern die enthaltenen Begriffe, Konzepte und Richtlinien in vielen Details modifiziert.

Für *Service Support* wird versucht weitgehend der Strukturierung der Darstellung in der ITIL zu entsprechen. Eine solche, wenigstens rudimentäre, durchgängige Darstellungsstruktur existiert allerdings für die Prozesse der *Service Delivery* nicht, für diese erfolgt daher eine allgemeine Zusammenfassung. Die Prozesse werden in der in gleichen Reihenfolge wie in den Original-Bänden vorgestellt.

#### Incident Management

Aufgabe des *Incident Management* ist die Steuerung der, oft unter Einhaltung zeitlicher Rahmenbedingungen zu erledigenden, Störungsbehebung und Behandlung von Service-Requests.

#### *Störungsbehebung*

**Zielsetzung** Das *Incident Management* ist mit der Behebung von Störungen befasst. Seine Aufgabe ist dabei primär einen gestörten Dienst, gegebenenfalls auch mittels einer Umgehungslösung (*Workaround*), so schnell wie möglich wieder in normaler Qualität herzustellen. Die Minimierung der Beeinträchtigung der Anwender (bei ihrer Arbeit) hat dabei Vorrang vor IT-orientierten Aspekten wie einer fundierten Ursachenanalyse.

#### *Incident und Service Request*

**Zentrale Konzepte und Begriffe** Ein *Incident* (Störung) ist ein Ereignis, das nicht zum regulären Betrieb eines Service gehört und das tatsächlich oder potenziell eine Unterbrechung oder eine Minderung der Service-Qualität verursacht. Ein *Service-Request* ist hingegen die Anfrage eines Anwenders nach Unterstützung, Service-Erweiterung, Dokumentation usw. im regulären Betrieb. An anderer Stelle trifft die ITIL allerdings die Aussage, dass ein *Service Request* jedes *Incident* sei, welches nicht auf eine Infrastrukturstörung zurückgehe – womit natürlich ein Widerspruch zur Definition von *Incident* entsteht. Auf alle Fälle ist nach der ITIL sowohl ein *Service Request* wie auch ein *Incident* im *Incident Management* zu behandeln.

Dies umfasst häufig die Weiterleitung des *Incident Record*, in dem die wichtigen Informationen zu einem *Incident* festzuhalten sind, zur weiteren Bearbeitung an eine andere Stelle (z.B. *Support Group*). Da die Behebung von *Incidents* häufig zeitkritisch ist, aber zu gewissen Zeiten sehr viel mehr *Incidents* auftreten können als Mitarbeiter zur Verfügung stehen, ist einem *Incident Record* auch eine *Priority* (Priorität) zuzuordnen.

*Incident Record*

Die reguläre Weiterleitung zu einer *Support Group* oder auch zu einem externen Support wird in der ITIL als *Functional Escalation* oder auch horizontale Eskalation bezeichnet. Muss hingegen beispielsweise der *Incident Manager* informiert werden, so stellt dies im Sprachgebrauch der ITIL eine *Hierarchical Escalation* oder vertikale Eskalation dar.

*Horizontale und vertikale Eskalation*

**Aktivitäten** Im eigentlichen Prozessablauf umfasst das *Incident Management* fünf Aktivitäten. Im *Incident Detection and Recording* werden Störungsmeldungen angenommen, ein *Incident Record* angelegt und dieser mit den wichtigsten, z.B. vom Störungsmelder aufgenommenen oder aus der CMDB abgefragten Informationen ergänzt. In der darauf folgenden Aktivität *Classification and Initial Support* wird die Kategorie des *Incidents* bestimmt und die Symptomatik des *Incident* mit den dokumentierten *Problems* und *Known Errors* verglichen (*Incident Matching*). Ist dies erfolgreich, so sollte eine sofortige Lösung möglich sein – falls nicht, muss das *Incident* nach der Zuweisung einer auf Basis von *Impact* (Auswirkung) und *Urgency* (Dringlichkeit) bestimmten Priorität an eine *Support Group* weitergeleitet werden (*Functional Escalation*). Nur in diesem Fall wird im Rahmen der *Investigation and Diagnosis* das *Incident* weiter analysiert und versucht eine Lösung zu spezifizieren. Wo notwendig, wird im Rahmen der *Resolution and Recovery* die Umsetzung der Lösung als *RFC* (vgl. Abschnitt 3.1.3 eingereicht und der Fortschritt dieses RFC regelmäßig überprüft. Schließlich wird in der *Incident Closure* mit dem von der Störung betroffenen Anwendern abgeglichen, ob die Lösung erfolgreich war und nach einer Überprüfen und Ergänzung das *Incident Record* geschlossen.

*Hauptprozess*

Das *Ownership, Monitoring, Tracking and Communication* ist die sechste unter „Incident Management Activities“ in *Service Support* aufgeführte Aktivität. Sie kann als nebenläufig zu den oben genannten Aktivitäten angesehen werden und hat die Aufgabe die Kommunikation mit dem Anwender aufrecht erhalten während der Verlauf der Bearbeitung des *Incident* überwacht und gegebenenfalls eine Eskalation ausgelöst wird. Auch das *Handling of Major Incidents*, ein gesonderter Abschnitt im Kapitel zu *Incident Management in Service Support* kann aber als eine Aktivität (oder Teilprozess) verstanden werden.

*begleitende und Ausnahme-Aktivitäten*

**Rollen** Die ITIL definiert zwei grundlegende Rollen: Den *Incident Manager*, den man als Prozessmanager (vgl. Abschnitt 2.9) des *Incident Management* an-

sehen kann, sowie den *Incident-handling support staff*, der für die eigentliche Ausführung der Aktivitäten zuständig ist.

**Prozesskontrolle** Wie auch für die anderen Prozesse im *Service Support*, gibt die ITIL nur Beispiele für mögliche KPIs vor: Die Gesamtzahl der *Incidents*, die durchschnittliche Lösungszeit und so weiter.

#### Problem Management

Das *Problem Management* ist für die Identifizierung und Behebung der den *Incidents* zugrunde liegenden Probleme zuständig. Außerdem soll es proaktiv Trends in der Dienstleistung analysieren, um das Auftreten von Störungen durch vorbeugende Maßnahmen minimieren zu können

#### Behebung der Ursache von Störungen

**Zielsetzung** Ziel des *Problem Management* ist die nachhaltige Lösung von Problemen. Dies bedeutet, wo sich das *Incident Management* auf die Wiederherstellung des Dienstes konzentriert, geht das *Problem Management* den Fehlerursachen auf den Grund (im Sinne einer *Root Cause Analysis*) und versucht deren Auswirkung zu minimieren oder sie zu beseitigen.

#### Problems und Known Errors

**Zentrale Konzepte und Begriffe** In der ITIL wird ein *Problem* als die Ursache eines oder mehrerer *Incidents* definiert. Als *Known Error* gilt ein *Problem*(oder ein *Incident*), dessen Ursache (*Root Cause*) bekannt ist – ein *Known Error* ist also die bekannte Ursache eines oder mehrerer *Incidents*. Allerdings ist nach ITIL auch ein identifizierter *Work-around* Teil Voraussetzung für einen *Known Error*<sup>8</sup>.

Ein solcher *Work-around* ist auch bei den entsprechenden Einträgen in der *Problem / Error Database* zu dokumentieren, in welcher die *Problems* und *Known Errors* abgespeichert sind.

*Problems* können im laufenden Betrieb, insbesondere durch Meldungen des *Incident Management*, durch Trendanalysen in der Beobachtung der Infrastruktur, aber auch schon in der Entwicklungs- bzw. Testumgebung erfasst werden.

**Aktivitäten** *Problem Management* besteht in der ITIL aus drei Teilprozessen. *Problem Control* und *Error Control* stellen den eigentlichen Hauptprozess der Problembehandlung dar, während das *Proactive Problem Management* vorbeugende Maßnahmen umfasst.

---

<sup>8</sup>In den Details ist die Zweckmäßigkeit dieser Definition – wie bei einigen anderen ITIL-Begriffen auch – allerdings zweifelhaft, vgl. Abschnitt 6.1.2

*Problem Control* und *Error Control* sind verknüpft. Wird also *Problem Control* durch das Finden der *Root Cause* abgeschlossen, so wird aus dem *Problem* ein *Known Error*, der in *Error Control* weiter behandelt wird. Dies geschieht so lange, bis die Aktionen zur Behebung der Ursache (des *Known Error*) abgeschlossen sind und ihr Erfolg durch ein *Post Implementation Review* verifiziert ist.

*Problem Control  
und Error  
Control*

*Proactive Problem Management* ist kein Prozess oder Teilprozess im eigentlichen Sinne, sondern befasst sich vor allem mit der frühzeitigen Identifikation von *Problems* oder allgemeinen Infrastrukturschwächen durch Trendanalysen. Konsequenz hieraus kann das Erstellen eines *Problem Record*, schon bevor durch das *Problem* überhaupt *Incidents* ausgelöst werden, sein oder Aktionen zur allgemeinen Verbesserung der Infrastruktur beziehungsweise der organisatorischen Abläufe (beispielsweise in Bezug auf Testprozeduren).

*Proactive  
Problem  
Management*

**Rollen** Ähnlich wie im *Incident Management* existieren zwei grundlegende Rollen: Der *Problem Manager*, den man als Prozessmanager (vgl. Abschnitt 2.9) des *Problem Management* ansehen kann, sowie der *Problem Support*, der für die eigentliche Ausführung der Aktivitäten zuständig ist (also die Entsprechung zu *Incident-handling support staff* darstellt)

**Prozesskontrolle** Es werden in der ITIL einige KPIs<sup>9</sup> für das *Problem Management* genannt, beispielsweise die Zahl der *Incidents*, die aufgetreten sind, bevor das zugrunde liegende *Problem* behoben werden konnte.

## Configuration Management

Das *Configuration Management* hat eine zentrale Bedeutung im Service Support, da alle anderen Dienstmanagementprozesse, die ITIL in *Service Support* und *Service Delivery* definiert, von seinen Leistungen abhängig sind.

**Zielsetzung** Ziel des *Configuration Management* ist die Dokumentation aller IT-Komponenten und ihrer Konfiguration. Hiermit sollen alle Prozesse des Dienstmanagements unterstützt werden. Wenn dies auch nicht explizit im zugehörigen Kapitel als Ziel genannt wird, so ist *Configuration Management* aber doch in der ITIL letztlich auch weitgehend für eine Informationsintegration zwischen den Dienstmanagementprozesse zuständig.

*Dokumentation  
der  
IT-Infrastruktur*

<sup>9</sup>allerdings ist im Kapitel *Problem Management* nicht von KPIs die Rede, sondern von „Metrics“

<i>CIs und die CMDB</i>	<p><b>Zentrale Konzepte und Begriffe</b> Ein unter Kontrolle des <i>Configuration Management</i> stehendes Element (z.B. Komponenten der IT-Infrastruktur, Dienst) wird <i>Configuration Item</i> (CI) genannt. Die Attribute, die zu einem <i>Configuration Item</i> erhoben werden, sowie die Beziehungen, in denen es mit anderen <i>Configuration Items</i> steht, werden in der <i>CMDB</i> (Configuration Management Database) gespeichert.</p>
<i>Einführung von Configuration Management</i>	<p><b>Aktivitäten</b> Es werden insgesamt sieben Aktivitäten genannt, wobei in den Ausführungen zu <i>Basic Concepts</i> im gleichen Kapitel wie auch in der meisten Sekundärliteratur nur die ersten fünf Erwähnung finden (vgl. z.B. [vB05]).</p> <p>In <i>Configuration Management Planning</i> werden die Rahmenbedingungen des zu implementierenden <i>Configuration Management</i> festgelegt, also Umfang, Ziel, Verantwortungsbereiche und so weiter. <i>Configuration Identification</i> befasst sich mit der Frage, welche Elemente als <i>Configuration Items</i> deklariert und wie diese in der <i>CMDB</i> dokumentiert werden sollten. Diese beiden Aktivitäten finden also vor allem im Rahmen einer Einführung von <i>Configuration Management</i> statt und haben eher den Charakter einer entsprechenden Projektphase.</p>
<i>Aktualisierung und Kontrolle der CMDB</i>	<p><i>Control of CIs</i> fasst die Teilaktivitäten zusammen, welche sicherstellen sollen, dass in der <i>CMDB</i> erfasste Eigenschaften der CIs auf dem aktuellen Stand gehalten werden (z.B. bei Änderungen an diesen). <i>Configuration Status Accounting</i> bezeichnet die Nachverfolgung des Status erfasster CIs. Dieser Status spiegelt die Position des CI in seinem Lebenszyklus wider (also beispielsweise: „Entwicklung“, „Test“, „Live“, ...). <i>Configuration Verification and Audit</i> beinhaltet Teilaktivitäten zur Planung und Durchführung von Prüfungen (<i>Configuration Auditss</i>) der Vollständigkeit, Genauigkeit und Aktualität der <i>CMDB</i>. Man kann hierbei <i>Control of CIs</i> als den eigentlichen „on-going“ Prozess des <i>Configuration Management</i> auffassen, <i>Configuration Status Accounting</i> als eine Spezialisierung davon und <i>Configuration Verification and Audit</i> als einen dazu gehörigen Kontrollprozess – allerdings bleibt, wie sehr häufig in der ITIL, ein breiter Interpretationsspielraum.</p>
<i>Pflege der CMDB und Berichterstattung</i>	<p><i>CMDB Back-ups, Archives and Housekeeping</i> soll für die Erstellung regelmäßiger Sicherheitskopien und Archivierungen der <i>CMDB</i> sorgen sowie eine weitgehende Redundanzfreiheit sicherstellen. <i>Providing a Configuration Management Service</i> schließlich ist eine Sammlung von Leistungen, die vom <i>Configuration Management</i> für die Organisation erbracht werden sollten, speziell die Erstellung von Reports. Diese beiden Aktivitäten finden allerdings, wie oben erwähnt, selten Beachtung.</p>
	<p><b>Rollen</b> Die Rollen werden in <i>Configuration Management</i>, anders als in den vorhergehenden Kapiteln, nicht in einer dedizierten Sektion, sondern teilweise im Abschnitt <i>Planning and Implementation</i> und zu einem anderen Teil im Anhang</p>

behandelt. Zunächst gibt es wie zu erwarten die Rolle *Configuration Manager*, welcher wieder als Prozessmanager anzusehen ist. Diesem ist, auch wenn diese Rolle nicht direkt als solche benannt wird, ein *Configuration Management Team* zur Seite gestellt. Eine Unterrolle in diesem Team nimmt der *Configuration Librarian* ein, welcher insbesondere mit der Datenhaltung und Archivierung betraut ist.

**Prozesskontrolle** Die Richtlinien zur Prozesskontrolle fallen für *Configuration Management* noch vager als in den vorherigen Prozessen aus. Es gibt eine Sektion zu Management Reporting, in der einige Parameter aufgezählt sind, die dem Management zu berichten seien, deren Kontrolle aber in großen Teilen eigentlich Prozessen der *Service Delivery* zufällt (finanzielle Wert von *Configuration Items*, Kapazitäts-Informationen usw.).

Unter *Key Performance Indicators* werden einige mögliche Parameter genannt, die teils als sinnvoll (z.B. „exceptions reported during configuration audits“), teils als eigentlich Prozessen zugehörig (z.B. „the cycle time to approve and implement Changes“), teils aber auch als redundant, wenig sinnvoll oder unverständlich einzuordnen sind (z.B. „occasions when the 'configuration' is not as authorised“).

## Change Management

Das *Change Management* soll sicher stellen, dass Veränderungen an der IT-Infrastruktur (und gegebenenfalls an anderen für die Dienstleistung wichtigen Elementen) kontrolliert ablaufen.

**Zielsetzung** Ziel des *Change Management* ist es standardisierte Verfahren für Änderungen zu etablieren und änderungsbedingte Störungen zu minimieren.

**Zentrale Konzepte und Begriffe** Jede Veränderung an der für die Dienstleistung eingesetzten Infrastruktur, das heißt jeder *Change*, muss in einem *Request for Change* (RFC) beantragt werden. Im den meisten Fällen gehen RFCs dabei auf Wünsche der Kunden (Fachabteilungen) zurück oder werden im Kontext des *Problem Management*, des *Incident Management* oder Prozessen *Service Delivery* gestellt.

RFC

Obwohl die ITIL dies nicht eindeutig klar stellt, kann man davon ausgehen, dass der RFC das zentrale Dokument des *Change Management* ist, ähnlich wie der *Incident Record* für *Incident Management* oder der *Problem Record* für *Problem Management*. Dies entspräche jedenfalls der Auffassung des *Change Management* des eng an ITIL angelegten MOF (vgl. Abschnitt 3.7) [Mic05]. Zwar existiert der Begriff *Change Record* in der ITIL, er findet aber im Kapitel zu *Change*

RFC der  
„Record“ für  
Change  
Management

*Management* mit keinem Wort Erwähnung, sondern taucht primär im Kontext des *Configuration Management* auf. Nach dem Glossar beschreibt ein *Change Record* welche *Configuration Items* wie von einem autorisierten *Change* betroffen sind<sup>10</sup>. Die Dokumentation der Prozessaktivitäten erfolgt also nicht hierin, sondern in einem *RFC*.

**FSC und PSA** Im *Change Management* findet eine große Anzahl weiterer Begriffe und Konzepte Verwendung. Es handelt sich hierbei überwiegend um Dokumente, wie den grundsätzlich für jeden *Change* zu erstellenden *Implementation Plan*, den *Back-out Plan* oder den globalen (alle *Changes* in einem Dokument erfassenden) *Forward Schedule of Changes* (FSC). Weitere Begriffe im *Change Management* sind *Change Model*, *Projected Service Availability* (PSA) oder das *Change Log*, in dem die Aktionen des *Change Management* dokumentiert werden sollen. Ebenso wie im *Incident Management* einem *Incident*, wird auch einem *Change* eine Priorität und eine Kategorie zugewiesen (siehe unten). Eine spezielle Kategorie stellen die so genannten *Standard Changes* dar, welche eine automatische Genehmigung erhalten (ansonsten aber dem gleichen Prozess unterliegen).

**CAB** Das *Change Advisory Board* (CAB) ist Haupt-Entscheidungsgremium für *Changes* in der IT Umgebung. Es evaluiert und autorisiert *Changes* von großer Komplexität und Auswirkung und nimmt allgemeine Kontrollaufgaben wahr. Mitglieder des CAB sind typischerweise der *Change Manager*, Repräsentanten aus anderen Dienstmanagementprozesse sowie Anwender-Vertreter.

Jeder umgesetzte *Change* durchläuft ein *Post Implementation Review* (PIR) in welchem überprüft wird, ob seine Durchführung planmäßig abgelaufen ist, ob die mit ihm verfolgten Ziele erreicht wurden sowie gegebenenfalls, wie der Prozess für zukünftige *Changes* verbessert werden kann.

**Aktivitäten** Unter *Activities* listet das Kapitel *Change Management* 15 Unterpunkte, von denen allerdings einige nicht als Aktivitäten zu sehen sind

**zahlreiche Aktivitäten** *Planning the Implementation of Operational Processes* enthält einen eher allgemeinen Hinweis, dass ein Prozess für das *Change Management* definiert werden sollte. *Change Logging and Filtering* hingegen ist eine echte Aktivität, in der *RFCs* erfasst und nach formalen Kriterien gefiltert werden. Hierauf folgen die Aktivitäten *Allocation of Priorities* und *Change Categorisation*. Als nächstes wird die Aktivität *CAB Meetings* genannt (siehe der Abschnitt zu Rollen weiter unten). Solche Treffen finden aber eher im Kontext der beiden nächstgenannten Aktivitäten statt, dem *Impact and Resource Assessment* oder dem *Change Approval*. Speziell das *Impact and Resource Assessment* ist in hohem Maße von der Bereitstellung genauer Daten durch das *Configuration Management* abhängig. Ist ein *Change* genehmigt worden, so kann er nach einer Zeitplanung im *Change*

<sup>10</sup>dies stimmt auch mit der Definition von *Change Record* in ISO/IEC 20000 [ISO05b] überein



*Scheduling* in der Aktivität *Change Building, Testing and Implementation* umgesetzt werden.

*Urgent Changes* und *Urgent Change Building, Testing and Implementation* sind weniger als eigenständige Aktivitäten denn als Richtlinien zur Anpassung der übrigen Aktivitäten für extrem dringliche Änderungen anzusehen. *Change Review* hingegen ist wieder als eine echte Prozessaktivität, welche für jeden *Change* ausgeführt werden soll, interpretierbar.

*Urgent Changes*

Die letzten drei genannten „Aktivitäten“ – *Reviewing the Change Management Process for Efficiency and Effectiveness*, *Roles and Responsibilities* und *Establishing a Change Advisory Board* – sind aber wiederum nicht als Prozessaktivitäten im üblichen Sinne aufzufassen.

*Pseudo-Aktivitäten*

**Rollen** Neben der kurioserweise in einem Unterabschnitt von *Activities* definierten Rolle des *Change Manager* wird keine weitere explizit festgelegt. Verteilt über die einzelnen Abschnitte finden aber zahlreiche Akteure Erwähnung. So ist unter anderem von *Staff*, der vom *Change Management* benötigt wird, die Rede oder von *Change Authority*, *Change Builder* oder *Change Tester* – auch das CAB kann als Rolle angesehen werden.

**Prozesskontrolle** Beispiele für Metriken sind die Anzahl von *Changes* pro Zeit, die Zahl der abgelehnten *RFCs*, die Anzahl der notwendig gewordenen *Backouts* oder auch die Zahl der unerwarteten Störungen, die sich nach der Umsetzung eines *Change* ergeben haben.

## Release Management

Das *Release Management* kontrolliert die Einbringung neuer Versionen von Software und Hardware in die Produktionsumgebung, steht also in einem engen Zusammenhang mit dem *Change Management*.

**Zielsetzung** Primäres Ziel des *Release Management* ist, die IT-Infrastruktur bei der Umsetzung von Veränderungen in einem bekannten und stabilen Zustand zu halten.

*Stabilhaltung der Infrastruktur*

**Zentrale Konzepte und Begriffe** *Release* ist ein Begriff, welcher ursprünglich aus dem Bereich der Softwareentwicklung beziehungsweise des *Software Engineering* stammt (vgl. [Kni06]) – in der ITIL steht er für eine Sammlung autorisierter *Changes*, die zusammen in die Produktionsumgebung eingebracht werden sollen. Ein *Release Unit* beschreibt einen Teil der IT-Infrastruktur, der in der Regel

*Release, DSL und DHS*

für ein Release zusammengefasst wird (z.B. ein Software-Paket). Die aktuellen, autorisierten Versionen einer Software werden in der *Definitive Software Library* (DSL) sicher gespeichert und archiviert. Die Entsprechung für Hardware ist der *Definitive Hardware Store* (DHS).

**Release-Ablauf** **Aktivitäten** Die ITIL definiert sechs Aktivitäten für das *Release Management*. Der Gesamtablauf für ein *Release* kann Wochen oder Monate in Anspruch nehmen (siehe beispielsweise die in [Kni06] dokumentierten Zeiten), daher lohnt eine Projekt-mäßige Planung in der Aktivität *Release Planning*. Die Zusammenstellung eines *Release* erfolgt in *Designing, Building and Configuring a Release*. *Release Acceptance* koordiniert die erforderlichen Tests. Sind diese erfolgreich abgeschlossen, bereiten das *Rollout Planning* und *Communication, Preparation and Training* die *Distribution and Installation*, das heißt die Einbringung des Release die Produktionumgebung, vor.

**Rollen** Im Kapitel *Release Management* selbst werden keine Rollen definiert. Die zu erwartende Rolle des *Release Manager* findet aber im Anhang zu *Configuration Management* (in *Setting up a Change, Configuration and Release Management Function*) Erwähnung.

**Prozesskontrolle** ITIL definiert einige KPIs, von denen die meisten ja/nein-Kriterien für ein erfolgreiches *Release Management* – oder aber auch *Change Management* – darstellen (z.B. „no evidence of use of unauthorised software at any site“). Zusätzlich Metriken umfassen beispielsweise die Zahl der Releases in einem Berichtszeitraum.

#### Service Level Management

**zentrale Stellung unter den Dienstmanagementprozessen** Die Aufgaben des Prozesses *Service Level Management* (SLM) liegen vornehmlich in dem Management der Beziehung zum Kunden. Dies umfasst vornehmlich das Management von SLAs, das heißt ihre Planung, Vereinbarung und die Überprüfung ihrer Einhaltung. Das Bewerten der Auswirkungen geplanter Änderungen auf *Service Levels* (auch wenn dieser Begriff also solcher in ITIL nicht definiert wird) geschieht ebenfalls durch das *Service Level Management*. Das *Service Level Management* nimmt somit unter den Prozessen der *Service Delivery* er ITIL eine zentrale Stellung ein.

**SLA, OLA und UC** In der ITIL finden, neben dem gängigen Begriff des *Service Level Agreement*, auch die des *Operating Level Agreement* (OLA) und des *Underpinning Contract* (UC) Verwendung (siehe Abbildung 3.2). Hierbei handelt es sich um Vereinbarungen, die zur Absicherung des Service, das heißt zur Sicherstellung der Erbringung

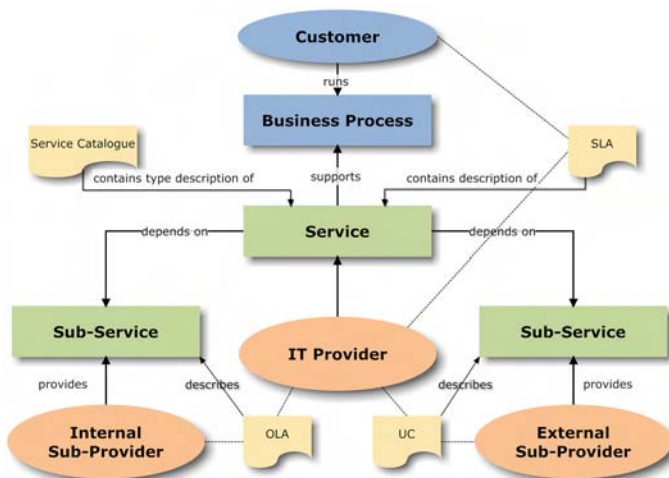


Abbildung 3.2: Zentrale Begriffe im ITIL Service Level Management

von Teildiensten, getroffen werden. OLAs sind dabei Vereinbarungen mit Abteilungen innerhalb der übergeordneten Organisation des IT-Service-Providers (also beispielsweise andere IT-Abteilungen im selben Unternehmen), UCs sind Vereinbarungen mit externen Zulieferern von Diensten. Die Koordination und Pflege dieser Beziehungen unterliegt ebenfalls dem SLM. Das *Service Level Management* erstellt auch den *IT Service Catalogue*, wenn auch kaum Angaben darüber gemacht werden, wie dieser auszusehen hat.

Für das *Service Level Management* wird eine Vielzahl von Aktivitäten definiert, die jeweils einem der Bereiche *Planning the process*, *Implementing the process* und *On-going Process* zugeordnet sind. Hierbei ist der *On-going Process* als der eigentliche Prozess des *Service Level Management* anzusehen.

Aktivitäten

Entsprechend den Aufgaben des Prozesses hat der *Service Level Manager* eine Schnittstellenfunktion: Er vertritt den Service-Provider gegenüber dem Kunden und agiert häufig als Interessensvertreter des Kunden innerhalb der Service-Provider-Organisation.

Service Level Manager

### Financial Management for IT Services

Ziel des *Financial Management for IT Services* ist die effiziente Verwaltung der finanziellen Ressourcen des IT-Service-Providers. Hierunter fallen die Bereiche *Budgeting* (Vorhersage und Kontrolle der Ausgaben), *IT Accounting* (Zuord-

finanzielle Ressourcen

nung der Ausgaben zu Aktivitäten, angeforderten Diensten usw.) und *Charging* (Rechnungsstellung). Prozessaktivitäten im eigentlichen Sinn werden nicht definiert, es gibt allerdings eine Aufstellung regelmäßig durchzuführender (täglich, monatlich, einmal pro Quartal, jährlich) Aktivitäten.

**Financial Manager** Die Rolle *Financial Manager* wird weder definiert noch genannt, findet aber in der Sekundärliteratur und in der Zusammenfassung des *Financial Management for IT Services* im Band *Service Support* Erwähnung. Richtlinien zur Prozesskontrolle werden in der Form von Hinweisen zum *Auditing* der drei Bereiche (*Budgeting*, *IT Accounting* und *Charging*) gegeben.

#### Capacity Management

**Trendanalysen und Pläne** Ziel des *Capacity Management* ist ein kosteneffektives Management von IT-Kapazitäten bei Sicherstellung der Erfüllbarkeit von vereinbarten *Service Levels*. Hierzu gehört die Erfassung aktueller Kapazitäten und die Vorhersage und Planung des zukünftigen Bedarfs. Das *Capacity Management* unterhält eine *Capacity Database* (CDB), analysiert Trends und erstellt einen *Capacity Plan* um sicherzustellen, dass die in den SLA vereinbarten Anforderungen kosteneffizient erfüllt werden können.

**Business, Service & Resource Capacity Management** Die ITIL teilt das *Capacity Management* in die drei Teilprozesse, das *Business Capacity Management*, *Service Capacity Management* und *Resource Capacity Management*. *Business Capacity Management* befasst sich mit der Vorhersage von Kapazitätsanforderungen seitens der Kunden. In der Verantwortung von *Service Capacity Management* liegt die Vorhersage entsprechender Trends für die Anforderungen an die IT-Dienste. Ihm obliegt aber auch eine eventuelle Steuerung der Beeinflussung der Nachfrage nach IT-Diensten, beispielsweise mittels einer Verrechnung (vgl. *IT Accounting* im *Financial Management for IT Services*). Das *Resource Capacity Management* ist schließlich mit der Kontrolle und Steuerung der Auslastung der Komponenten der IT-Infrastruktur beauftragt.

Auch hier ist die Rolle *Capacity Manager* nicht explizit definiert, aber in der Sekundärliteratur (vgl. [vB05]) fest verankert.

#### IT Service Continuity Management

**Planung für Katastrophenfälle** Das Ziel des *IT Service Continuity Management* (ITSCM) ist Sicherung des Überlebens der ITSP-Organisation. Es befasst sich daher mit Maßnahmen, die ein Weiterführen oder die Wiederaufnahme eines Dienstes auch nach schweren oder katastrophalen Störungen, z.B. einem Brand, sicherstellen sollen. Dieser Prozess befasst sich also hauptsächlich mit der Einschätzung von Risiken, das heißt möglichen Ausnahmefällen und deren Auswirkungen auf die ITSP-Organisation, sowie

Maßnahmen zur Reduktion von Risiken. Für trotzdem eintretende Ausnahmefälle sind Notfallpläne zu entwickeln, regelmäßig zu überprüfen und testen, sowie entsprechende Trainings der Mitarbeiter durchzuführen.

Es werden keine Rollen explizit festgelegt, aber angeregt, dass die Rollen eines vorhandenen *Business Continuity Managements* Entsprechungen im ITSCM haben – in diesem Zusammenhang sind die Rollen *ITSC Manager*, *ITSC Team Leader* und *ITSC Team Member* genannt.

### Availability Management

Die Aufgabe des *Availability Management* besteht darin die Zuverlässigkeit der Dienste sicherzustellen. Es ähnelt in vielerlei Hinsicht dem *Capacity Management*. Auch das *Availability Management* kontrolliert den Status der Infrastruktur, bestimmt Anforderungen und versucht die Gestaltung der Infrastruktur so zu beeinflussen, dass diese Anforderungen möglichst kosteneffizient umgesetzt werden können – nur eben unter Verfügbarkeitsaspekten statt aus Kapazitäts- oder Performance-Sicht.

*Planung unter Verfügbarkeitsaspekten*

Aktivitäten als solche werden nicht definiert, es werden aber verschiedene Methoden zur Überwachung, Analyse und Planung der Infrastruktur besprochen. So werden beispielsweise verschiedene, zum großen Teil aus dem Ingenieurwesen entlehnte Techniken zur Infrastrukturanalyse empfohlen, wie *Component Failure Impact Analysis* (CFIA) oder *Fault Tree Analysis* (FTA). Die Rolle des *Availability Manager* wird im Anhang des Kapitels beschrieben, auf die Darlegung von KPIs für das *Availability Management* verzichtet die ITIL (es werden allerdings Metriken zur Berichterstattung über Dienstverfügbarkeiten diskutiert).

*Infrastruktur-analyse*

### 3.1.4 Werkzeuge für ITIL

In *Service Support* und *Service Delivery* finden sich an verschiedenen Stellen Hinweise zu ITSM-Tools. Zum einen existiert in beiden Bänden ein (identisches) Kapitel zu *Service Management Software Tools*. Dieses enthält allerdings auf drei Seiten nur zwei nicht aufeinander aufbauende Aufzählungen von Tool-Kategorien, ein paar sehr allgemeine Auswahlkriterien sowie die Aussage, dass für die Werkzeuge Trainings stattfinden sollten. Zum anderen finden sich auch in einigen Prozess-Kapiteln, vornehmlich in *Service Support*, Abschnitte zu Tools, welche im Umfang stark variieren. So umfasst der entsprechende Abschnitt in *Incident Management* gerade einmal sieben Stichpunkte (mit je einer kurzen Aussage), der Abschnitt in *Configuration Management* hingegen beinahe drei Seiten. Letzterer überschneidet sich thematisch mit den Abschnitten zu Werkzeugen in *Change Management* und *Release Management*.

*kaum Angaben zur Werkzeugunterstützung*

keine Kriterien  
für „ITIL-Tools“

Generell bleiben die Aussagen der ITIL zu Werkzeugen aber wenig aussagekräftig und belastbar. In den entsprechenden Abschnitten werden Aufzählungen und Beschreibungen von einsetzbaren Standard-Werkzeugen, funktionale und nicht-funktionale Anforderungen an spezialisierte Werkzeuge in verschiedenstem Detaillierungsgrad sowie allgemeine Auswahl-Kriterien häufig ohne erkennbare Strukturierung zusammen gewürfelt, so dass sich kein klares Bild ergibt. Auch außerhalb der Bände *Service Support* und *Service Delivery* gibt der Eigentümer der ITIL, das OGC, keine Publikationen heraus, die sich mit einer Toolunterstützung befassen, noch sind anscheinend in diese Richtung gehende Aktivitäten für die ITIL geplant<sup>11</sup>. Dies führt zu der Situation, dass es keinerlei anerkannte Kriterien für ein „ITIL-Tool“ existieren (eine ausführlichere Diskussion hierzu findet sich in Abschnitt 4.1).

#### 3.1.5 Diskussion

Die ITIL bezeichnet sich selbst als „de-facto Standard“. Sie ist bei genauerer Betrachtung allerdings deutlich anders beschaffen als es der Begriff *Standard*, den man normalerweise mit einer konkreten und prüfbaren Vereinheitlichung verbindet, erwarten ließe.

„descriptive  
guidance“

Die ITIL formuliert viele Praktiken nur als Vorschlag oder Option. Die Inhalte der ITIL werden daher auch häufig als deskriptiv (*Descriptive Guidance*) bezeichnet, die sich mehr um das „was?“ und „warum?“ dreht, als das „wie?“ einer Prozessimplementierung (im Gegensatz zu *Prescriptive Guidance*, d.h. konkreten, präskriptiven Richtlinien [Pul02]). Diese nicht-präskriptive Natur der ITIL ist dabei durchaus beabsichtigt. Das selbst gesteckte Ziel der ITIL ist schließlich Richtlinien für IT-Organisationen aller Sektoren und aller Größen zur Verfügung zu stellen (siehe z.B. *Preface* und Kapitel 1 in [OGC00]). Hierbei ist natürlich nachvollziehbar, dass nicht jede Vorgehensweise für alle Organisationen mit Vorteil anwendbar ist – beispielsweise können einige weiterreichende Vorschläge zur Dokumentation und Formalisierung der Kommunikation für sehr kleine IT-Einheiten, in denen die Kommunikation auch informell gut funktioniert, eventuell keinen Vorteil bringen, der den mit der Einführung einhergehenden bürokratischen Aufwand rechtfertigen würde.

Kritik an ITIL

Aber auch unter diesen Voraussetzungen wäre eine klarere Struktur und eine konkretere Darstellung der Inhalte unter vielen Gesichtspunkten wünschenswert. So kommen mehrere Arbeiten zu dem Ergebnis, dass die ITIL (schon aufgrund des mangelnden Formalismus in der Darstellung) nur bedingt als Referenzmodell angesehen werden kann (siehe [Pro03] und [HZB05]), da es *Grundsätze ordnungsgemäßer Modellierung* (GoM) [RSD03] nicht erfüllt. Auch der Mangel an einer einheitlichen Struktur in der Prozessbeschreibung (das heißt innerhalb der

<sup>11</sup>Im Gegensatz zur eTOM, deren Eigentümer TMF die Werkzeugunterstützung zu einem zentralen Thema gemacht hat, vgl. Abschnitt 3.2

Kapitel) und die daraus resultierende schwierige Navigation („wo findet innerhalb eines Kapitels welche Information?“) ist eine häufig geäußerte Kritik an der ITIL (siehe auch die Befragung der ITIL-Nutzer bezüglich ihrer Wünsche für die nächste ITIL-Version [OGC05b]).

Aber nicht nur die Auffinden von Inhalten, auch die informelle Darstellung scheint nicht ideal. Die Beschreibung der Prozesse erfolgt überwiegend in Fließtext und Aufzählungen. Strukturierte Darstellungsformen (z.B. wie Tabellen) werden nur selten eingesetzt, was viele der vorgestellten Richtlinien schwer im Detail nachvollziehbar macht. Auch die in der ITIL vorhandenen Illustrationen können meist nicht zur Präzisierung dienen. Diese lassen beispielsweise oft im Text erwähnte Aktivitäten ohne ersichtlichen Grund aus – vor allem aber halten sich die Grafiken nicht an eine etablierte Notation. Auch eine von ITIL selbst vorgeschlagene (in Anhang B.2 von [OGC00]) Notation zur grafischen Darstellung von Prozessen findet praktisch keine korrekte Anwendung. Eine enger gefasste Richtlinie zur Illustration von Prozessabläufen und anderen Sachverhalten gab es bei der Erstellung der ITIL offenbar nicht – selbst bei der Darstellung von Grafiken prinzipiell gleichen Typs (zum Beispiel Ablaufdiagramme) wird keine einheitlichen Notation durchgehalten.

*Darstellung der Inhalte*

Es mangelt aber auch an einer einheitlichen Begriffsverwendung. So kann man sich höchstens denken, was der Unterschied zwischen KPIs, *Metrics* und den an einigen Stellen unter *Management Reporting* aufgezählten Parametern ist – erklärt wird es in der ITIL nicht. Ob die häufig direkt unter einer *Activity* aufgezählten Teilaktivitäten nun wieder als „Activity“ (oder beispielsweise wie in MOF als *Procedure*) zu bezeichnen sind, bleibt ebenfalls undefiniert, ebenso eigentlich zentrale Begriffe wie *Service Level*.

*keine einheitliche Begriffsverwendung*

Die resultierende Unübersichtlichkeit ist offenbar schon bei der Erstellung der ITIL, bei der eine große Anzahl von Autoren beteiligt war, nicht ohne Folgen geblieben. So gibt es immer wieder Begriffe, bei denen sich verschiedene Erklärungen oder Verwendungen zumindest in Details unterscheiden (z.B. *Incident*, vgl. Abschnitt 3.1.3). Auch ansonsten sind die Richtlinien nicht durchgängig konsistent. Zum Beispiel: An welchem Punkt des Incident-Management-Prozesses gabelt sich der Workflow zu *Service Request Procedures*? Nach *Classification and Initial Support* wie in Abbildung 5.2 in Kapitel 5 in *Service Support* [OGC00] oder bereits nach *Incident Detection and Recording* wie es die Beschreibung dieser Aktivität und die Abbildung im Anhang des gleichen Kapitels zu suggerieren scheinen?

*vereinzelte Inkonsistenzen*

Insgesamt bleiben daher viele Inhalte der ITIL sehr schwer fassbar. Dies ist wohl auch der Grund, weshalb praktisch die gesamte Sekundärliteratur in vielen Punkten nicht den konkreten Inhalt der ITIL erläutert oder diskutiert, sondern eher die Grundideen der Prozesse übernimmt und diese mit teils umbenannten, weggelassenen oder hinzugefügten Elementen neu interpretiert. In Bezug auf eine Werkzeugunterstützung ist der Umstand, dass die ITIL einen solch großen Inter-

pretationsspielraum lässt, natürlich keine ideale Ausgangslage.

## 3.2 eTOM

Die *enhanced Telecom Operations Map* (eTOM) kann als das Gegenstück der Telekommunikationsbranche zur der im Data-Center-Umfeld entstandenen ITIL angesehen werden. Da in der eTOM, ebenso wie bei der ITIL, die deklarierte Zielsetzung sehr allgemein gehalten ist, ist es für ein Verständnis lohnend im folgenden Abschnitt den Kontext zu betrachten in dem eTOM entstanden ist. Danach wird in Abschnitt 3.2.2 auf den Aufbau und die Inhalte der aktuellen eTOM-Version eingegangen. Nach einer Betrachtung wie die eTOM auf ITIL abbildbar ist (Abschnitt 3.2.3), folgt eine abschließende Zusammenfassung und Bewertung (Abschnitt 3.2.4).

### 3.2.1 Herkunft und Zielsetzung

*standardisiertes  
Prozessmodell  
für ICSPs*

Einen ganz ähnlichen Ansatz wie die OGC mit der ITIL verfolgt auch das *Telemanagement Forum* (TMF)<sup>12</sup> mit seiner eTOM, die den Standard für die Managementprozesse der *Internet and Communications Service Provider* (ICSP), d.h. Unternehmen der Telekommunikationsindustrie, definieren soll. Die eTOM könnte in der Zukunft auch außerhalb des Telekom-Umfeldes eine wichtige Rolle spielen. Schließlich wachsen die Geschäftsfelder der IT- und der Telekommunikationsindustrie zunehmend zusammen. Beispielsweise werden mit Voice-over-IP in vielen IT-Infrastrukturen mittlerweile klassische Telekommunikationsleistungen unterstützt, auf der anderen Seite beginnen vor allem im Mobilfunksektor die Telekommunikationsprovider zunehmend Dienste, die über die reine Sprach- oder Datenkommunikation hinausgehen, anzubieten. Die Management-Aufgaben, die hierbei entstehen, unterscheiden sich also nicht grundsätzlich und die Verbreitung der eTOM auch in andere IT-Sektoren ist vom TMF mehrfach propagiert worden.

Die eTOM besitzt zwar eine gleiche grundlegende Zielsetzung wie die ITIL, in vielen ihrer Konzepte stimmen sie aber nicht überein, was zu einem großen Teil auf ihre unterschiedliche Herkunft zurückgeführt werden kann. Es lohnt sich daher, sich kurz mit der Geschichte des TMF, dem Herausgeber und Eigentümer der eTOM, sowie den Ursprüngen der eTOM und ihrer Einordnung in die anderen Aktivitäten des TMF auseinanderzusetzen.

---

<sup>12</sup>auch: TM-Forum



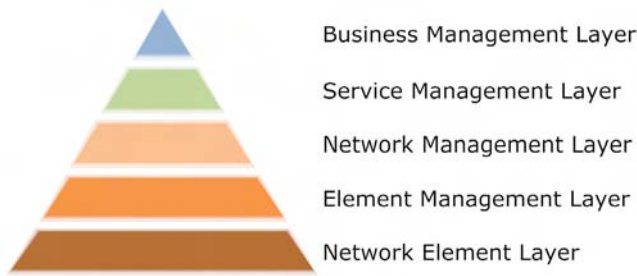


Abbildung 3.3: TMN-Pyramide [IT96]

### Telemanagement Forum

Das *Network Management Forum* (NMF), aus dem später das TMF entstand, wurde 1988 als Vereinigung amNetzmanagement interessierter Firmen aus der Telekommunikationsbranche (Gründungsmitglieder u.a. AT&T, BT) gegründet. Ziel des NMF war es zunächst standardisierte Empfehlungen für die Umsetzung des *Telecommunications Management Network* (TMN) der *International Telecommunication Union – Telecommunication Standardization Sector* (ITU-T) zu erarbeiten, zu dessen Erweiterung beizutragen und die Verbreitung in der Industrie zu fördern. TMN ist eine Management-Architektur, die sich relativ eng an die Vorgaben für das *OSI Management* der ISO anlehnt (vgl. Abschnitt 3.10). Zur Anwendung kommt sie vor allem im Betrieb öffentlicher Netze. Sie ist dabei die bislang einzige, auf *OSI Management* basierende Management-Architektur, die eine relevante (wenn auch nicht wirklich große) Verbreitung erfahren hat.

*Telecommuni-  
cations  
Management  
Network*

Die TMN-Managementarchitektur setzt den Rahmen für die Funktionalität von *Operations Support Systems* (OSS), das heißt Betriebsunterstützungs- bzw. Managementsysteme, deren Interaktionen untereinander sowie die Kommunikation mit den zu managenden Komponenten, den *Network Elements*. Netzwerkkomponenten TMN-konform managebar zu machen ist allerdings auch erheblich aufwändiger und damit kostspieliger als eine vergleichsweise simple Instrumentierung für Internet-Management, was wohl einer der wesentlichen Hinderungsgründe einer weiteren Verbreitung war.

*Operation  
Support  
Systems*

TMN teilt verschiedene Bereiche des IT-Managements anhand eines einfachen Grundmodells verschiedenen Schichten zu (siehe Abbildung 3.3) und versucht so die Komplexität des Managements für die einzelnen Bereiche auf ein beherrschbares Maß zu reduzieren. Jeder Bereich interagiert mit denen benachbarten Schichten über definierte Schnittstellen, die deren Komplexität weitgehend verschatten. Nicht-benachbarte Schichten sind, zumindest konzeptionell, völlig transparent (unsichtbar). Das Konzept soll verdeutlichen, dass das Management in je-

*TMN Modell*

der Schicht unterschiedliche Methoden und Werkzeuge umfasst und auf anderen Objekten operiert. Jede Schicht stellt dabei Funktionalität für die nächsthöhere Schicht zur Verfügung und unterliegt in ihren Management-Entscheidungen Beschränkungen und Vorgaben dieser.

#### TOM und eTOM als Basis von NGOSS

Die Version 1.0 der Telecom Operation Map (TOM) wurde 1998 vom TMF als „Evaluation Version“ veröffentlicht. Das Ziel der TOM, wie auch der eTOM, ist die Bereitstellung eines industrieeigenen Modells von Geschäftsprozessen. Dies umfasst auch die Definition einer gemeinsamen unternehmensunabhängigen Terminologie für das Dienstmanagement. Die TOM sollte darüber hinaus eine Diskussionsgrundlage für die zur Ausführung der Prozesse jeweils notwendigen Informationen darstellen. Die Ergebnisse hieraus sind mittlerweile in eine eigene Dokumentenfamilie des TMF, das *Shared Information and Data Model* (SID) [TMF04e] eingeflossen. eTOM und SID bilden inzwischen zusammen mit den Richtlinien der *Technology Neutral Architecture* (TNA) [TMF04d] und der Entwicklung von Konformitätstests [TMF04c] die Säulen des ehrgeizigen Projektes *New Generation Operations Systems and Software* (NGOSS) des TMF [RC05]. NGOSS soll eine herstellerunabhängige Architektur definieren, die den Aufbau kompletter Management-Lösungen aus Modulen mit standardisierten Schnittstellen erlaubt. Mit der Version 2.1 erschien im März 2000 die letzte Fassung der TOM [TMF00]. Die TOM besitzt eine klare, übersichtliche Grundstruktur (siehe Abbildung 3.4), schien dem TMF aber doch zu spezifisch auf klassische Telekommunikationsdienste ausgerichtet, um den Anforderungen des sich zunehmend verändernden Geschäftsumfeldes seiner Mitgliedsunternehmen gewachsen zu sein. Die Weiterentwicklungen der TOM geschahen dann unter dem neuen Namen eTOM.

#### 2.1 letzte Version der TOM

#### „verbesserte“ TOM

#### „e“ steht für „enhanced“ und „e-commerce“

Im Mai 2001 wurde die erste Version (1.0) der eTOM veröffentlicht. Dieses neue Prozessmodell beinhaltete so viele Neuerungen gegenüber der letzten TOM Version, dass das TMF einen neuen Namen für angemessen hielt. Das „e“ steht sowohl für „enhanced“ (verbessert) als auch für „electronic“ im Sinne von 'e-commerce', was die stärkere Berücksichtigung dieser neuen Art von Geschäftsmodellen unterstreichen sollte. Diese Neuausrichtung ist allerdings noch nicht vollständig vollzogen, die tatsächlichen „e-business“-Aspekte der eTOM beschränken sich zur Zeit hauptsächlich auf die Berücksichtigung von Lieferanten/Subprovider-Beziehungen.

#### Entwicklung der eTOM

Der Veröffentlichung der nächsten Versionen bis zur eTOM 3.0, der letzten kostenlos veröffentlichten Version, vollzog sich bis zum Mai 2002, also innerhalb

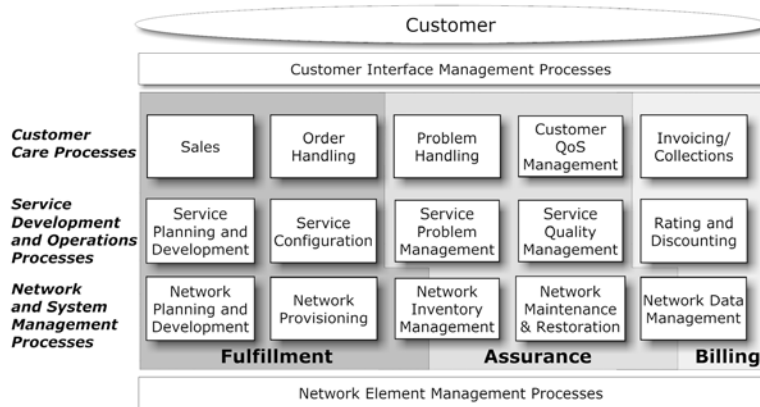


Abbildung 3.4: Telecom Operations Map [TMF00]

eines Jahres, relativ schnell. Allerdings schien auch diese Version im Vergleich zur TOM 2.1 noch in vielen Punkten der Prozessdarstellung unvollständig. Diese Lücken wurden erst durch die mit der Version 3.5 eingeführten Addenda und *Application Notes* geschlossen, die in der gegenwärtig neuesten, im Februar 2004 erschienenen Version 4.0 nun noch etwas umfangreicher ausgefallen sind.

Leider hat sich kurz danach die Politik des Telemanagement-Forum hinsichtlich der Distribution seiner Dokumente geändert. Die eTOM-Dokumente können mittlerweile nur noch von Mitgliedern des TMF kostenfrei über die Webseite bezogen werden, Nichtmitglieder müssen für den Bezug der Dokumente bezahlen. Dies betrifft auch die älteren Dokumente, die bis Ende 2002 kostenlos zugreifbar waren. Auch zahlungswillige Nichtmitglieder müssen in der Regel aber auf die jeweils neuesten Versionen der Dokumente verzichten und, speziell bei den kostengünstigeren Paketen von mehreren Dokumenten zu einem Thema (den so genannten *Solution Suites*), mit einer zum jeweiligen Zeitpunkt nicht mehr aktuellen Version der eTOM vorlieb nehmen. Diese Umstände stellen für eine weite Verbreitung der eTOM, insbesondere bei kleinen und mittleren Unternehmen, ein weiteres Hemmnis dar.

*eTOM nicht frei verfügbar*

### 3.2.2 Struktur und Inhalte

Im folgenden werden Konzepte und Inhalt der eTOM<sup>13</sup> dargelegt. Zunächst folgt ein Überblick die zur eTOM gehörenden Dokumente, die Grundstruktur und

<sup>13</sup>Die folgenden Ausführungen basieren auf der Version 4.0 der eTOM, siehe Literaturverweise. Aktuell ist zur Zeit die Version 4.1

die grundlegenden Prozessbereiche. In den darauf folgenden Abschnitten werden dann die Inhalte der Prozessbereiche dargelegt, wobei besonderes Gewicht auf das *Operations*, den ältesten und ausgereiftesten Prozessbereich, gelegt wird.

#### eTOM Dokumentenfamilie

*eTOM besteht aus mehreren Dokumenten* Neben dem eigentlichen Hauptdokument (GB921, [TMF04a]) besteht die eTOM noch aus mehreren begleitenden Dokumenten, die zum Teil wichtige Ergänzungen beinhalten. Hierzu gehören in der Version 4.0 die *Application Note C* (GB921C) [TMF04b] mit einem noch recht grobem Entwurf zu einer *Public B2B Business Operations Map*, das *Addendum D* mit *Process Decompositions and Descriptions* (GB921D) [TMF04f], das *Addendum F* (GB921F) mit *Process Flow Examples* [TMF04g], sowie schließlich die *eTOM-ITIL Application Note* (GB921L) [TMF04h]). Während die *Application Notes* nützlich, aber nicht mit den Kernfragestellungen der eTOM befasst sind, ist vor allem das *Addendum D (Process Decompositions and Descriptions)* unverzichtbar, da viele der in der eTOM enthaltenen Prozesselemente nur hier beschrieben sind.

#### View Levels

*unterschiedlich detaillierte Sichten* Die eTOM definiert verschiedene Sichten auf die Prozesse, die in Niveaus (*Levels*) eingeteilt sind, wobei numerisch höhere Levels eine zunehmend feinere Sicht der Unternehmensprozesse bieten. Auf der am niedrigsten detaillierten Stufe, dem *Level 0*, sind nur drei grundsätzliche Prozesse bzw. Prozessbereiche definiert, während bei der derzeit höchsten Stufe, dem *Level 3*, der allerdings bislang noch nicht vollständig ist, bereits recht spezialisierte Aktivitäten bzw. Teilprozesse dargestellt sind. Es ist auch eine weitere Aufteilung in ein *Level 4* vorgesehen. Prozesselemente, die aus einer Aufspaltung eines *Level 4* Prozesses entstehen, sollen allerdings auch wieder *Level 4* Elemente genannt werden. Dies erscheint etwas seltsam, ist zur Zeit allerdings auch von praktisch keiner Relevanz, da bislang noch nicht einmal ein einziger *Level 4* Prozess definiert und nur ein Teil der eTOM überhaupt bis *Level 3* spezifiziert ist.

*Baumstruktur der Views* Die beschriebene Aufteilung ordnet die in der eTOM definierten Prozesse in eine Baumstruktur ein (siehe Abbildung 3.5 auf der nächsten Seite). Prozesse niedriger Levels können also in Prozesse des nächst höheren Level aufgespalten werden, umgekehrt lässt sich jeder Prozess der Level 1 und aufwärts eindeutig einem Prozess des nächstniedrigeren Levels zuordnen. Der Prozessbegriff der eTOM entspricht der üblichen Definition, allerdings wird für die Prozesse der *Level 0* auch der Ausdruck *Process Area*, für *Level 1* Prozesse *Process Grouping* (Prozessgruppe oder -gruppierung), und für die Prozesse der höheren Level *Process Element* (Prozesselement) verwendet. Insgesamt sind diese Begriffe aber als synonym zu „Prozess“ anzusehen und werden im folgenden auch so verwendet.

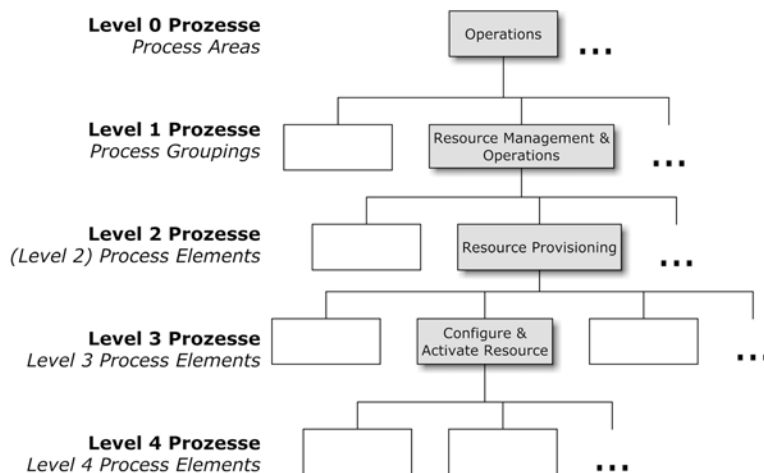


Abbildung 3.5: Strukturierung der eTOM-Prozesse [TMF04a]

Jeder definierte Prozess ist im eTOM Hauptdokument oder im *Addendum D* beschrieben. Diese Beschreibung ist aber meistens auf eine, insbesondere bei den Prozesselementen der höheren Levels recht kurze, allgemeine Aufgabenbeschreibung beschränkt. Weitergehende Umsetzungsempfehlungen, wie sie die ITIL enthält, werden nicht gegeben.

### Level 0 – Gesamtansicht des Unternehmens

Im Level 0 der eTOM werden die drei grundlegenden Unternehmensprozesse von ICS-Providern, sowie fünf wichtige externe Entitäten (entities), mit denen das Service Provider Unternehmen interagiert, festgelegt (siehe Abbildung 3.6). Diese Entitäten sind nach dem üblichen Verständnis externe Rollen, namentlich die des Kunden (*Customer*), die der Lieferanten bzw. Subprovider und Partner (*Suppliers/Partners*), der Angestellten (*Employees*), Anteilseigner (*Shareholders*) sowie der weiteren interessierten Parteien (*Other Stakeholders*, z.B. Regulierungsbehörden, Gewerkschaften, Medien usw.).

*Prozessbereiche  
und externe  
Rollen*

Der klare Schwerpunkt liegt hierbei auf der Rolle des Kunden, welche die einzige ist, die schon in der TOM dokumentiert war – aber auch die eTOM definiert sich selbst nach wie vor als „customer centric“. Die im Vergleich zur TOM neu hinzugekommene Lieferanten/Subprovider-Rolle spiegelt die Erfahrung wider, dass auch in der Telekommunikationsindustrie, insbesondere in zunehmend deregulierten Märkten, oft längst nicht mehr alle Leistungen der Wertschöpfungskette in-

*Schwerpunkt  
auf Rolle des  
Kunden*

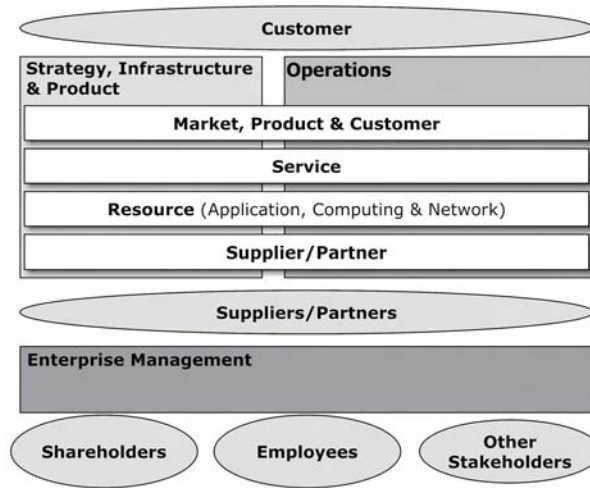


Abbildung 3.6: eTOM Level 0 Sicht [TMF04a]

house erbracht werden. So kann beispielsweise ein Provider bei der Realisierung eines IP-Dienstes auf Glasfaserstrecken eines anderen Unternehmens zurückgreifen. Dass die Qualität des IP-Dienstes, den der Provider seinem Kunden anbietet, nun aber nicht mehr nur von der eigenen Infrastruktur, sondern auch von den angemieteten Leitungen abhängt, muss im Management dieses Dienstes berücksichtigt werden. Die Suppliers/Partners-Rolle und die mit ihr befassten Prozesse (s.u.) stellen daher eine wichtige Ergänzung dar. Die anderen, unterhalb des Enterprise Management angesiedelten Rollen sind eine sinnvolle Vervollständigung, stehen aber nicht im Kern des eTOM-Gesamtkonzeptes.

*Berücksichtigung von Subprovider-Beziehungen*

*keine internen Rollen*

Im Vergleich zu anderen Modellen, wie z.B. MOF, ist das Rollenmodell der eTOM nur rudimentär entwickelt. So werden keinerlei unternehmensinterne Rollen, wie z.B. Prozessinhaber, definiert. Auch die mittlerweile im Service Management gängige Unterscheidung zwischen Kunde und Benutzer (Customer/User) trifft die eTOM nicht, es wird lediglich erwähnt, dass die Rolle des *Customer* in die des *Subscriber* und die des *End User* verfeinert werden kann – einen Niederschlag im Prozessmodell findet dies aber nicht. Alle von der eTOM definierten Rollen sind somit bereits im *Level 0* sichtbar und werden, im Gegensatz zu den Prozessen, in den höheren Levels nicht weiter verfeinert.

*Operations und SIP und Enterprise Management*

Direkt auf den Kunden ausgerichtet ist der Prozessbereich *Operations*. Dieser Bereich entspricht weitgehend der Abdeckung der TOM und beinhaltet im wesentlichen auch noch deren ursprüngliche Prozesse. Der Bereich *Strategy, Infrastructure & Product* (SIP) hingegen beinhaltet Prozesse, die nicht direkt zur

Wertschöpfung beitragen, sondern sich mit den längerfristigen dienstbezogenen Fragestellungen befassen. Im Großen und Ganzen befasst sich Operations mit der Bereitstellung und dem Betrieb von Dienstinstanzen, während die SIP-Prozesse für das Design neuer Dienstklassen, die Planung der dazu notwendigen Infrastruktur und die allgemeinen Produkt- und Marketingstrategien zuständig sind. Das *Enterprise Management* beinhaltet dagegen die allgemeinen, an betriebswirtschaftlichen Fragestellungen orientierten Prozesse, die bei jedem ICSP vorhanden, aber überwiegend nicht industriespezifisch sind.

Quer über den Bereichen *Operations* und *Strategy, Infrastructure & Product*, liegen vier so genannte funktionale Bereiche (*Functional Areas*), die nach den von den in ihnen enthaltenen Prozessen hauptsächlich gemanagten Objekten benannt sind. Diese Unterteilung ist ursprünglich aus der Schichtung des TMN-Modells (vgl. Abbildung 3.3), unter Weglassung der komponentenbezogenen untersten zwei Schichten, entstanden. Dafür hinzugekommen ist der nun unterste Bereich *Suppliers/Partners* und aus dem *Enterprise Management* des TMN ist nur der Aspekt *Market, Product & Customer* übernommen worden. Die anderen, nicht direkt Service-Provisioning orientierten Prozesse des *TMN Enterprise Management* haben in der eTOM ihren eigenen Bereich erhalten. Das TM-Forum arbeitet nach eigenen Aussagen mit der ITU-T daran, die Konzepte des TMN und der eTOM in Zukunft wieder stärker zu harmonisieren.

*Functional Areas*

## Operations

Die *Operations* Prozesse bilden in vielerlei Hinsicht den eigentlichen Kern der eTOM und werden im folgenden genauer vorgestellt. Die hierbei erläuterten grundlegenden Konzepte der Einteilung in funktionale und end-to-end Prozessgruppen (*end-to-end Process Groupings*), sowie der *Process Flows*, gelten prinzipiell auch für die Prozesse des Bereiches der *Strategy, Infrastructure & Support*, sind aber teilweise bisher nur im Operations Bereich ausreichend klar ausgeführt und dokumentiert.

*zentrale Process Area der eTOM*

Der Operations-Prozess wird in Level 1 entsprechend seiner funktionalen Schichtung in vier Teilprozesse unterteilt (siehe horizontale Unterteilung des Bereiches in Abbildung 3.7), gleichzeitig werden aber auch vier vertikale end-to-end Prozessgruppen definiert. Es gibt nun also nicht mehr nur eine horizontale, sondern auch vertikale Prozessunterteilung. Dieses Strukturierungsschema, das die weiter verfeinerten *Level 2* Prozesse sowohl einer funktionalen (horizontalen) sowie mindestens einer end-to-end (vertikalen) Prozessgruppe zuordnet, ist wohl mit das bemerkenswerteste Teilkonzept der eTOM. Zunächst wird auf die vier funktionalen Prozessgruppen und deren Verfeinerung in *Level 2* Prozesselemente eingegangen, sowie ein Beispiel für die Zerlegung eines *Level 2* Elementes in *Level 3* Prozesselemente gezeigt. Das Konzept der *end-to-end Process Groupings* wird in Abschnitt 3.2.2, das der *Process Flows* in Abschnitt 3.2.2 erläutert.

*vertikale und horizontale Einordnung*

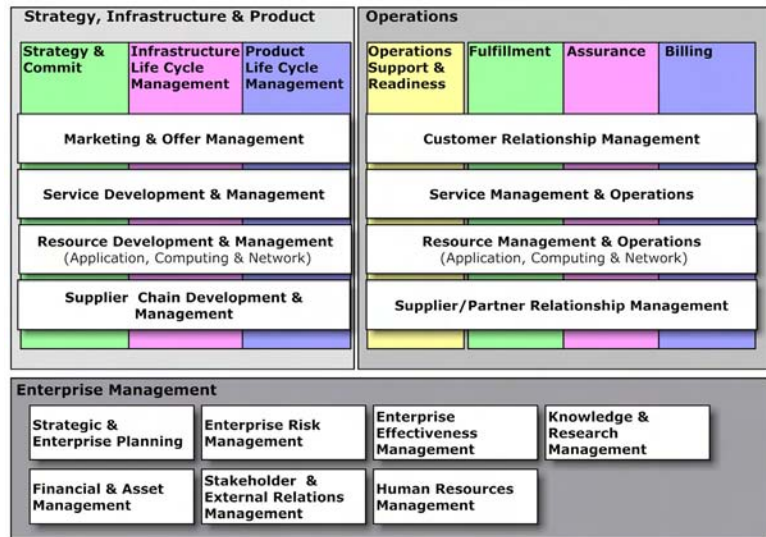


Abbildung 3.7: eTOM Level 1 in der Übersicht [TMF04a]

*Customer Relationship Management*

**Funktionale Prozessgruppierungen** Am oberen Ende der funktionalen Einteilung steht das *Customer Relationship Management* (CRM), das sich mit dem Aufbau, der Verbesserung und der Erhaltung von Kundenbeziehungen befasst. Es umfasst alle Aktivitäten und Subprozesse, die eine direkte Interaktion mit dem Kunden bzw. Nutzer der angebotenen Dienste beinhalten. Dazu gehören unter anderem Verkauf und Marketing der Dienste, die Überwachung der abgeschlossenen *SLAs*, die Rechnungsstellungs- und Inkasso-Management, aber auch die Entgegennahme von Störungsmeldungen und Überwachung des Reparaturfortschritts. In Abbildung 3.8 wird gezeigt, wie dieser Level 1 Prozess weiter in Level 2 Prozesse aufgeteilt wird.

*Level 3 bisher nur in Operations definiert*

Nur für die *Level 2* Prozesselemente des Operations Bereiches wird in der eTOM die Verfeinerung bis *Level 3* vollzogen und im *Addendum D* [TMF04f] dokumentiert. Diese Definitionen haben aber noch nicht die gleiche Reife wie die des *Level 2* und sind noch nicht vollständig mit den Prozessablaufbeispielen des *Addendum F* [TMF04g] (vgl. Abschnitt 3.2.2) abgeglichen. Die *Level 3* Zerlegung wird daher hier nur an einem Beispiel, dem *Problem Handling* gezeigt. In Abbildung 3.9 ist die Dekomposition des *Level 2* Prozesselementes *Problem Handling* in *Level 3* Prozesselemente dargestellt.

Wie auch in *Level 2* liefert die eTOM zu jedem *Level 3* Prozess bzw. Prozesselement eine kurze, in der Regel einen Absatz umfassende Beschreibung der jeweili-



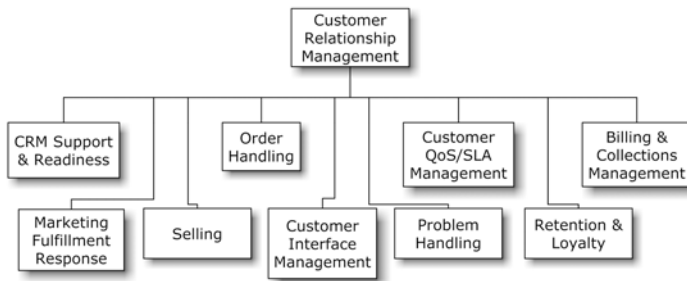


Abbildung 3.8: eTOM – Customer Relationship Management – Zerlegung in Level 2 Prozesselemente [TMF04f]

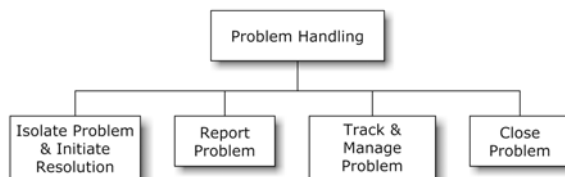


Abbildung 3.9: eTOM – Problem Handling – Zerlegung in Level 3 Prozesselemente

gen Aufgaben und Ziele. Die Aufgabe von *Isolate Problem & Initiate Resolution* ist die Entgegennahme von Problemberichten der Kunden (im Regelfall über das *Customer Interface Management*), deren Erstanalyse und die Einleitung der weiteren Schritte für eine Problemlösung. *Report Problem* erstellt Problemberichte und übermittelt diese an andere Prozesse und die Kunden (wiederum via *Customer Interface Management*). *Track & Manage Problem* verfolgt und überwacht die Aktivitäten der Problemlösung, während *Close Problem* für die Aktivitäten nach der Lösung des Problems, also z.B. Information des Kunden und Rücksprache mit diesem über seine Zufriedenheit mit der Lösung, zuständig ist.

In der Prozessgruppe *Service Management & Operations* (SM&O), vgl. Abbildung 3.10, stehen die verschiedenen Aspekte der angebotenen Dienste, wie Zugriff, Features, Konnektivität usw. im Mittelpunkt. Die Prozesse in dieser Gruppe interagieren im Sinne der Schichtung der eTOM weder direkt mit dem Kunden noch befassen sie sich direkt mit den physischen Ressourcen. „Service Management“ im Sinne der eTOM unterscheidet sich daher gravierend von der anderen Interpretationen dieses Begriffes, die meist das *Service Level Management*, d.h. die Aushandlung von SLAs, also direkten Kundenkontakt, als wesentliches Element beinhalten. Ebenso wenig ist das Design von Diensten Aufgabe dieser

*Service  
Management &  
Operations*

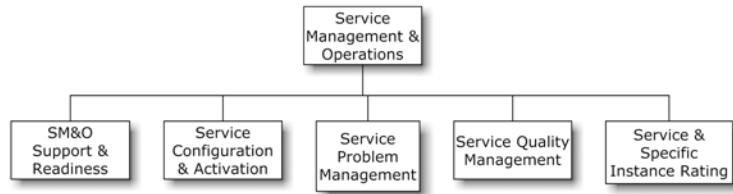


Abbildung 3.10: Service Management & Operations – Zerlegung in Level 2 Prozesselemente

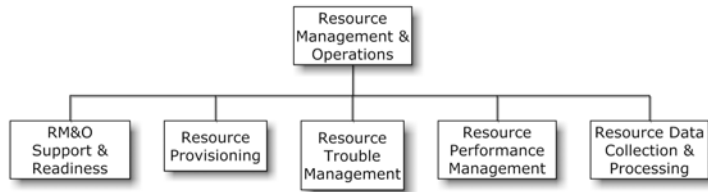


Abbildung 3.11: Resource Management & Operations – Zerlegung in Level 2 Prozesselemente

Prozessgruppe. Zwar ist das *Service Management & Operations* für die Bereitstellung zusätzlicher Dienstinstanzen für die Kunden verantwortlich und muss daher auch Maßnahmen der kurz- bis mittelfristigen Kapazitätsplanung übernehmen – das Neudesign neuer Dienstypen und die Berücksichtigung derer Anforderungen ist aber Aufgabe des *Service Development & Management* aus dem *Strategy, Infrastructure & Product* Bereich.

**Resource Management & Operations**

Im funktionalen Bereich des *Resource Management & Operations* (RM&O, siehe Abbildung 3.11) ist das Know-how zu den IT-Ressourcen (Anwendungen, Rechner, Netzwerkinfrastruktur) gebündelt. Seine Aufgabe ist das Management all dieser Ressourcen, ob sie nun direkt einem Service für einen Kunden zugeordnet sind oder zur allgemeinen Unternehmens-Infrastruktur gehören. Im Gegensatz zur entsprechenden Prozessgruppe in der TOM wird der Schwerpunkt nun nicht mehr im Netzwerkmanagement gesehen. Stattdessen soll ein integriertes Management von Applikationen, Systemen und Netzwerkkomponenten angestrebt werden.

**Supplier/Partner Relationship Management**

Das *Supplier/Partner Relationship Management* (S/PRM) (Dekomposition in *Level 2* Prozesse siehe Abbildung 3.12) war in der TOM nicht enthalten und ist ein erst mit der eTOM hinzugekommenes, neues Element. Es soll damit die verstärkte Ausrichtung auf eBusiness-Aktivitäten durch die Integration der unternehmensinternen Prozesse mit denen von Zuliefer- oder Partnerunternehmen

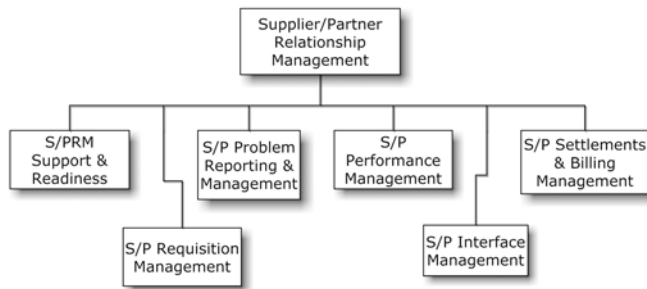


Abbildung 3.12: Supplier/Partner Relationship Management: Zerlegung in Level 2 Prozesselemente

berücksichtigen. Das Supplier/Partner Relationship Management befasst sich mit der externen Beschaffung von Produkten oder Diensten, der Qualitätsüberprüfung, der Meldung von aufgetretenen Problemen und der Abrechnung. Es ist somit in vielerlei Hinsicht eine Spiegelung des Customer Relationship Managements und interagiert mit den entsprechenden CRM-Prozessen, des Zulieferers/Partners, falls diese ebenfalls nach der eTOM geordnet sind.

Mit dieser Neuerung ist aber das Konzept der Schichtung in der eTOM weniger streng durchgesetzt als in der TOM. Durch die Einordnung des S/PRM am unteren Ende der Schichtung wird stillschweigend das vom TMN-Modell abgeleitete Prinzip der TOM, dass für jede Schicht die nicht angrenzenden funktionalen Schichten verschattet sind, aufgegeben. Mit der Integration von Diensten eines Partners in die eigene Dienstleistung ist beispielsweise das Service Quality Management nicht mehr nur auf Informationen des Resource Performance Management, sondern auch auf die Kommunikation mit dem S/P Performance Management angewiesen. Damit finden Prozessinteraktionen im Gegensatz zur TOM nun nicht mehr nur zwischen direkt benachbarten funktionalen Schichten statt. Das TMF arbeitet zur Zeit recht intensiv an der Problematik der B2B-Interaktionen, was auch das erst in der Version 4.0 neu hinzugekommene *Addendum B (Public B2B Business Operations Map)* belegt. In diesem Zusammenhang wird darauf hingewiesen, dass die S/PRM und CRM Prozessgruppen in den kommenden Versionen der eTOM einige Veränderungen erfahren könnten.

*Konzept der Schichten weniger stringent als in TOM*

**End-to-end Prozessgruppierungen** Wie in Abbildung 3.7 dargestellt, unterteilt die *Level 1* Sicht den *Operations* Prozess nicht nur horizontal in verschiedene funktionale Schichten, sondern orthogonal hierzu auch in die so genannten end-to-end Prozessgruppen. Während die im vorhergehenden Abschnitt untersuchte funktionale Unterteilung oft mit der Strukturierung eines Unternehmens in Fachabteilungen relativ deckungsgleich ist, entspricht die kunden- oder zielorientier-

*End-to-end Gruppierungen folgen dem gängigen Prozessbegriff*

te Gruppierung viel eher dem eigentlichen Gedanken des Prozessmanagements. Auch mit der prinzipiellen Auffassung von Prozessen in der ITIL stimmt dieses Konzept weit mehr überein, als die Einteilung von Prozessen zu funktionalen Einheiten, die in der ITIL, mit Ausnahme des *Service Desk*, nicht berücksichtigt wird.

**FAB-Prozesse** Die zentralen, bereits seit der TOM vorhandenen end-to-end Prozesse sind *Fulfillment*, *Assurance* und *Billing*<sup>14</sup> (*FAB Processes*). Jede dieser end-to-end Gruppierungen beinhaltet dabei Prozesselemente aus allen funktionalen Prozessgruppen des *Operations* (siehe Abbildung 3.13). Dabei können einzelne Prozesselemente zu mehr als einem end-to-end Prozess zugeordnet sein. Ein Beispiel hierfür ist das Customer Interface Management, das die zentrale und alleinige Schnittstelle für die Kommunikation mit dem Kunden bereitstellt und somit an alle FAB-Prozessgruppen Anteil hat. Die end-to-end Gruppierungen sind nicht Teil der durch die verschiedenen View Levels definierten Baumstruktur der eTOM, sondern ein spezielles Strukturierungsmerkmal, das nur im *Level 1* vorkommt.

**Fulfillment** Das *Fulfillment* bündelt alle Prozesselemente, die in direkter Weise dazu beitragen, dass Kunden die angeforderten Produkte bzw. Dienste entsprechend den im *SLA* festgelegten Vorgaben (Dienstfunktionalität, Zeitpunkt der Bereitstellung usw.) erhalten. Das *Fulfillment* umfasst also die primäre Wertschöpfungskette des Unternehmens, beginnend beim Einkauf von Ressourcen (und evtl. auch Teildiensten), über die interne Bereitstellung vorhandener Ressourcen und der Aktivierung von Diensten, bis hin zu deren Verkauf als Teile eines Produktes durch den *Selling* Prozess.

**Assurance** Der *Assurance* Prozess ist für die Sicherstellung der Dienstqualität verantwortlich. Er umfasst die reaktive Fehlerbehandlung in allen funktionalen Schichten, ebenso wie die Überwachung der QoS-Parameter und Ressourcenparametern, sowie die Ergreifung proaktiver Maßnahmen, wenn eine bevorstehende Unterschreitung von im *SLA* vereinbarten QoS-Levels droht.

**Billing** *Billing* schließlich ist mit der Sammlung von dienstbezogenen Nutzungsdaten und der Erstellung von Rechnungen auf Basis der im *SLA* dargelegten Vereinbarungen - d.h. beispielsweise unter Einberechnung von im Falle von *SLA*-Verletzungen anfallenden Pönalen - befasst. In seine Verantwortung fallen auch die Rechnungszustellung, die Beantwortung abrechnungsbezogener Fragen des Kunden, dessen Versorgung mit Abrechnungs- und Nutzungsdaten, das Inkasso sowie die Behandlung auftretender abrechnungsbezogener Probleme.

Die vier in Abbildung 3.13 aus Platzgründen nicht mehr dargestellten Level 2 Prozesselemente des *Operations*-Bereiches (*CRM-Support & Readiness*, *SM&O-*

<sup>14</sup>Wie auch aus Abbildung 3.7 ersichtlich, existieren noch vier weitere vertikale Prozesse (*Operations Support & Readiness*, sowie die drei end-to-end Prozesse des *Strategy, Infrastructure & Product* Prozessbereiches), welche jedoch z.Zt. in der eTOM nicht in dem selben Maße beschrieben und definiert sind - siehe auch Abschnitt 3.2.2.

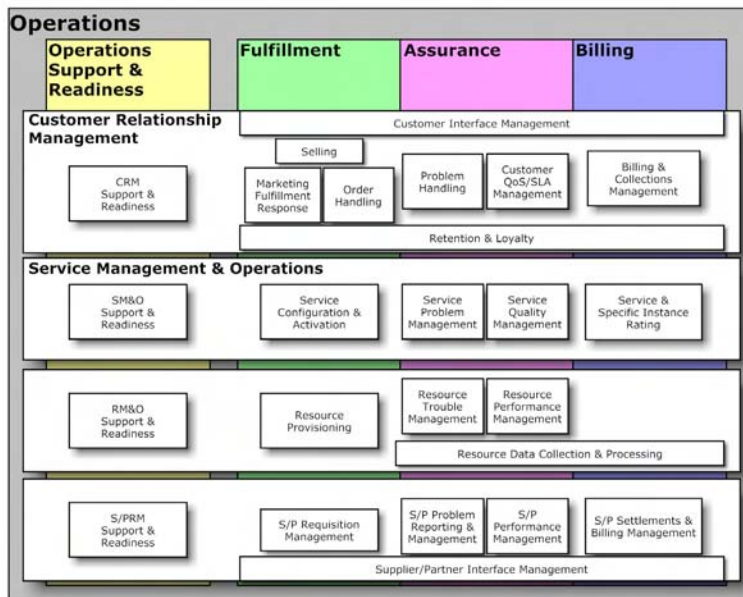


Abbildung 3.13: Zuordnung Operations Level 2 Prozesse zu den vertikalen end-to-end Prozessgruppierungen

Support & Readiness, RM&O- Support & Readiness, S/PRM Support & Readiness) sind der end-to-end Prozessgruppe Operations Support & Readiness zugeordnet. Diese Prozessgruppe soll die Prozesse des Fulfillment, Assurance und Billing unterstützen und steuern. Hierzu kann die Pflege entsprechender Datenbanken (z.B. Knowledge Bases), die Bereitstellung so genannter 'Customer Self Help Tools' (beispielsweise ein webbasiertes Diagnose- und Hilfesystem), aber auch die Planung des kurz- bis mittelfristigen Personaleinsatzes (z.B. die Besetzung des Call-Center) gehören. Die Prozesse dieser Gruppe sind grundsätzlich etwas längerfristig ausgerichtet als die vergleichsweise echtzeitorientierten anderen Operations end-to-end Prozesse. Die Level 2 Prozesse dieser Gruppe sind im Vergleich zu den übrigen Operations Prozesselementen auch relativ vage beschrieben und es wird von den Autoren der eTOM eingeräumt, dass sie sich in manchen Szenarios nicht immer klar von diesen trennen lassen.

*Operations  
Support &  
Readiness*

#### Process Flows

*konkrete  
Prozessausprä-  
gungen* Mit *Process Flows*, also Prozessabläufen oder -flüssen, bezeichnet eTOM die Ausprägung einer dynamischen Verkettung von Prozesselementen zu einem Gesamtprozess im klassischen Sinne mit einem auslösenden Ereignis und einem Endergebnis. Dies geschieht meist am prägnantesten entlang der end-to-end Prozessgruppen Fulfillment, Assurance und Billing. Prozessabläufe gibt es selbstverständlich nicht nur hier, sondern auch in allen andern Prozessbereichen, aber die Prozessabläufe in diesen Gruppen spiegeln das eigentliche Kerngeschäft eines Service Providers wider.

*Addendum F  
liefert nur  
Beispiele* Die eTOM sieht es nicht als ihre Aufgabe alle realistischen Prozessabläufe zu modellieren oder bestimmte Prozessabläufe vorzuschreiben oder auszuschließen. Vielmehr sind die Anwender der eTOM angehalten für ihre eigenen Services, Organisation und Geschäftsziele die Prozessabläufe (d.h. die Prozesssteuerung) selbst geeignet festzulegen. Im *Addendum F* [TMF04g] werden aber zur Illustration einige Prozessabläufe vorgestellt.

*Durchgang eines  
Beispiels* In Abbildung 3.14 ist ein solches Beispiel in einer vereinfachten Version dargestellt (man beachte, dass das Layout der Prozesselemente gegenüber Abbildung 3.13 um 90 Grad gedreht ist – der gesamte Ablauf findet innerhalb der end-to-end Prozessgruppierung *Assurance* statt). Das Beispiel zeigt, wie zunächst eine vom *Customer Interface Management* entgegengenommene Meldung (0) über die Störung eines Service an das *Problem Handling* weitergeleitet wird (1). Dieses holt zunächst zusätzliche Informationen beim *Customer QoS/SLA Management* und *Retention & Loyalty* ein um dieses Problem zu priorisieren. Das in diesem Rahmen informierte *Customer QoS/SLA Management* beginnt den Verlauf der Störung zu verfolgen um gegebenenfalls, wenn die Störung so lange anhält, dass eine Überschreitung im *SLA* vereinbarter Zeitgrenzen droht, entsprechende Maßnahmen (z.B. Einsatz von Ersatzgeräten) zu treffen. Das *Problem Handling* übermittelt dann die relevanten Daten, einschließlich der ermittelten Priorität und der im *SLA* vereinbarten Levels der betroffenen QoS-Parameter, an das *Service Problem Management*. Das *Service Problem Management* versucht nun das Problem zu diagnostizieren und fragt in diesem Zusammenhang den Status von mit diesem Dienst in Zusammenhang stehenden Ressourcen beim *Resource Trouble Management* nach. In diesem Beispiel wird kein Problem festgestellt und das *Service Problem Management* versichert sich beim *Service Quality Management*, dass der Dienst tatsächlich innerhalb der vereinbarten QoS-Levels, d.h. in ausreichender Qualität funktioniert. Nachdem dies bestätigt ist gibt das *Service Problem Management* seine Erkenntnisse an das *Problem Handling* zurück. Dieses informiert nun wiederum *Customer QoS/SLA Management* und *Retention & Loyalty*, welche den Abschluss des Problems vermerken, und informiert via *Customer Interface Management* den Kunden über die gegenwärtige Dienstqualität.

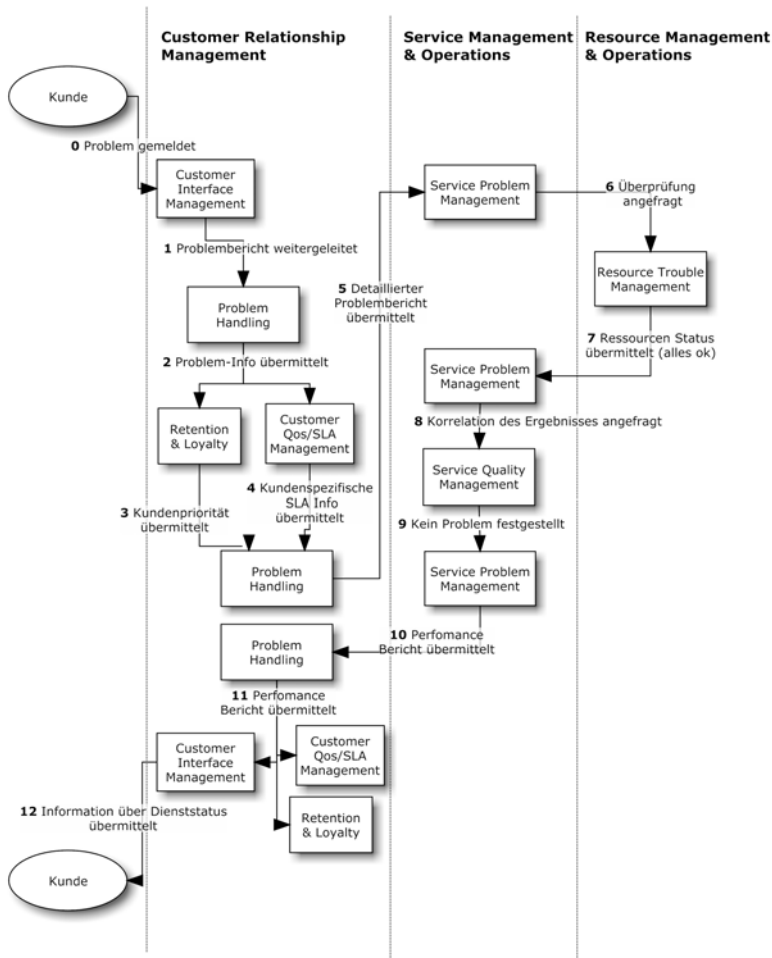


Abbildung 3.14: Beispiel für einen Prozessablauf in der Assurance-Prozessgruppe

*Prozesse verlaufen meist „senkrecht“* Wie gesagt, spiegelt dieses Beispiel weder die alleinig mögliche, noch die notwendigerweise beste Steuerung solch eines Ablaufes wider. Man erkennt jedoch recht deutlich, wie eine Ereignis (Meldung des Kunden), das zunächst ein Prozesselement der funktionalen Prozessgruppe *Customer Relationship Management* betrifft, eine Kette von Aktivitäten auslöst, die über das *Service Management & Operations* bis in das *Resource Management & Operations* reichen. Alle diese Aktivitäten finden aber innerhalb des end-to-end Prozesses *Assurance* statt. Leider gilt folgende Regel für die mittlerweile recht komplex gewordene eTOM in einem geringeren Ausmaß als dies bei der TOM der Fall war, dennoch trifft sie auf die meisten und wichtigsten Abläufe zu: Prozessflüsse bzw. -ketten laufen hauptsächlich entlang der vertikalen end-to-end Prozesse und eine, direkt auf eine andere folgende Aktivität liegt immer in der gleichen oder einer angrenzenden funktionalen Schicht (also kein Überspringen von Schichten).

#### Strategy, Infrastructure and Product Prozesse

*„SIP“-Prozesse* Die Prozesse des *Strategy Infrastructure & Product* Bereiches werden in funktionale Gruppierungen *Marketing & Offer Management*, *Service Development & Management*, *Resource Development & Management*, und *Supplier Chain Development & Management* eingeteilt (vgl. Abbildung 3.7). Wie beim *Operations* entspricht diese Einteilung wieder den funktionalen Prozessbereichen des *Level 0* (vgl. Abbildung 3.6).

*Zeithorizont der Prozesse von rechts nach links immer kleiner* Auch der SIP-Bereich hat eine zusätzliche vertikale Unterteilung in end-to-end Prozessgruppierungen. *Product Life Cycle Management*, *Infrastructure Life Cycle Management* und *Strategy & Commit* repräsentieren dabei Prozesse, deren Zeithorizont zunehmend (in der Abbildung also von rechts nach links) größer wird. Produkte ändern sich also häufiger als die Infrastrukturen, auf denen sie erbracht werden, und Infrastrukturen häufiger als Strategien. Eine genauere Definition und Beschreibung der SIP *end-to-end Process Groupings* und der entsprechenden *Process Flows* verspricht die eTOM für eine der kommenden Versionen. Auch auf die Unterteilung der funktionalen Prozessgruppierungen in Prozesselemente und deren Zuordnung zu den end-to-end Gruppen wird im folgenden nicht weiter eingegangen. Die *Level 2* Sicht des SIP-Bereiches ist offensichtlich momentan einem so schnellen Wandel unterzogen, dass ihre Darstellung innerhalb der eTOM 4.0 Dokumente zur Zeit nicht konsistent ist. Auch eine *Level 3* Verfeinerung ist derzeit nicht verfügbar.

#### Enterprise Management Prozesse

Das *Enterprise Management* wird in sieben (*Level 1*, vgl. Abbildung 3.7 auf Seite 58) Prozesse untergliedert, welche sich mit den in der *Level 0* Sicht (vgl.



Abbildung 3.6) unten angeordneten externen Entitäten bzw. Rollen sowie allgemeinen strategischen Fragestellungen des Unternehmensmanagements befassen.

Wie die eTOM selbst betont, sind die Prozesse des *Enterprise Management* allgemein in dem Sinne, als sie nicht spezifisch für ICSPs oder andere Service Provider sind. Eine Verfeinerung bis *Level 3* ist daher von vornherein nicht geplant - man möchte auch zunächst abwarten, ob sich für diesen Bereich nicht auch industrieunabhängige Modelle entwickeln, auf die man dann zurückgreifen könnte. Es ist auch nicht klar, ob dieser Prozessbereich eine Rolle im NGOSS-Projekt spielen wird, und wenn ja, wie diese aussehen könnte. Interessant im Vergleich mit der ITIL ist allerdings, dass sich die in der ITIL im Rahmen des Dienstmanagements behandelten Prozesse *Business Continuity Management* und *Security Management* hier, unter denselben Namen und auch mit in etwa äquivalenten Zielsetzungen, als Prozesselemente des *Enterprise Risk Management* wieder finden lassen.

*allgemeine,  
nicht industrie-  
spezifische  
Prozesse*

### 3.2.3 Abbildung auf ITIL

Für eine Konkretisierung der ITIL-Richtlinien lässt sich die eTOM nicht heranziehen – sie ist in den relevanten Punkten weniger detailliert. Ihre Stärke liegt eher in der klaren Struktur und der praktisch kompletten Abdeckung aller Unternehmensbereiche. Da beide Standards ihre Schwerpunkte so unterschiedlich gelegt haben, lassen sich beide prinzipiell, wie beispielsweise bei Grawe [Gra03] vorgeschlagen, parallel in einer Organisation implementieren.

*unterschiedliche  
Schwerpunkte*

Während von Seiten der OGC keinerlei Vergleich der ITIL mit der eTOM angestellt wird, bemüht sich das TM-Forum mit seiner *eTOM-ITIL Application Note* [TMF04h] einen Zusammenhang herzustellen und einen Weg aufzuweisen, wie beide Frameworks in Kombination miteinander eingesetzt werden können. Dort werden „correlations“ zwischen eTOM- und ITIL-Prozessen aufgezeigt, wobei sich allerdings eben verschieden starke Beziehungen mal zwischen einem eTOM-Prozess zu mehreren ITIL-Prozessen, mal zwischen einem ITIL-Prozess und mehreren eTOM-Prozessen ergeben. Es gibt auch eTOM-Elemente, welche sich nicht auf ITIL-Prozesse abbilden lassen – umgekehrt werden von den ITIL-Prozessen häufig nur vereinzelte Aspekte abgebildet. Eine konkrete und vollständige Abbildung dieser beiden Frameworks aufeinander, und damit ein sinnvoller kombinierter Einsatz, bleibt aber aufgrund der unterschiedlichen Charaktere dieser, oberflächlich in der Ausrichtung ähnlichen Standards schwierig.

*eTOM-ITIL  
Application  
Note*

Auch praktisch erscheint die gleichzeitige Umsetzung beider Modelle aber keine triviale Aufgabe: Alleine die Umsetzung der ITIL-Richtlinien stellt IT-Organisationen bereits vor erhebliche Herausforderungen und gelingt selten in kurzer Zeit in einem befriedigenden Umfang. Eine Restrukturierung nach eTOM dürfte, insbesondere für nicht primär mit Telekommunikationsdienstleistungen befass-

*gemeinsamer  
Einsatz  
problematisch*

ten Organisationen, eine noch wesentlich stärkere organisatorische Umstrukturierung, wie auch erheblich mehr Designaufwand bei der konkreten Prozessgestaltung erfordern. Zwar ergeben sich für den letzteren Punkt Effizienzen bei der gleichzeitigen Heranziehung der ITIL-Richtlinien, die punktuell durchaus geeignet erscheinen einzelne Bereiche der eTOM zu untermauern, dennoch erscheint der Aufwand beide Frameworks in ein unternehmensspezifisches Prozessdesign zu integrieren sehr hoch.

#### 3.2.4 Diskussion

*gleiche  
Zielsetzung wie  
ITIL aber  
unterschiedliche  
Grundkonzepte*

Trotz ähnlicher Zielsetzungen unterscheiden sich ITIL und eTOM stark voneinander. Im Prinzip wollen die ITIL und die eTOM das gleiche erreichen: Eine Basis standardisierter Geschäftsprozessdefinitionen mit einer gemeinsamen Nomenklatur für Service Provider, nur eben einmal für den Telekommunikations- und einmal für den IT-Sektor. Wenn man dies bedenkt, kann es schon erstaunen, zu welchem unterschiedlichen Ergebnissen hierbei das TMF und die OGC gekommen sind. Diese Unterschiede lassen sich nicht alleine durch die (mittlerweile meist gar nicht mehr so) unterschiedlichen Dienstangebote bzw. Geschäftsfelder der beiden Industrien erklären. Die Unterschiede der beiden Ansätze sind vielmehr in ihrer Herkunft, ihrer Historie und denen aus diesem Kontext heraus entstandenen Grundideen und -konzepten zu suchen.

*ITIL strebt nach  
Optimierung des  
Bestehenden*

Schwerpunkt der ITIL (vgl. Abschnitt 3.1) sind unterstützende Prozesse. Die ITIL ist aus den Erfahrungen des Betriebes von Rechenzentren und Endanwender-Infrastruktur in staatlichen Organisationen entstanden. Bei deren Betrieb mangelte es häufig an Effizienz und die Anwender waren mit den Leistungen oft unzufrieden. Also wurden aus allen Organisationen die so genannte *Best Practice* zusammengetragen, dokumentiert und damit auch eine gemeinsame Sprache für das IT-Dienstmanagement festgelegt. In der Hauptsache ging es also darum, bestehende Betriebsprozesse effektiver und effizienter zu gestalten. Was in der Konsequenz in der ITIL niedergelegt wurde, sind eben weniger Beschreibungen für die alltäglichen Betriebsprozesse, sondern in erster Linie Definitionen von Prozessen, die diese steuern und unterstützen. Viele alltägliche Prozesse von Service Providern sind in der ITIL nicht explizit berücksichtigt. So liefert die ITIL beispielsweise keine Richtlinien zu einem Prozess des Service Provisioning, d.h. was passiert, wenn ein Kunde einen Dienst (also eine neue Dienstinstanz) ordert. Zwar ist klar, dass der Auftrag vom Service Level Management Prozess angenommen wird, für die Definition aller weiteren Schritte bis der Dienst freigeschaltet ist, gibt es aber praktisch keine Anleitung durch die ITIL. Es ist lediglich klar, dass noch einige andere Dienstmanagementprozesse der ITIL daran beteiligt werden müssen.

Den *Process Flow* innerhalb der eTOM für solch einen Fall zu modellieren ist hingegen relativ leicht - im Großen und Ganzen braucht man nur die *Fulfill-*

ment Prozessgruppe durchlaufen. Die eTOM kann hier eine wesentlich bessere Vollständigkeit bei der Berücksichtigung der relevanten Prozesse und Aktivitäten vorweisen. Die eTOM eignet sich also prinzipiell, im Gegensatz zur ITIL, als Blaupause für die Neuordnung sämtlicher Aktivitäten einer IT-Organisation. Damit ist der Ansatz der eTOM wesentlich umfassender. Man kann argumentieren, dass die eTOM damit in der Tradition des Business Process Reengineering steht, während die ITIL eher in der Tradition des Total Quality Management der allmählichen, aber kontinuierlichen Verbesserung steht. Die eTOM eignet sich also gut zur Definition (oder Einordnung von bestehenden) Prozessen und deren Abgrenzung voneinander, was u.a. für die Schnittstellendefinitionen der NGOSS-Architektur benötigt wird. Allerdings sind Teile dieser eTOM-„Landkarte“ für viele IT-Dienstleister außerhalb des Telekommunikationsumfeldes nicht sonderlich relevant. Die starke Ausrichtung der eTOM auf QoS-Management und nutzungsbezogene Abrechnung zielt, zumindest zur Zeit, doch eher auf das Provisioning von Kommunikations-Diensten.

*eTOM ist umfassender...*

Geht es zunächst ganz pragmatisch darum, in das bestehende Management von IT-Infrastrukturen einzugreifen und dieses effizienter zu gestalten, so erscheint hierzu die ITIL in vielerlei Hinsicht die zielführenderen Richtlinien zu bieten. Viele in der ITIL behandelten Probleme, wie zum Beispiel die des *Change Management*, werden in der eTOM nicht adressiert. Man kann zwar prinzipiell zum ITIL *Change Management* gehörende Aktivitäten in verschiedenen Prozesselementen der eTOM wieder finden - ein klares Konzept für einen globalen Change Prozess liefert die eTOM aber, zumindest in der aktuellen Version, nicht. Auch zahlt die eTOM den Preis für ihre Vollständigkeit zur Zeit noch mit einer erheblich geringeren Tiefe bei der Prozessbeschreibung. Im Gegensatz zur ITIL enthält sie keinerlei Richtlinien für eine Umsetzung der Prozesse, die über eine allgemeine Beschreibung und die Dekomposition in feinere Prozesselemente hinaus gehen.

*...aber weniger pragmatisch als ITIL*

Die eTOM kann als ein formales, wenn auch in großen Teilen nicht sonderlich detailliertes Referenzmodell für Geschäftsprozesse des IT-Dienstmanagements angesehen werden. Mit SID existiert auch ein umfassendes, wenn auch bislang nur punktuell mit eTOM integriertes Modell der Datensicht. Als Modell ist eTOM umfassender, klarer strukturiert und erheblich formalisierter als die ITIL – der Preis hierfür ist aber, neben der aus Sicht eines allgemeinen Dienstmanagements immer noch spürbaren Ausrichtung auf Telekommunikationsdienste, vor allem ein Mangel an Detaillierung und konkreten Umsetzungsempfehlungen.

*eTOM stellt ein funktionales Referenzmodell dar*

### 3.3 CMM und CMMI

Die Reifegradmodelle CMM und CMMI stellen nicht Standards für IT-Dienstmanagement sondern für *Software Engineering* dar, werden aber hier dennoch vorgestellt, da ihre Grundkonzepte nachhaltigen Einfluss auf viele andere Rah-

menwerke zur Verbesserung von IT-Prozessen gehabt haben.

### 3.3.1 Herkunft und Zielsetzung

*Bewertung von  
Software-  
entwicklungs-  
organisationen*

Das von der *Carnegie Mellon University* (CMU) betriebene *Software Engineering Institute* (SEI) entwickelte Ende der 80er Jahre das *Capability Maturity Model* (CMM), eine Methode zur Beurteilung von Organisationen, die sich mit Softwareentwicklung befassen [Gli99]. Die Unterscheidung verschiedener Reifegrade und damit verbundener *Best Practice* soll Organisationen helfen, konkrete Verbesserungsmaßnahmen zu bestimmen.

Das CMM fand großen Zuspruch, so dass im Laufe der Jahre Anpassungen an andere Bereiche bzw. andere Arten von Organisationen entwickelt wurden und das ursprüngliche Modell in *Software CMM* (SW-CMM) umbenannt wurde. Am bekanntesten sind (neben SW-CMM) [MAS<sup>+</sup>03]:

- *People CMM* (P-CMM) (für Personalmanagement)
- *Software Acquisition CMM* (SA-CMM)
- *Integrated Product Development CMM* (IPD-CMM)
- *Systems Engineering CMM* (SE-CMM)

*Viele von CMM  
abgeleitete  
Rahmenwerke*

Dabei waren teilweise von CMM abgeleitete Standards ihrerseits wieder Basis für weitere CMM-Spezialisierungen. So war das SE-CMM Grundlage für verschiedene andere Modelle, beispielsweise das IPD-CMM. Einige CMM-Modelle wurden dabei nicht von der *Carnegie Mellon University*, sondern von anderen Organisationen entwickelt, wie das ebenfalls auf SE-CMM aufbauende *Systems Security Engineering Capability Maturity Model* (SSE-CMM) [SSE03]. Das ursprüngliche CMM war auch die Basis für das EU-Projekt *Bootstrap*, welches wiederum die Grundlage für *Software Process Improvement and Capability dEtermination* (SPICE) bildete [Sti99]. SPICE zielt dabei auf eine differenziertere Bewertung als CMM ab, im Prinzip sind sich beide Methodiken aber recht ähnlich. SPICE ist mittlerweile in einen ISO-Standard (ISO 15504) überführt worden.

*CMMI integriert  
mehrere ältere  
Modelle*

Eine teilweise Konsolidierung dieser im Lauf der Jahre entstandenen Vielzahl von Rahmenwerken [She97, She01] erfolgte 2002 als das auf SW-CMM, IPD-CMM und SE-CMM basierende *CMM Integration* (CMMI) vorgestellt wurde, welches die früheren Modelle ablösen soll [CKS03]. Es umfasst die Disziplinen *Systems Engineering*, *Software Engineering* (SE), *Integrated Product and Process Development* (EPD) und *Supplier Sourcing*.

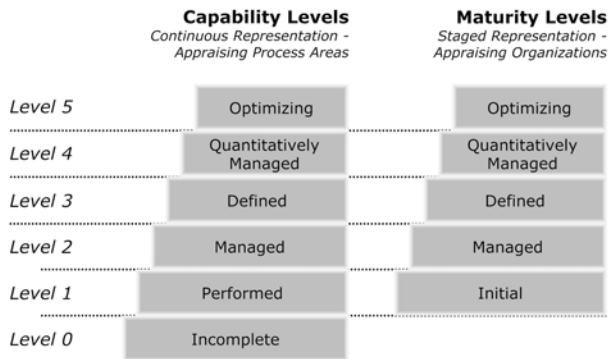


Abbildung 3.15: Reifegrade und Fähigkeitsgrade im CMMI

### 3.3.2 Struktur und Inhalte

Das CMMI enthält 22 *Process Areas* (Prozessbereiche), die verschiedene Management-Aufgaben adressieren – dazu kommen fünf *Generic Goals and Practices*. Jedem Prozessbereich sind *Goals* (Ziele), *Practices* (Praktiken) und *Subpractices* (Teilpraktiken) zugeordnet. Die Prozessbereiche des CMMI befassen sich hierbei mit Entwicklungsprozessen und deren Unterstützung, nicht mit dem Betrieb von IT-Infrastrukturen oder dem Dienstmanagement.

*Process Areas*

Einem Prozess oder einer *Process Area* einer Organisation aus dem Softwareentwicklungsbereich wird dabei, je nach dem Stand der Institutionalisierung und Steuerung der Prozesse, ein bestimmter *Capability Level* (Fähigkeitsniveau) zugewiesen. Der Begriff *Maturity Level* (Reifegrad) ist ähnlich definiert, bezieht sich aber immer auf eine Organisation als Ganzes. Das Erreichen eines höheren *Maturity Level* kann dabei die Umsetzung von bestimmten Prozessen in einem geforderten *Capability Level* zur Voraussetzung haben.

*Capability und  
Maturity Levels*

*Capability Level* ist daher ein Begriff der so genannten *Continuous Representation* des CMMI für Organisationen, die spezifische Ziele bei der Verbesserung bestimmter Prozessbereiche haben. Bei der *Staged Representation* hingegen wird durch die Zuordnung eines Reifegrades, auch ein Pfad für die weitere Qualitätsverbesserung in der Organisationen vorgegeben – durch die schnelle Bestimmbarkeit der konkreten Prozessbereiche, welche zur Erreichung des nächsten Reifegrades eingeführt beziehungsweise auf einen höheren *Capability Level* gebracht werden müssen.

*konkreter Ver-  
besserungspfad  
in der Staged  
Representation*

### 3.3.3 Abbildung auf ITIL

wenig direkte  
Berührungspunkte

Eine Abbildung auf ITIL macht für diese Standards wenig Sinn, da sie sich inhaltlich mit der Verbesserung von Prozessen für Entwicklungsorganisationen befassen. Für Betriebsorganisationen beziehungsweise das ITSM-Umfeld ließen sich nur die *Generic Goals and Practices* und einige wenige *Subpractices* verwenden. Die methodischen Ansätze sind aber weitgehend für die Schaffung ähnlicher Standards für andere Organisationstypen nutzbar.

### 3.3.4 Diskussion

Idee der  
Reifegrade oft  
nachgeahmt

CMM und seine Nachfolger haben nachhaltigen Einfluss auf das Denken in Bezug auf Prozesse in der IT gehabt. So ist beispielsweise das Konzept der *Capability Level* beziehungsweise *Maturity-Level* von verschiedenen anderen Standards wieder aufgegriffen worden. Dieses kommt auch im Umfeld der ITIL-Beratung häufig zur Anwendung, wohl um den Mangel zu adressieren, dass die ITIL zwar Richtlinien zum endgültigen Sollzustand bereitstellt, aber keinen konkreten Verbesserungspfad mit einer stufenweise Einführung der in ihr beschriebenen Praktiken bietet.

## 3.4 ITS-CMM

Der Erfolg des CMM-Ansatzes hatte zur Folge, dass auch eine Anzahl von Reifegradmodellen entwickelt wurde, zum Teil ohne Beteiligung des *SEI*, deren Anwendungsbereich außerhalb der Domäne des *Software Engineering* liegt. Ein solches Modell ist auch das *IT Service CMM (ITS-CMM)*, welches eine an das ursprüngliche CMM angelehnte Methodik auf den Bereich IT-Dienstmanagement anwendet.

### 3.4.1 Herkunft und Zielsetzung

Das ITS-CMM [NCTvV05] wird seit 1995 als Projekt wechselnder Firmen in Zusammenarbeit mit der Freien Universität Amsterdam entwickelt. Es soll eine strukturierte Alternative zu ITIL bieten – durch Verwendung des CMM wird ein Weg definiert, auf dem Organisationen in die Best Practices „hineinwachsen“ können.

### 3.4.2 Struktur und Inhalte

Das ITS-CMM hält sich an die gleiche grundlegende Struktur wie das Software-CMM des *SEI* (vgl. Abschnitt 3.3). Die Reifegrade sind also genau die gleichen

*Basis ist  
SW-CMM*

wie bei CMM bzw. CMMI (vgl. rechte Seite von Abbildung 3.15) . Im Prinzip wurden nur die entwicklungsspezifischen Prozessbereiche durch betriebsrelevante Prozessbereiche ersetzt. Das ITS-CMM legt, insbesondere in den Prozessbereichen der höheren Reifegrade, den Schwerpunkt weniger auf operative Prozesse, sondern auf Kontrollprozesse wie *Service Quality Management* oder *Quantitative Process Management*.

### 3.4.3 Abbildung auf ITIL

*keine direkte  
Abbildbarkeit*

Ein Vergleich zur ITIL wird im ITS-CMM [NCTvV05] selbst angestellt. Dabei wird unter anderem in einer Tabelle eine Gegenüberstellung von Prozessbereichen des ITS-CMM mit verwandten ITIL-Prozessen vorgestellt. Allerdings ergeben sich dabei kaum 1:1-Abbildungen. In der Philosophie des ITS-CMM, welches primär mit Planung, Steuerung und Qualitätsverbesserung befasst ist, sind viele der Prozesse, die in der ITIL vorgegeben und mehr oder weniger ausgearbeitet sind, von der ITSP-Organisation selbst zu definieren.

### 3.4.4 Diskussion

*interessanter  
Ansatz aber  
geringe  
Verbreitung*

Der Ansatz, ein Referenzmodell für IT-Dienstleister in eine CMM-Struktur einzubetten, ist vielversprechend und hat mittlerweile, beispielsweise bei ITIL-Assessments von Beratungsunternehmen, auch Nachahmer gefunden. Zwar definiert das ITS-CMM sinnvolle zusätzliche Prozesse, fügt aber der ITIL in Bezug auf eine klarere Darstellung oder Konkretisierung der Prozesse des *Service Support* und *Service Delivery* kaum etwas hinzu. Der Verbreitungsgrad des ITS-CMM ist bisher gering und es ist fraglich, ob das Projekt genügend Ressourcen und Fürsprecher besitzt, um in absehbarer Zukunft international eine größere Rolle zu spielen.

## 3.5 CobiT

*Control objectives for information and related Technology* (CobiT) verfolgt einen IT-Governance- oder Controlling-Ansatz. Es nimmt also eine deutlicher von außen auf die IT-Organisation blickende Perspektive, die eines IT-Auditors, ein.

### 3.5.1 Herkunft und Zielsetzung

*Control objectives for information and related Technology* (CobiT) wurde von der *Information Systems Audit and Control Association* (ISACA), einem Berufsverband der IT-Auditoren, und dem mit ISACA verbundenen *IT Governance In-*

*für IT-Auditoren*

stitute (ITGI) entwickelt. CobiT erschien 1996 in der ersten Auflage und liegt mittlererweile in einer vierten Version [ITG05b] vor.

### 3.5.2 Struktur und Inhalte

Die CobiT-Rahmenstruktur ordnet 34 Prozesse (beziehungsweise so genannte *High-Level Control Objectives* – die Bezeichnungen der Prozesse geben ihr jeweiliges Ziel wieder) je einer von 4 Lifecycle-orientierten Domänen (*Domains*) zu. Diese vier Domänen sind *Plan and Organize*(PO), *Acquire and Implement*(AI), *Deliver and Support*(DS) und *Monitor and Evaluate*(ME) (siehe Abbildung 3.16).

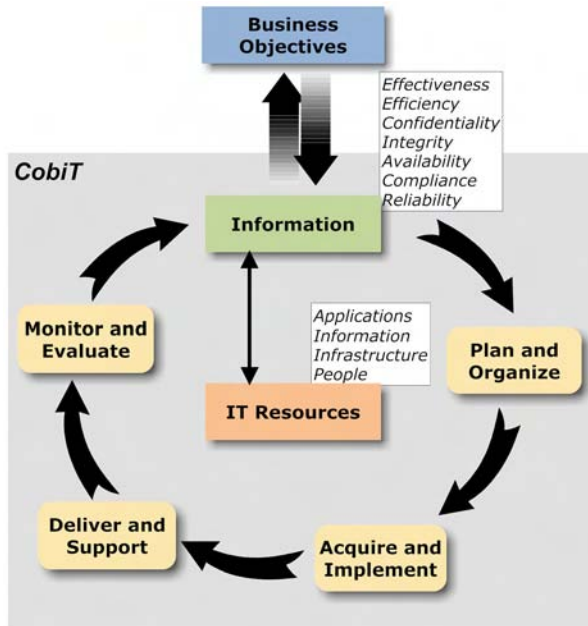


Abbildung 3.16: CobiT-Rahmenstruktur

Es werden dabei 10 PO-Prozesse, 7 AI-Prozesse, 13 DS-Prozesse und 4 Prozesse im Bereich *Monitor and Evaluate* (ME-Prozesse) definiert, denen jeweils ein aus dem Kürzel der Domäne und einer Nummer zusammengesetzter Identifier zugeordnet ist (also beispielsweise AI5 usw.). Die Beschreibung jedes Prozesses ist in vier Abschnitte gegliedert, wobei ein Abschnitt meistens nur eine Seite umfasst.

Zu jedem Prozess liefert CobiT zunächst einen Überblick und eine Einordnung. So wird, neben der Zuordnung zu einer der vier Domänen, angegeben, welche der



sieben *Information Criteria* (Informationskriterien)<sup>15</sup> durch den Prozess beeinflusst werden, welche der vier Arten von IT-Ressourcen<sup>16</sup> gesteuert werden und mit welchem von fünf Kernbereichen der IT-Governance<sup>17</sup> er verknüpft ist.

Ein zweiter Abschnitt enthält die so genannten *Detailed Control Objectives*, welche eine Verfeinerung des Prozesses darstellen (in etwa mit *Activities* in der ITIL vergleichbar).

Im dritten Abschnitt werden Tabellen zu verschiedenen Prozessaspekten (Input/Output, Verantwortlichkeiten, Ziele und Metriken), während der vierte Abschnitt ein Reifegradmodell mit sechs Niveaus<sup>18</sup> enthält, wozu eine für den jeweiligen Prozess spezifische Beschreibung der Reifegrade gehört. Der Begriff *Maturity Level* der Cobit bezieht sich also auf die Reife eines Prozesses (was eher dem *Capability Level* des CMMI entspricht).

### 3.5.3 Abbildung auf ITIL

Die OGC und ITGI bemühen sich um ein „Alignment“ von ITIL und CobiT. So wurde bereits für die Version 3 von CobiT ein gemeinsames Dokument [ITG05a] erstellt, das Zusammenhänge zwischen den Standards erläuterte und darlegen sollte, dass beide komplementär zu sehen sind, also gemeinsam eingesetzt werden können und nicht in direkter Konkurrenz zueinander stehen.

*gemeinsames  
„Alignment“  
durch OGC und  
ITGI*

In Cobit 4.0 kann man gegenüber CobiT 3 in der Abgrenzung der einzelnen Prozesse und der Verwendung von Begriffen eine erhebliche Annäherung zur ITIL erkennen. So waren Aufgaben das *Release Management* der ITIL in CobiT 3 teilweise AI5 (*Install and Accredite Systems*), teilweise AI6 (*Manage Changes*) zuzuordnen (vgl. die Beschreibung der entsprechenden *Detailed Control Objectives* in [ITG00]). In der aktuellen CobiT Version sind hingegen die zentralen Aufgaben und Konzepte des *Release Management* deutlich im neu geschaffenen AI7 (*Install and Accredite Solutions and Change* wieder zu erkennen<sup>19</sup>).

*CobiT und ITIL  
näheren sich  
aneinander an*

Aber auch wenn sich die überschneidenden Bereiche zunehmend gut aufeinander abbilden lassen – eine vollständige Abbildung ist in keine Richtung annähernd möglich. CobiT deckt einen wesentlich größeren Bereich ab, der Großteil der *Control Objectives* lässt sich nicht ohne weiteres in Richtlinien von *Service Support* oder *Service Delivery* wiedererkennen. Umgekehrt besitzt die Beschreibung der *Practices* nur eine vergleichsweise geringe Detaillierung. Was in ITIL in einem

<sup>15</sup> *Effektivität, Effizienz, Vertraulichkeit, Integrität, Verfügbarkeit, Compliance, Verlässlichkeit*

<sup>16</sup> *Personal, Anwendungen, Information, Infrastruktur*

<sup>17</sup> *Strategic Alignment, Value Delivery, Resource management, Performance Measurement, Risk Management*

<sup>18</sup> *Non-existent, Initial, Repeatable, Defined, Managed, Optimised*

<sup>19</sup> AI5 befasst sich in CobiT 5 nun mit dem Beschaffungsprozess

Kapitel abgedeckt wird, findet sich in CobiT meist auf zwei Seiten zusammengefasst – genauere Hinweise zur konkreten Gestaltung der Prozesse werden nicht gegeben. Zur Konkretisierung der ITIL-Richtlinien eignet sich CobiT daher im Allgemeinen nicht. In Bezug auf Kennzahlen zur Prozesssteuerung und zum Reporting kann es aber die ITIL durchaus sinnvoll ergänzen.

#### 3.5.4 Diskussion

*entstanden aus der Wirtschaftsprüfung*

CobiT hat, neben der ITIL, die wohl größte Verbreitung der in diesem Kapitel diskutierten Standards. Dass es aus dem Umfeld der Wirtschaftsprüfung entstammt, ist am Vokabular deutlich spürbar – beispielsweise an der unklaren Abgrenzung der Begriffe „Prozess“ und „Control Objective“, sowie am Gebrauch von „Information“, wo man im ITSM-Umfeld eigentlich von „IT-Service“ sprechen würde.

*übersichtliche Darstellung*

Die Darstellung der Inhalte ist übersichtlich und konsistent. Die Darstellung der ITSM-relevanten Inhalte bleibt jedoch insgesamt zu viel zu kurz um CobiT für eine Konkretisierung der ITIL-Prozesse heranziehen zu können. CobiT versteht sich aber auch weniger als Norm oder Vorlage für die Ausgestaltung von IT-Prozessen, sondern als Prüfwerkzeug.

### 3.6 eSCM

Das eSCM ist ein relatives junges Rahmenwerk, welches das Prinzip der Reifegradbewertung auf Outsourcing-Anbieter allgemein anwendet. Wenn es auch nicht spezifisch auf IT-Dienstleister zugeschnitten ist, so hat somit doch für IT-Service-Provider, zumindest solche, die ihre Dienste extern anbieten, eine größere Relevanz als das ebenfalls an der *Carnegie Mellon University* entwickelte und sehr viel bekanntere CMM.

#### 3.6.1 Herkunft und Zielsetzung

*Bewertung von Outsourcing-Anbietern*

Das *eSourcing Capability Model* (eSCM) richtet sich im Schwerpunkt an Kunden und Anbieter von externen erbrachten, also *outsourced* so genannten „IT enabled“ Services. „IT enabled“ bedeutet, dass die Schnittstelle für die Dienstleistung IT-basiert ist, die Funktionalität des Dienstes beim Outsourcing-Anbieter aber nicht vollständig automatisiert sein muss. eSCM wurde wie die CMM-Familie an der *Carnegie Mellon University* entwickelt, allerdings nicht am *SEI*, sondern an der *School of Computer Science*. Das eSCM versucht, die relevanten Kriterien von verbreiteten Qualitätsmanagement-Standards wie ISO 9001 und der CMM-Familie zu integrieren.

Die Version 1.0 des eSCM erschien 2001, darauf folgende Version 1.1 bereits im Jahr darauf [HKM<sup>+</sup>02]. Im Mai 2004 wurde eine Version 2 publiziert, welche zahlreiche inhaltliche und strukturelle Änderungen mit sich brachte und, minimal überarbeitet, 2006 als die Version 2.01 des *eSourcing Capability Model for IT-enabled Outsourcing Service Providers* (eSCM-SP) neu aufgelegt wurde [HHP06a, HHP06b]. Der nun längere Name hat seine Begründung in der Einführung eines neuen *eSourcing Capability Model for Client Organization* (eSCM-CL) [HL06a, HL06b], welches sich mit *Best Practice* für *Sourcing Management* auf der Kunden- bzw. Dienstnehmerseite befasst. Mit Version 2 ging die Entwicklung und Publikation des eSCM an das *Institute for Software Research International* (ISRI)<sup>20</sup>, die jüngsten Publikationen des eSCM (seit 2005) stammen mittlerweile vom, offenbar am ISRI speziell für die Pflege und Verbreitung des eSCM gegründeten *IT Services Qualification Center* (ITsqc)<sup>21</sup>.

viele  
Aktualisierungen  
in kurzer Zeit

### 3.6.2 Struktur und Inhalte

Das eSCM-SP v2 beinhaltet 84 *Practices*, die je einer von 10 *Capability Areas* zugeordnet sind. Bei den *Capability Areas* wird ihre Position im *Sourcing Life-Cycle* unterschieden, welcher in die Phasen *Initiation*, *Delivery* und *Completion* aufgeteilt ist. Die phasenspezifischen *Practices* finden sich in den vier *Capability Areas Contracting, Service Design & Deployment, Service Delivery* und *Service Transfer*. Praktiken, die nicht die nicht-phasenspezifisch sind, werden als *Ongoing Practices* bezeichnet und sind einer der sechs *Capability Areas Knowledge Management, People Management, Performance Management, Relationship Management, Technology Management* oder *Threat Management* zugeordnet.

84 Practices

Die Bewertungsmethode ist an die CMMI angelehnt und basiert auf *Capability Levels*. Wie bei CobiT (siehe Abschnitt 3.5) wurde dabei nicht genau der CMMI-Terminologie entsprochen, wo der *Maturity Level* organisationsweit definiert ist und der *Capability Level* ein Maß für die Qualität des Managements eines einzelnen Prozess ist. Hingegen wird im eSCM mit einem *Capability Level* die Organisationsreife bemessen. Die einzelnen *Capability Levels* des eSCM sind in Abbildung 3.17 dargestellt. Die *Capability Levels* 2-4 werden jeweils durch Umsetzung der ihnen zugeordneten Praktiken erreicht. *Capability Level 4* wird vergeben, wenn in einer Organisation alle 84 Praktiken umgesetzt sind. *Capability Level 5* wird (sozusagen automatisch) erreicht, wenn *Capability Level 4* über einen längeren Zeitraum nachgewiesen werden kann.

auch hier  
Bewertungsme-  
thode  
CMM-basiert

<sup>20</sup><http://www.isri.cmu.edu/>

<sup>21</sup><http://itsqc.cs.cmu.edu/>

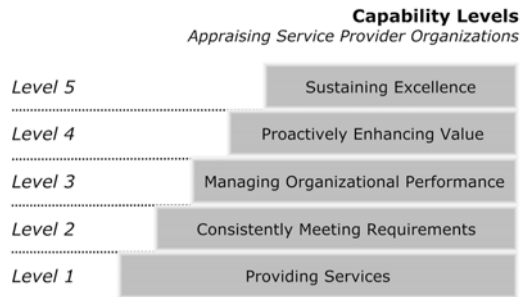


Abbildung 3.17: Fähigkeitsgrade im eSCM

### 3.6.3 Abbildung auf ITIL

*Positionierung  
zu BS 15000*

Es existiert zwar in der Literatur kein Vergleich zwischen eSCM und ITIL, ein solcher wird aber in einem von der CMU veröffentlichten dem Dokument [IDG<sup>+</sup>04] zwischen eSCM und an ITIL ausgerichteten BS 15000 (zu BS 15000 und dem Nachfolgestandard ISO/IEC 20000 vgl. Abschnitt 3.9) angestellt. Der Fokus liegt dabei auf der Frage, inwiefern die Qualitätsziele des eSCM bereits durch Erfüllung von BS 15000 erreicht werden. Zusammenfassend ist das Ergebnis, dass BS 15000, mit Ausnahme von *Practices* aus *Service Transfer*, welches überhaupt nicht adressiert wird, Anforderungen bezüglich der *Practices* aus allen anderen *Capability Areas* ganz oder teilweise erfüllt – allerdings wird dabei keine *Capability Area* vollständig abdeckt.

*eSCM sehr viel  
allgemeiner  
ausgerichtet als  
ITIL*

In der umgekehrten Richtung wird eine größere Abdeckung gesehen, also dass große Bereiche der von BS 15000 gestellten Anforderungen durch eSCM-Praktiken bereits abgedeckt seien. Inwiefern dies aber tatsächlich bei den eher IT-spezifischen Prozessen<sup>22</sup> tatsächlich in dem suggerierten Umfang der Fall ist, kann bezweifelt werden. So oder so, lässt sich das Ergebnis nicht auf die Beziehung von eSCM zu ITIL übertragen – für viele Inhalte in *Service Support* finden sich den eSCM-Praktiken, schon aufgrund derer allgemeiner Ausrichtung und der im Vergleich weniger umfangreichen und detaillierten Darstellung, kaum annähernd vollständige Entsprechungen.

### 3.6.4 Diskussion

Im Vergleich zur ITIL deckt eSCM einen recht großen Bereich ab, da es sich auch mit Fragen der allgemeinen Unternehmensorganisation beschäftigt und auf

<sup>22</sup>also in BS 15000 die *Relationship Processes* und die *Resolution Processes*, welche in der ITIL *Service Support* zuzuordnen wären, vgl. Abschnitt 3.9

viele Unternehmensbereiche abgedeckt

Outsourcing-spezifische Themen wie die Übergabe der IT-Systeme nach Ablauf des Outsourcing-Vertrags eingeht. Es bietet eine gute Strukturierung durch Zerlegung in relativ kleine Bausteine, die *Practices*. Allerdings lässt es sich nur unvollständig in Bezug zu den Konzepten der ITIL setzen lässt. Das eSCM zeigt somit zwar einen interessanten Weg zur Darstellung von *Best Practices* für ein allgemeineres Service-Management, ist aber eben nicht spezifisch für IT-Dienste ausgelegt.

## 3.7 MOF

Die Firma Microsoft enagiert sich seit Beginn dieses Jahrzents im ITIL-Bereich. So wurden die ITIL-Bände *Planning to Implement Service Management* und *Application Management* in Kooperation mit Microsoft erstellt [GM01]. Insbesondere ist die ITIL aber die Basis für ein Microsoft-eigenes Dienstmanagement-Framework, das *Microsoft Operations Framework* (MOF).

### 3.7.1 Herkunft und Zielsetzung

MOF [Mic04] ist eine von Microsoft herausgegebene Methodik zur Gestaltung des IT-Dienstmanagements, welches zu großen Teilen auf der ITIL aufbaut. Daneben finden sich in MOF auch Konzepte von *Microsoft Solutions Framework* (MSF), der Microsoft-Methodik für den Prozess der Softwareentwicklung und -verteilung, sowie (laut Microsoft) auch von ISO 15504 / SPICE (vgl. Abschnitt 3.3).

basierend auf ITIL, MSF und SPICE

### 3.7.2 Struktur und Inhalte

Die MOF-Dokumente umfassen *White Papers*<sup>23</sup>, Ausbildungsunterlagen, so genannte *Operations Guides* für ausgewählte Microsoft-Produkte und ein in mehreren Sprachen erschienener *MOF Pocket Guide* [PQA03]). Die erste Version von MOF erschien 2001, aktuell erscheint bereits die dritte Version (MOF 3.0) – allerdings liegen einige Dokumente zur Zeit immer noch ausschließlich in Version 2 vor.

MOF vereint unter seinem Dach, mehr oder weniger lose, drei Teilmodelle: Das *Process Model*, das *Team Model* und das *Risk Model*<sup>24</sup>. Hiervon lehnt sich das *Process Model* am stärksten an die ITIL an. Jeder Dienstmanagementprozess in der ITIL hat eine Entsprechung in einer *Service Management Function* (SMF). Das Risikomodell und Teammodell hingegen sind stark an das MSF angelehnt. Hinsichtlich der Integration des Team- und des Prozessmodells bleiben einige

drei Teilmodelle

<sup>23</sup>Die *White Papers* sind auf der Website <http://www.microsoft.com/mof/> frei erhältlich.

<sup>24</sup>in Version 3.0 in *Risk Discipline* umbenannt

Fragen offen, wenn dies auch in den Dokumenten der Version 3.0 mittlerweile besser adressiert wird.

*Prozessmodell ordnet Prozesse in einen IT Life Cycle ein*

Kern des MOF ist aber eindeutig das sehr stark am ITIL-Dienstmanagement orientierte Prozessmodell, dessen Aufbau in Abbildung 3.18 zu sehen ist. MOF ordnet hierbei die SMFs in einen (an den *Deming Cycle* erinnernden) *IT Life Cycle* ein, der in einen *Changing Quadrant*, einen *Operating Quadrant*, einen *Supporting Quadrant* und einen *Optimizing Quadrant* unterteilt ist. Dieses Konzept, dessen Grundlagen aus MSF übernommen sind, definiert auch vier *Operations Management Reviews*: das *Release Approved Review*<sup>25</sup>, das *Release Readiness Review* (in unregelmäßigen Intervallen, bei anstehenden Veränderungen stattfindend) sowie das *Operations Review* und *SLA Review* (beide in regelmäßigen Intervallen stattfindend).

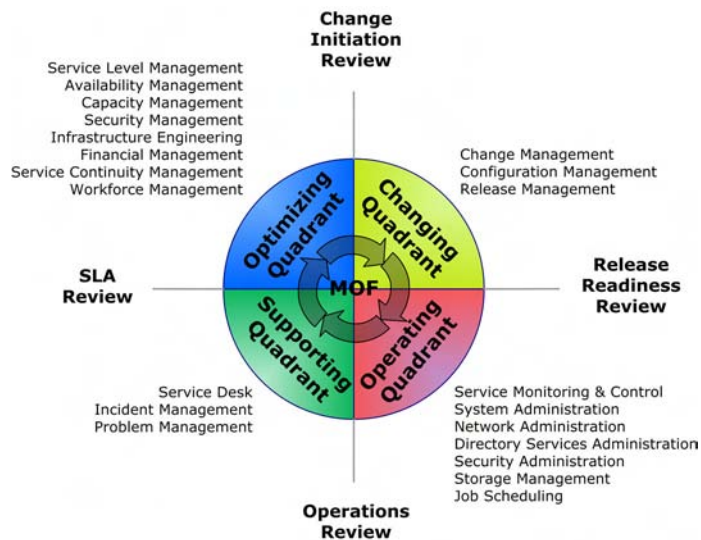


Abbildung 3.18: MOF Prozessmodell

### 3.7.3 Abbildung auf ITIL

Wie zu erkennen ist, sind die aus der ITIL *Service Delivery* bekannten Prozesse alle im *Optimizing Quadrant* untergekommen<sup>26</sup>, die Prozesse der *Service*

<sup>25</sup>In Version 3.0 umbenannt in *Change Initiation Review*

<sup>26</sup>In MOF Version 3.0 kommt hier das meist ebenfalls dem ITIL-Dienstmanagement zugerechnete *Security Management* hinzu

*Support*<sup>27</sup> verteilen sich auf den *Supporting Quadrant* und *Changing Quadrant*. Neu ist hier nur das *Workforce Management*, welches sich mit Fragen des Personalwesens befasst. Im *Operating Quadrant*, welcher sich mit dem alltäglichen, regulären Betrieb befasst, befinden sich allerdings administrative *SMFs*, die alle keine exakte Entsprechung in einem entsprechenden Dienstmanagementprozess der ITIL haben. In Version 3.0 kommt hier auch die *SMF Infrastructure Engineering* hinzu, um die Empfehlungen des ITIL *ICT Infrastructure Management* zu integrieren.

*Operating Quadrant hat keine ITIL-Entsprechung*

### 3.7.4 Diskussion

MOF übernimmt die wichtigsten ITIL-Konzepte und erweitert diese. Bei mancher Erweiterung aber, wie beispielsweise dem *Team Model*, erschließt sich in der gegenwärtigen Version nicht so ganz, wie diese mit dem ITIL-basierten Prozessmodell vernünftig zu integrieren sind, und ob sie wirklich einen nennenswerten Vorteil bringen.

Auch das Versprechen einer größeren Konkretheit (*Prescriptive Guidance*) unter Verzicht auf Technologieunabhängigkeit (durch Beschränkung auf Infrastrukturen mit Microsoft-basierter Software) wird zur Zeit noch nicht ganz eingelöst. Die *White Papers* zu den *SMFs* enthalten zwar teilweise klarere Handlungsanweisungen als die entsprechenden ITIL-Kapitel – und auch die eine oder andere Tabelle oder das eine oder andere nützliche Workflowdiagramm sind hinzu gekommen. Insgesamt sind diese Unterschiede aber noch gering und auch MOF noch so allgemein, dass es, im Gegensatz zur ursprünglich definierten Zielsetzung, durchaus noch als technologieunabhängig gelten kann.

*MOF erscheint in Teilen unfertig...*

Man merkt MOF an, dass viel daran gearbeitet wird, auch wenn das Thema in jüngster Zeit bei Microsoft wieder mehr in den Hintergrund zu treten scheint. Umfang und Qualität haben sich bisher mit jeder der bis vor kurzem beinahe im Jahreszyklus erscheinenden Versionen verbessert. Dass der Einsatz eines Rahmenwerkes für prozessorientiertes IT-Dienstmanagement lohnen kann, ist bislang am überzeugendsten für MOF nachgewiesen – allerdings ließe sich argumentieren dass die in [ER06] dargelegten Ergebnisse auch für ITIL übertragbar sind.

*...gewinnt aber mit jeder Version an Qualität*

Jedenfalls wird MOF, falls Microsoft sein Engagement aufrecht erhält, der ITIL ein höheren Bekanntheitsgrad bescheren – insbesondere in den USA, wo die ITIL bisher wenig verbreitet ist. Auf der anderen Seite weckt dies auch Befürchtungen, dass Microsoft eines Tages versuchen könnte, mit MOF die ITIL zu verdrängen [GM01] – für die nähere Zukunft erscheint dies allerdings unwahrscheinlich.

<sup>27</sup>inklusive der ITIL-„Function“ *Service Desk*

## 3.8 ITUP

Der *IBM Tivoli Unified Process* (ITUP) [IBM07] ist ein von IBM herausgegebenes Prozessmodell zum IT-Betrieb, welches teilweise die in der ITIL abgesteckten Dienstmanagementprozesse aufgreift, aber auch Erweiterungen enthält.

### 3.8.1 Herkunft und Zielsetzung

ITUP wird von IBM seit 2005 als Modell für IT Service Management angeboten. Zielsetzung scheint zu sein, einen Rahmen von Dienstmanagementprozessen zu schaffen, in den sich die Produkte der Tivoli-Familie einordnen lassen können.

### 3.8.2 Struktur und Inhalte

*ITUP-Tool* ITUP wird nicht als Dokument angeboten. Die Inhalte des Modells sind ausschließlich über ein (nach Registrierung) kostenfrei erhältliches Werkzeug, das *IBM Tivoli Unified Process Tool*<sup>28</sup>, zugreifbar, welches eine Browser-basierte Ansicht des „ITUP Method Content“ erlaubt.

*insgesamt 41 Prozesse* Im Rahmenwerk finden sich insgesamt 41 Prozesse in acht Kategorien, davon sind allerdings nur 18 Prozesse „core to IT Service Management“ und ausführlicher dargestellt. Zu jedem dieser Kern-Dienstmanagementprozesse findet sich eine allgemeine Beschreibung (*Description*), ein Workflow-Darstellung in einer lose an UML-Aktivitätsdiagramme angelehnten Notation (*Work Breakdown Structure*), eine Tabelle mit den im Prozess vorkommenden Rollen (*Team Allocation*) und ein eine Tabelle der Prozessartefakte (*Work Product Usage*).

### 3.8.3 Abbildung auf ITIL

*ITIL-Kernprozesse in Grundzügen erfasst* Alle zehn Dienstmanagement-Prozesse der ITIL finden sich auch in ITUP wieder, wenn auch in teilweise leicht abgeänderter Form. In der vergleichsweise knappen und prägnanten Darstellung in ITUP finden sich naturgemäß einige Inhalte der deutlich umfangreicheren ITIL-Prozessbeschreibungen nicht wieder, andererseits kommen an einigen Stellen auch Elemente, wie zusätzliche Rollen oder Aktivitäten, hinzu.

---

<sup>28</sup>eigentlich eine Sammlung verknüpfter HTML-Dokumente



### 3.8.4 Diskussion

In ITUP werden die Dienstmanagementprozesse der ITIL von einer der größten IT-Firmen in ihren Grundelementen aufgegriffen und in ein umfassendes prozessorientiertes Rahmenwerk des IT-Betriebs eingegliedert. Damit erinnert der grundlegende Ansatz stark an MOF (vgl. der vorhergehende Abschnitt 3.7).

*Parallelen zu MOF*

Das mittelfristige Ziel ist wohl, die Tivoli-Produkte in dieses Rahmenwerk einordnen zu können. Zuordnungen von Tools zu Prozessen finden sich in ITUP für einige Prozesse in der *Description*. Dies ist aber zur Zeit noch wenig detailliert und die umgekehrte Einordnung („welche Prozesse unterstützt ein Werkzeug?“) gestaltet sich vergleichsweise komplex nicht immer intuitiv. So sind beispielsweise die neuen *IBM Tivoli Process Manager* nicht immer Werkzeuge für den Prozess, mit dem sie einen Teil des Namens teilen – der *IBM Tivoli Availability Process Manager* (ITAPM) stellt beispielsweise nicht ein Werkzeug für das *Availability Management* im Sinne der ITIL dar, sondern laut ITUP eines von neun Werkzeugen, welche das *Incident Management* unterstützen<sup>29</sup> (vgl. auch [San06]).

*Ausrichtung der Tivoli-Produkte*

## 3.9 ISO/IEC 20000

Für das IT-Dienstmanagement existiert mittlerweile auch ein offizieller Standard der *International Organization for Standardization* (ISO). Aus einem auf ITIL basierenden britischen Standard entstanden, erlaubt es der neue Standard ISO/IEC 20000 Service-Provider-Organisationen, sich das Vorhandensein eines qualitätsorientierten IT-Dienstmanagements anhand eines minimalen Satzes von Kriterien zertifizieren zu lassen.

### 3.9.1 Herkunft und Zielsetzung

In den späten 1980er Jahren wurde von der *British Standards Institution* (BSI) Arbeiten an einem *Code of Practice* für IT Service Management unternommen, welcher unter der Dokumentnummer PD0005<sup>30</sup> veröffentlicht wurde [OGC00]. Die letzte Version erschien 1998 mit der Dokumentnummer PD0005:1998 und war bereits vor der Veröffentlichung der entsprechenden ITIL-Bände an den Begriffen und Prozessabgrenzungen des *Service Support* und *Service Delivery* der ITIL-Version 2 ausgerichtet.

<sup>29</sup>„ITAPM provides a solution that integrates processes and products around Incident Management.“ [IBM07]

<sup>30</sup>Später wurde statt dem *Code of Practice*, der dann Teil von BS 15000 war, allerdings der *Manager's Guide to Service Management* veröffentlicht. Nach einer Umstellung der Kürzel im BSI erscheint dessen aktuelle Version nun unter BIP 0005.

<i>Britische Standards als Ausgangspunkt</i>	PD0005 und Prinzipien des ISO 9000 Standards waren die Basis für einen ersten, 2002 vom BSI veröffentlichten, nationalen Standard BS 15000 für Service Management (BS15000:2000) [Dug04]. Ergänzt wurde dieser um ein <i>IT Service Management Self-assessment Workbook</i> PD 0015. Inhalt hiervon war ein Fragenkatalog, den Organisationen im Vorfeld einer BS 15000 Zertifizierung für eine erste Selbstbeurteilung verwenden konnten.
<i>Verbreitung von BS 15000</i>	Nachdem die Erfahrungen von „Early Adopters“ eingeholt worden waren, flossen diese in die zweite und letzte Version von BS 15000 ein, die 2002 und 2003 erschien [MG02][Dug04]. Während das angepasste <i>Self-assessment Workbook</i> PD0015:2002 ein eigenständiges Dokument blieb, umfasste der Standard nun neben der eigentlichen Spezifikation BS15000-1:2002 auch die Überarbeitung des <i>Code of Practice</i> BS15000-2:2003. Diese Version von BS 15000 wurde praktisch unverändert in Südafrika unter SANS 15000 [Dep04], sowie in Australien unter AS 8018 [Sta04] als nationaler Standard adaptiert.
<i>„Fast Track“ zum ISO-Standard</i>	Ende 2004 wurde schließlich ein Verfahren eingeleitet um auf Basis von BS 15000 in einem so genannten „Fast Track“ einen Internationalen ISO/IEC-Standard zu etablieren. Obwohl ISO/IEC-Standards eine ISO-Nummer erhalten, unterliegen sie nicht den Regelungen der reinen ISO-Standards, sondern denen des mit der <i>International Electrotechnical Commission</i> (IEC) gemeinsam gebildeten <i>Joint Technical Committee for Information Technology</i> (JTC 1) [Dug06]. Der internationale Standard für Service Management wurde durch das <i>Sub-Committee Software and System Engineering</i> (JTC 1 / SC 7) entwickelt und im Dezember 2005 unter der Nummer ISO/IEC 20000 veröffentlicht [ISO05a].

#### 3.9.2 Struktur und Inhalte

<i>zwei Teile</i>	Zwar ergaben sich bei der Adoption von BS 15000 zum ISO/IEC-Standard formal 450 Änderungen, hiervon sind allerdings nur sehr wenige inhaltlicher Art (vor allem kleinere Begriffsanpassungen), den überwiegenden Teil stellen kleine Änderungen an der formalen Struktur (insbesondere Nummerierung) dar [Dug06]. So ist auch der grundlegende Aufbau der gleiche geblieben und ISO/IEC 20000 besteht aus zwei Dokumenten: <b>ISO/IEC 20000-1</b> Part 1: Specification [ISO05b] <b>ISO/IEC 20000-2</b> Part 2: Code of Practice [ISO05c]
<i>Zertifizierungen</i>	Hierbei enthält nur der erste Teil (20000-1) die für eine Zertifizierung unbedingt zu erfüllenden Anforderungen, der zweite Teil (20000-2) hat nur empfehlenden Charakter. Zertifizierungen können von einem beim ITSMF akkreditierten <i>Registered Certification Body</i> (RCB) ausgestellt werden (in Deutschland z.B. TÜV Süd, DQS GmbH). Die BS 15000 begleitende Literatur, wie das <i>Self-assessment-workbook</i> ist nicht normativ und daher auch nicht internationalisiert worden –

angepasste Versionen sowie weitere ISO/IEC 20000-bezogene Literatur werden aber vom BSI in Zukunft angeboten.

Die beiden Teile des Standards sind analog aufgebaut:

*Aufbau der  
Dokumente*

- 1 Scope
- 2 Terms and Definitions
- 3 (Requirements for a) Management System
- 4 Planning and Implementing Service Management
- 5 Planning and Implementing New or Changed Services
- 6 Service Delivery Processes
- 7 Relationship Processes
- 8 Resolution Processes
- 9 Control Processes
- 10 Release Process

Die Inhalte von ISO/IEC 20000 ist an ITIL ausgerichtet („aligned“). Während sich also die Abschnitte 3 und 4 mit allgemeinen Qualitätsmanagement-Aspekten befassen und Abschnitt 5 sich mit Entwicklung, adressieren die Abschnitte 6-10 je eine Kategorie von Dienstmanagementprozesse. In Abbildung 3.19 ist die Zuordnung der einzelnen Prozesse zu diesen Kategorien abgebildet. Man sieht hieran, dass die grundlegende Struktur zwar an ITIL angelehnt ist, sich jedoch einige Abweichungen ergeben. Generell sind einige Prozesskategorien hinzugekommen, von der Aufteilung der ITIL in *Service Support* und *Service Delivery* nur letzteres erhalten. Dort sind *IT Service Continuity Management* und *Availability Management* aus der ITIL in einen Prozess zusammen gelegt. *Budgeting and Accounting for IT Services* ist in seinem Aufgabenumfang gegenüber dem *Financial Management for IT Services* der ITIL leicht beschnitten (eine mögliche Verrechnung der Dienste, das *Charging*, wird hier nicht berücksichtigt). Außerdem wird auch *Security Management* nun zu *Service Delivery* gerechnet (was allerdings auch dem Stand der aktuellen Prüfungs- und Ausbildungsrichtlinien für die ITIL-Personenzertifizierungen entspricht) und mit *Service Reporting*, welches sich primär mit dem Erstellen von *Service Level Reports* befasst, kommt ein Prozess hinzu der als solcher in der ITIL nicht definiert ist. Dies gilt auch für die beiden Prozesse *Supplier Management* und *Business Relationship Management*, welche den Bereich *Relationship Processes*<sup>31</sup> bilden.

*„aligned with  
ITIL“*

<sup>31</sup>in PD0005:1998 noch *Supplier Processes* genannt

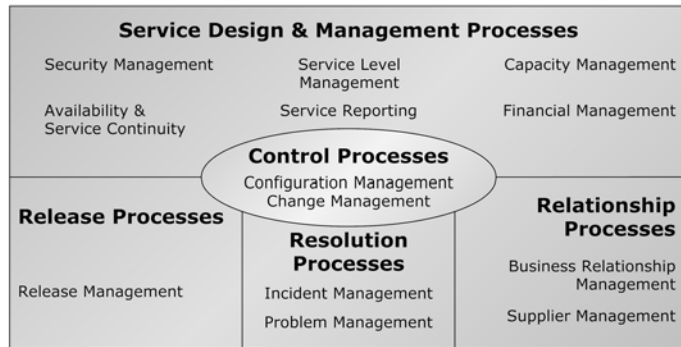


Abbildung 3.19: ISO/IEC 20000 – Service Management Prozesse [ISO05b]

### 3.9.3 Abbildung auf ITIL

„minimum set of requirements“

ISO/IEC 20000 basiert in vielen Punkten auf ITIL und hält sich weitestgehend an die gleichen Begriffe. Es enthält einige Richtlinien, die sich so nicht in der ITIL wieder finden. Die Neuerungen halten sich aber trotz ein paar hinzugekommener Prozessen und kleinerer Abänderungen in Grenzen. Inhaltlich betrachtet repräsentiert die ISO/IEC 20000 dabei eher eine Teilmenge der ITIL-Inhalte. Die Zielsetzung bei ISO/IEC 20000 ist, nur die notwendigen Kriterien (*minimum set of requirements*) für ein erfolversprechendes IT-Dienstmanagement darzustellen, was dann eben auf gerade mal 16 Seiten *Specification* geschieht (der *Code of Practice* umfasst 34 Seiten).

### 3.9.4 Diskussion

zu kurz um ohne ITIL verstanden zu werden

Bei der Kürze der ISO/IEC 20000 Dokumente scheint es kaum realistisch, dass ein Service Provider alleine aus ihnen sinnvolle Richtlinien für den Aufbau eines Service Management ableiten kann – dies wird in aller Regel immer nur mit einem Rückgriff auf ITIL als zusätzliche Quelle möglich sein. Der Einsatz von ISO/IEC 20000 kann aber dennoch in ITIL-Projekten in vielen Fällen nützlich sein. Vor allem bietet es natürlich für externe Service Provider die Möglichkeit sich die eigene Management-Qualität und Zuverlässigkeit mit einem wohlklingenden Siegel bestätigen zu lassen.

Zertifizierungen als möglicher Meilenstein in ITSM-Projekten

Das Vorhandensein einer solchen Zertifizierung kann in Zukunft im Outsourcing-Umfeld bei der Auftragsakquise ein Vorteil sein – möglicherweise wird sie bei mancher Ausschreibung auch zur Anforderung. Aber ISO/IEC 20000 scheint selbst für interne Service Provider interessant: Da es tatsächlich auf Basis der

umfangreichen und wenig strukturierten ITIL-Richtlinien einen minimalen Satz an übersichtlichen Anforderungen stellt, kann das Erfüllen der ISO/IEC 20000 Richtlinien der erste große Meilenstein in einem ITIL-Einführungsprojekt darstellen, dessen Erreichen auch noch relativ objektiv und vergleichbar mittels einer Zertifizierung nachgewiesen werden kann. Auf jeden Fall ist aber damit zu rechnen, dass die Etablierung eines internationalen, von der ISO mit getragenen Standards der auf ITIL basiert, dieser zu weiterer Popularität verhelfen wird.

### 3.10 Weitere Frameworks

Es folgen kurze Erläuterungen zu einigen anderen Arbeiten und Standards, welche zwar nicht Rahmenwerke für Dienstmanagementprozesse darstellen, dennoch aber in deren Zusammenhang häufig genannt werden.

Da ist zunächst die *Application Services Library* (ASL) [vdP04] der *ASL Foundation*<sup>32</sup>. Sie bemüht sich um eine Integration des Dienstmanagements mit dem *Application Management*. In der Definition der ASL liegt der Schwerpunkt des *Application Management* allerdings weniger auf der effizienten und bedarfsge rechten Neuentwicklung von Software (vgl. *Application Management* [OGC02a] in der ITIL, siehe Abschnitt 3.1.2) oder der Steuerung laufender Applikationen (vgl. beispielsweise die Verwendung des Begriffs *Application Management* in [KKS01]), sondern in der Wartung, Verbesserung und Renovierung bestehender Anwendungen.

ASL

Die *ASL Foundation* hat mittlerweile auch die Pflege der *Business Information Services Library* (BiSL)<sup>33</sup> übernommen. Allerdings liegen relevante Publikationen bislang nur auf niederländisch vor (z.B. [vdPDvO05]). Die BiSL verfolgt einen Ansatz zum unternehmensweiten Informationsmanagement und einer besseren Abstimmung verschiedener Management-Bereiche. Hierbei erinnert das Grundmodell stark an das *Strategic Alignment Model* von **Henderson** und **Venkatraman** [HV93]. ASL und BiSL haben bislang international keine große Verbreitung erfahren, werden aber häufig in Zusammenhang mit ITIL-basiertem IT-Dienstmanagement genannt. Dies mag vor allem daran liegen, dass beide im Kontext einer sehr aktiven niederländischen ITSM-Community – zu der neben *EXIN* auch der Verlag *Van Haren* und zahlreiche Beratungsfirmen wie *PinkRocade*<sup>34</sup> zählen – entstanden sind.

BiSL

Ebenfalls vor allem dank seiner Herkunft häufig in Zusammenhang mit ITIL gebracht, ist die zweite Version der Methodik für *Projects In Controlled Environments* (Prince2) [OGC06b]. Es handelt sich hierbei um eine allgemeine Projektmanagement-Methode ohne direkten IT-Bezug. Allerdings ist Prince2, wie die

Prince2

<sup>32</sup><http://www.aslfoundation.com/>

<sup>33</sup><http://www.bisl.nl/>

<sup>34</sup><http://www.getronicspinkroccade.nl/>

ITIL, ursprünglich von der CCTA entwickelt worden und wird mittlerweile von der OGC verwaltet. Als die „hauseigene“ Projektmanagement-Methode der OGC findet sie daher in der ITIL mehrfach Erwähnung.

#### ARIS und UML

So wie im Prozessmanagement allgemein, kommt auch bei Prozessen im IT-Dienstmanagement immer wieder die Sprache deren Modellierung. Wie schon in Abschnitt 2.5 diskutiert, sind ARIS und UML hierfür gängige Modellierungsmethoden. Während das Ziel der Erstellung von ARIS-Modellen aber immer schon die Beschreibung von Prozessen war, entspringt die von der *Object Management Group* (OMG) standardisierte UML ursprünglich dem Bereich des *Software Engineering*. UML ist also nicht für das Prozessmanagement im engeren Sinn entworfen, sie wird aber, teils unter Rückgriff ihrer Standard-Diagrammtypen, teils unter Verwendung spezieller Erweiterungen (*Profiles*, z.B. in [EP98], [OWS<sup>+</sup>03]), zunehmend auch für die allgemeine Prozessmodellierung eingesetzt. Weder ARIS noch UML sind also ITSM-spezifisch. Es gibt daher, beispielsweise im Rahmen eines Arbeitskreises des deutschen itSMF, Ansätze für die Entwicklung speziell auf die ITIL zugeschnittener Modellierungsansätze [PS06], deren Ergebnisse bislang aber nicht veröffentlicht sind. Inwiefern eine domänenspezifische Modellierungssprache für IT-Dienstmanagement notwendig ist, ist bislang allerdings eine kaum untersuchte Frage. Ein Ansatz zur Modellierung ausgewählter ITIL-Prozesse unter Verwendung existierender Modelltypen wird im folgenden Kapitel vorgestellt.

#### XML-basierte Prozess- beschreibungssprachen und BPMN

Im Umfeld der Forschung zu SOAs (vgl. Abschnitt 2.1) wurden ab ca. 2002 zum Zwecke der prozessorientierten Verknüpfung von Web-Services zahlreiche, meist XML-basierte Sprachen zur Beschreibung von Workflows entworfen. Viele Dialekte sind mittlerweile schon wieder obsolet oder in Nachfolgerstandards aufgegangen – prominentere Beispiele aktueller Sprachen sind die *Business Process Execution Language* (BPEL)<sup>35</sup> [ACD<sup>+</sup>03], welche unter anderem von IBM und Microsoft propagiert wird, sowie die *XML Process Definition Language* (XPDL) der *Workflow Management Coalition* (WfMC) [WFM05]. Um für auf der Ausführung von Web-Services basierenden Prozesse auch eine für Menschen vergleichsweise intuitiv verstehbare Darstellungsform zu schaffen, ist von der *Business Process Management Initiative* (BPMI) die *Business Process Modeling Notation* (BPMN) [OMG06] als grafische Modellierungssprache entwickelt worden<sup>36</sup>. Sie ist aber auch unabhängig von XML-basierten Prozessbeschreibungssprachen einsetzbar<sup>37</sup>. Da BPMI nun mittlerweile mit OMG verschmolzen ist, besteht momentan die kuriose Situation, dass die OMG zur Prozessmodellierung nun gleichzeitig zwei unterschiedliche Standards anbietet. Eine Abbildung zwischen diesen

<sup>35</sup> je nach Version auch BPEL4WS (Business Process Execution Language for Web Services) oder WS-BPEL genannt

<sup>36</sup> Ursprünglich wurde BPMN als grafische Notation zur XML-basierten *Business Process Modeling Language* (BPML) entworfen. BPML stammt ebenfalls von BPMI, wird aber anscheinend nicht mehr, oder nur als Speicherformat für BPMN-Modelle, von der OMG weiter entwickelt.

<sup>37</sup> Für eine Anwendung der BPMN zur Modellierung eines Dienstmanagements siehe [Cla06]

wird wohl erst das zur Zeit bei der OMG in Entwicklung befindliche *Business Process Definition Metamodel* (BPDM) ermöglichen.

Auf Konzepte des *OSI Management* für Datennetze [ITU92] wird häufig im Kontext des IT-Managements Bezug genommen. Obwohl die Qualität der dort dargelegten Konzepte unumstritten ist, hat ihre vergleichsweise hohe Komplexität eine größere Verbreitung von Implementierungen verhindert (vgl. Abschnitt 3.2.1). Dies blieb im Bereich des Netz- und Systemmanagements bislang dem *Internet-Management* vorbehalten, welches Konzepte des *OSI Management* stark vereinfacht und dessen Kernstück des ist das *Simple Network Management Protocol* (SNMP) [CFSD90] darstellt.

*Network  
Management  
Standards*

Der Begriff *Service Management* kommt, wie bereits in Abschnitt 1.1.1 diskutiert, auch im Netz- und Systemmanagement vor und findet dort wahrscheinlich schon mindestens ebenso lange Verwendung wie im Zusammenhang mit IT-Prozessen. In der Forschung zur *Quality of Service* von Netzwerkdiensten sind dabei auch zahlreiche QoS-unterstützende Architekturen für Rechnernetze entstanden (für Überblick und Diskussion siehe z.B. [Roe05]). Das in Abschnitt 2.9 angesprochene MNM-Service-Model erweitert und ordnet Konzepte dieser Sicht auf das Management von Diensten, indem Aspekte der Dienstbereitstellung und des Dienstmanagements voneinander abgegrenzt und Verantwortlichkeiten in einer IT-Dienstleistungskette definiert werden. Es stellt also einen Einordnungsrahmen für die technischen und organisatorischen Komponenten des IT-Service-Provisioning zur Verfügung. Einen allgemeineren, eher dem *Software Engineering* zuzuordnenden und sehr umfassenden Einordnungsrahmen für Modelle eines *Enterprise System*, also alle Aspekte eines durch IT-Systeme unterstützten Geschäftsprozesses, stellt das *Zachman Framework* dar, welches im folgenden Kapitel noch näher vorgestellt wird.

*QoS-  
Architekturen,  
MNM-  
Dienstmodell  
und Zachman  
Framework*

### 3.11 Einordnung und Ausblick

Wie in den vorhergehenden Abschnitten gesehen, existiert also eine große Zahl von Rahmenwerken und Standards für prozessorientiertes IT-Management, zwischen denen erhebliche, wenn auch meistens nicht explizit gemachte, Unterschiede in Bezug auf Zielsetzung, Präsentation und angewandte Methoden bestehen. Eine Herausarbeitung der Gemeinsamkeiten, Unterschiede und Beziehungen zwischen den ITSM-Rahmenwerken wurde bisher nur punktuell versucht.

Im nächsten Abschnitt (3.11.1) werden die generelle Zusammenhänge zwischen ITSM-bezogenen Rahmenwerken diskutiert. In Abschnitt 3.11.2 erfolgt eine überblicksmäßige Einordnung dieser, welche die Grundlage für eine Werkzeugklassifikation im nächsten Kapitel bilden wird.

### 3.11.1 Beziehungen zwischen ITSM-relevanten Rahmenwerken

**Abgrenzungen und Einordnungen gegenüber anderen Frameworks** Wie teilweise bereits in den Ausführungen zu den jeweiligen Standards ausgeführt, existieren von einigem der herausgebenden Organisationen Versuche das eigene Rahmenwerk gegenüber anderen zu positionieren. Dies geschieht vor allem gegenüber der ITIL, was wohl auf die große Verbreitung und momentane Popularität dieser zurückzuführen ist. Dabei wird meist versucht darzustellen, dass das jeweilige Framework nicht eine Alternative, sondern eine Ergänzung zu ITIL darstellt – dass also der Einsatz des betreffenden Standards keine Entscheidung gegen ITIL darstellen muss, sondern ein gemeinsamer Einsatz möglich ist.

#### Bestehende Vergleiche und Einordnungen

**eTOM und ITIL** So wird in der *eTOM-ITIL Application Note* des TM-Forums [TMF04h] argumentiert, dass sich ITIL-Prozesse mit Hilfe der eTOM-Prozesselemente modellieren lassen und somit beide Frameworks zusammen eingesetzt werden können (vgl. Abschnitt 3.2.3). Allerdings wird eingeräumt, dass ein eins-zu-eins Mapping nur in den wenigsten Fällen möglich ist und die Korrelation zwischen eTOM-Prozesselementen und ITIL-Prozessen unterschiedlich stark ausfällt. So bleibt die Abbildung recht vage und es wird weder ein überzeugendes Beispiel geliefert das den Nutzen eines solchen kombinierten Einsatzes unterstreichen würde, noch nachgewiesen dass dies für alle Dienstmanagementprozesse der ITIL hinreichend detailliert möglich ist.

**CobiT und ITIL** Nur zwischen ITIL und CobiT existiert eine symmetrische, von beiden beteiligten Seiten gemeinsam erarbeitete, gegenseitige Positionierung. Wie in Abschnitt 3.5.3 erwähnt, gibt es zu CobiT Version 3 und ITIL ein „Mapping“ [ITG05a], welches von den beiden Standard-Organisationen, der OGC und dem *IT Governance Institute*, zusammen herausgegeben wurde. Betrachtet man, wie sich die neue Version 4.0 der CobiT mittlerweile sehr viel stärker in den in diesem Mapping aufgezeigten Überschneidungsbereichen an ITIL-Begriffen orientiert, so scheint eine Koexistenz und Abbildbarkeit zwischen ITIL und CobiT von beiden Seiten auf längere Sicht erwünscht zu sein.

**Positionierung von eSCM** Die aus dem Umfeld der CMU stammenden Herausgeber des vergleichsweise jungen und in der Zielsetzung breiteren eSCM bemühen sich um eine Positionierung ihres Rahmenwerkes gegenüber mehreren verschiedenen anderen Standards, unter anderem auch mehreren der hier betrachteten ITSM-Standards (vgl. Abschnitt 3.6.3). So wird eSCM zwar nicht mit ITIL verglichen, wohl weil für dieses keine Organisationsbewertung oder -zertifizierung vorgesehen ist, aber mit CobiT und dem auf ITIL aufbauenden BS 15000 (dem Vorgänger von ISO/IEC 20000).

**Positionierung von ITUP** Auch ITUP enthält Abbildungen (*Process Mappings*) zu verschiedenen anderen prozessorientierten IT-Standards. Die Abbildung erfolgt genau genommen



aber nicht zu ITUP, sondern zu Prozessen des *IBM Process Reference Model for IT* (PRM-IT). Dieses stellt eine Sammlung der Prozessmodelle dar, die ITUP und anderen IBM-Rahmenwerken zugrunde liegen. Somit kann auch eine Abbildung auf Rahmenwerke des *Software Engineering* wie CMMI erfolgen, aber eben zu Prozessen des breiteren PRM-IT und nicht zu Kernprozessen des Dienstmanagements, wie sie in ITUP detailliert werden. Die konkreten Abbildungen beschränken sich auf eine tabellarische Zuordnung jeweils überschneidender oder sich partiell entsprechender Prozesse, gehen also nicht sonderlich ins Detail.

Von der ASL Foundation gibt es den Versuch einer Platzierung der ASL in das Dickicht der Dienstmanagement-relevanten Frameworks, dort „forest of models“ genannt [ASL05]. Allerdings ist dieses eher eine allgemeine Aufzählung und Beschreibung der verschiedenen Standards – eine Klarstellung von Unterscheidungskriterien oder Einordnung wird nicht geliefert – es bleibt bei einer Beschreibung der allgemeinen Zielsetzung und Herkunft der verschiedenen Standards. Das gleiche trifft auch auf die meisten allgemeinen Übersichtsartikel zu (beispielsweise [Sal04]). Nur Modelle des *Software Engineering* und IT-Dienstmanagement werden, jeweils anhand dessen welche Phasen im Lebenszyklus eines IT-Systems sie abdecken, in [Kni06] und [WK06] nach klar genannten Kriterien verglichen.

„Forest of Models“ von der ASL Foundation

### Übernahme existierender Konzepte

Eine stärkere Beziehung zwischen zwei Standards als eine im nachhinein erstellte Abbildung stellt natürlich das Aufgreifen von Konzepten oder Methoden dar. Gemessen an der großen Zahl der existierenden Rahmenwerke geschieht diese aber nur selten. Eine explizite Übernahme durch ein anderes Rahmenwerk wird – außer von den Inhalten des direkten Vorgängers des jeweiligen Standards – nur von einigen Konzepten der ITIL und dem bereits Anfang der 1990er erschienenen CMM vorgenommen. Auf ITIL-Grundlagen bauen sowohl BS 15000 wie ISO/IEC 20000 auf, ebenso MOF. An den Konzepten von CMM orientieren sich eSCM und ITS-CMM. Auch CobiT übernimmt die zentrale Idee der *Maturity Levels* von CMM, wenn es auch die Zuordnung von Reifegraden zu Prozessen weniger formal beschreibt und über kein CMM entsprechendes Modell zur Bestimmung der Organisationsreife verfügt.

Aufgreifen vorhandener Konzepte selten

Ansonsten scheinen sich die verschiedenen Standards, abgesehen von den erwähnten Abbildungen, die überwiegend der einseitigen Abgrenzung gegenüber des eigenen Rahmenwerks gegenüber einem populäreren zu dienen scheinen, eher unabhängig und ohne große Kenntnisnahme voneinander zu entwickeln. Am erstaunlichsten unter diesem Blickpunkt ist, dass sich keiner der prozessorientierten Dienstmanagement-Standards einer gängigen, genormten Methode der Prozessbeschreibung bedient. Kein ITSM-Rahmenwerk verwendet also beispielsweise UML, ARIS oder eine andere etablierte Modellierungssprache für die

keine einheitlichen Darstellungsmitel

Darstellung der in ihm definierten Prozesse. Dabei hat sich die Verwendung von UML als einheitliche Darstellungsform mittlerweile in verschiedenen Bereichen der Informatik bewährt – beispielsweise bei der Dokumentation von Entwurfs- oder Analysemustern (Patterns, z.B. *Design Patterns* [GHJV95]), was letztlich ja auch die Niederschrift von von *Best Practice* ist.

#### Zusammenfassung

Dass sich im Bereich IT-Dienstmanagement viele Standards parallel entwickeln, die trotz anscheinend verwandter Ziele und Methoden, von unterschiedlichen Darstellungsmitteln und Begrifflichkeiten Gebrauch machen, erschwert Vergleiche und Einordnungen. Für Organisationen, die Dienstmanagement anwenden wollen, ist es so praktisch unmöglich Gewissheit darüber zu erlangen, welches Rahmenwerk bzw. welche Kombination von Rahmenwerken für einen Einsatz in ihrer Situation am geeignetsten ist – zumindest ohne eine aufwändige Analyse zahlreicher, teilweise sehr umfangreicher Dokumente. Aber nicht nur für ITSM-Anwender, auch für Entwickler von ITSM-Werkzeugen ist diese Situation unerfreulich – für ein Werkzeug das für ein Framework entwickelt wurde, ist die Anwendbarkeit und der Nutzen im Kontext des Einsatzes eines anderen Dienstmanagement-Standards nur schwer nachvollziehbar.

#### *ITSM Framework Quagmire*

Die Vielzahl von Standards mit größtenteils unübersichtlichen Beziehungen zueinander erinnert an eine ähnliche Situation im Bereich des *Software Engineering*, die dort bisweilen als *Framework Quagmire* [She97, She01] bezeichnet wird. Dieses „Rahmenwerk-Dickicht“<sup>38</sup> ist in Abbildung 3.20 zusammengefasst. Dort sind die Rahmenwerke allerdings bereits nach ihrem ITSM-Bezug angeordnet – Standards zur Prozessbeschreibung sind oben platziert, in der Mitte die eigentlichen prozessorientierten ITSM-Standards, ganz unten CMM und CMMI, welche prozessorientierte IT-Entwicklungsstandards darstellen. Die Abbildung gibt einen Überblick über die im Laufe der letzten Jahre erschienenen Standards mit ihren Vorgängern und stellt dar, wo wesentliche Konzepte aus anderen Rahmenwerken wiederverwendet wurden sowie für welche Paarungen von Standards Vergleiche oder Abbildungsversuche existieren – das letztere schließt dabei das erstere ein: wo Konzepte aus einem anderen Standard wiederverwendet wurden, wird auch praktisch immer auf diesen referenziert und es existiert eine zumindest fragmentarische Abbildung.

#### 3.11.2 Einordnungsschema für Dienstmanagement-Rahmenwerke

Ein anerkannter Rahmen zur Einordnung von ITSM-Standards beziehungsweise IT-Management-Standards, also ein Meta-Framework, existiert bislang nicht.

<sup>38</sup>Übersetzung von „quagmire“: Sumpf, Morast; Schlamassel (fig.)



so wer letztlich diese Werkzeuge in welchem organisatorischen Kontext bedient, bleiben dabei, abgesehen von technischen Mechanismen zur Rechteverwaltung, meistens außen vor. Als die Vision hinter diesen Rahmenwerken kann man die Schaffung von Tools sehen, die so mächtig sind und so viele Aufgaben automatisieren, dass sich damit riesige, hochkomplexe Infrastrukturen durch eine einzelne Person administrieren lassen. Beim Einsatz eines solchen hypothetischen Werkzeugs gäbe es dann natürlich auch keine Notwendigkeit mehr für die Abstimmung von organisatorischen Abläufen – die gesamte Steuerung könnte ja in die Hand eines einzelnen Administrators gelegt werden. Aus dieser Vision heraus ist auch verständlich, dass in diesem Umfeld organisatorischen Aspekten und prozessorientiertem Management – aus diesem Blickwinkel betrachtet ja nur bald überflüssig werdende Krücken für die bislang bestehende Lücken in der Automatisierung – meist keine große Beachtung geschenkt wird.

#### *technisch orientierte Standards*

Zu den technisch orientierten Standards sind natürlich in erster Linie die klassischen Netzmanagement-Standards das *OSI Management Framework* und das SNMP-basierte *Internet-Management* zu sehen (vgl. [HAN99]), wenn auch das Konzept der *Systems Management Functional Areas* aus dem *OSI Management* anscheinend bisweilen auch zur organisatorischen Gestaltung des IT-Managements herangezogen wird (vgl. Umfrageergebnisse in [Sch04]). Unter den Prozessrahmenwerken steht die eTOM den technisch orientierten Standards noch am nächsten, liegt der Fokus der in ihr bislang feiner ausgearbeiteten Bereiche doch auf automatisierbaren Prozesselementen, die überwiegend der Infrastruktursteuerung dienen. Auch die UML ist ein eher technisch orientierter Standard: Zwar lassen sich mit ihr auch von der technischen Realisierung abstrahierte Sachverhalte ausdrücken oder Prozesse modellieren, ihrem Ursprung entsprechend findet sie aber immer noch vorwiegend dort Anwendung, wo Modelle zur Spezifikation von IT-Systemen erstellt werden sollen.

#### *prozessorientierte Standards*

Organisatorisch ausgerichtete oder prozessorientierte Rahmenwerke legen den Schwerpunkt hingegen typischerweise auf die Steuerung von Abläufen zwischen Personen, auch wenn hierbei neben Personal auch grundlegende IT-Systeme eine Prozessressource darstellen kann. Die *Managed Objectss* sind also in erster Linie die Prozesse – von den technischen Spezifika der Infrastruktur und der Infrastrukturmanagement-Tools wird weitgehend abstrahiert. Mehr organisatorisch als technisch orientiert sind, mit Ausnahme der eTOM, deren Schwerpunkt nicht eindeutig in eine Richtung tendiert, alle in Abschnitt 3.1 bis Abschnitt 3.9 vorgestellten ITSM-Rahmenwerke.

#### *Orientierung an Lebenszyklusphase*

Ein prozessorientiertes IT-Dienstmanagement umfasst mehrere Phasen, die sich in der Natur der in ihnen zu bewältigenden Aufgaben und in Konsequenz auch in den angewandten Mitteln und Methoden signifikant voneinander unterscheiden – entsprechend scheint eine Unterscheidung der Frameworks danach, auf welche Phase sie den Schwerpunkt legen, sinnvoll. Die grundlegenden Phasen Planung, Ausführung (bzw. Betrieb) und Analyse (vgl. Abschnitt 2.4) existieren

aber nicht nur im Prozessmanagement, sie lassen sich auch auf das Management von Infrastruktur-Lösungen übertragen. Die Einordnung im Lebenszyklus erfolgt am vereinfachten Lebenszyklusmodell aus Abbildung 2.4 (auf Seite 22). Ebenso wie bei der Ordnung nach der technischen Orientierung können Standards hier nur nach ihrem Schwerpunkt eingeordnet werden. Tatsächlich decken viele Rahmenwerke Aspekte aus mehreren Lebenszyklusbereichen ab – aber eben nicht im gleichen Maß und der gleichen Detaillierung.

Relativ eindeutig ist die Zuordnung bei Modellierungsstandards, wie der ARIS-Methodik oder UML. Zwar stellt eine Istmodellierung (vgl. Abschnitt 2.5) ein sinnvolles Werkzeug für die Analyse dar – die vorrangige Zielsetzung bleibt jedoch in aller Regel die Planung neuer oder angepasster Lösungen (ob organisatorischer oder technischer Natur) mittels einer Sollmodellierung. ITIL enthält zwar auch Ratschläge zur Prozessplanung und -einführung (letzteres wird nicht im hier verwendeten, vereinfachten Lebenszyklusmodell betrachtet), allerdings bleiben diese recht allgemein und oberflächlich.

*planungs-  
orientierte  
Standards*

Ob ein Rahmenwerk mehr Wert auf die Ausführung – das heißt in diesem Zusammenhang, welche Gestalt die Prozesse haben sollen – oder auf die Analyse Wert legt, ist bisweilen weniger eindeutig. Viele der weiter oben betrachteten Standards enthalten zahlreiche Richtlinien zu beiden Aspekten oder zumindest Richtlinien, die man als beide Aspekte betreffend interpretieren kann. Beispielsweise bietet die ITIL für einige Prozesse Vorschläge für KPIs, welche die Datenbasis für die Prozessanalyse legen können. Diese Aspekte nehmen aber vergleichsweise geringen Raum ein, und werden auch nicht für alle Prozesse adressiert – man kann also sagen, dass die sehr viel ausführlicheren Richtlinien zur Prozessgestaltung also in der ITIL den Schwerpunkt bilden. Für eTOM und MOF beispielsweise ist dies eindeutig der Fall: Beide enthalten viele Vorgaben zur Prozessgestaltung, aber praktisch keine Inhalte welche die Prozessanalyse betreffen.

*ausführungs-  
bzw. gestalt-  
ungsorientierte  
Standards*

In ITS-CMM, CobiT und eSCM bilden im Vergleich Richtlinien und Hinweise zur Prozessanalyse – KPI-Definitionen, Kennzahlensysteme, Merkmale von Prozessreife usw. – den Schwerpunkt. In all diesen Rahmenwerken werden aber auch Prozesse oder Praktiken vorgegeben. Allerdings steht dabei meist eine Kontrollfunktion im Mittelpunkt. Diese Prozesse stellen überwiegend, stärker als dies beispielsweise bei Prozessen der ITIL oder MOF der Fall ist, Kontroll- oder Analyseprozesse dar. ISO/IEC 20000 liegt in dieser Hinsicht zwischen den analyseorientierten Rahmenwerken und ITIL: Es übernimmt von letzterem die grundlegenden Gestaltungsrichtlinien in stark verkürzter Form, legt aber in seinen Anforderungen vergleichsweise viel Wert auf Qualitätsmanagementaspekte wie eindeutig zugeordnete Verantwortlichkeiten und Erfassung wichtiger Kennzahlen.

*analyse-  
orientierte  
Standards*

Eine zusammenfassende Abschätzung, inwieweit diese Aspekte berücksichtigt wurden beziehungsweise wo der jeweilige Fokus des Frameworks zu sehen ist, ist für die Abschnitt 3.1 bis Abschnitt 3.9 vorgestellt ITSM-Rahmenwerke und einige

*tabellarische Zu-  
sammenfassung*

	ITIL	eTOM / NGOSS	ITS-CMM	CobiT	eSCM	MOF	ITUP	ISO/IEC 20000	UML	ARIS	Interner Mgmt.	OSI Mgmt.
<b>Lebenszyklus</b>												
Planung	★	★	☆	☆	☆	☆	☆	☆	★★★	★★★	☆	☆
Ausführung	★★★	★★★	★	★★	★	★★★	★★★	★★	-	-	★★	★★★
Analyse	★★	-	★★★	★★★	★★★	-	☆	★★★	☆	☆	★	★
<b>Prozesse</b>	★★★	★★	★★★	★★★	★★★	★★★	★★★	★★★	★	★★★	-	☆
<b>Technische Infrastruktur</b>	☆	★★	-	-	-	☆	★	-	★★★	☆	★★★	★★★

Legende

- Aspekt nicht adressiert
- ☆ Aspekt nur indirekt adressiert
- ★ Nur als Nebenaspekt adressiert
- ★★ Aspekt explizit adressiert, aber geringe Detaillierung
- ★★★ Aspekt schwerpunktmäßig adressiert

Abbildung 3.21: Grundlegende Ausrichtung gängiger ITSM-Rahmenwerke

in Abschnitt 3.10 angesprochenen Standards (UML, ARIS, Internet-Management und *OSI Management*) in Abbildung 3.21 dargestellt.

Konzentriert man sich auf die jeweiligen Schwerpunkte der Rahmenwerke (unter Vernachlässigung ihrer Breite bei der Adressierung der verschiedenen Aspekte) so lassen sie sich im Überblick wie in Abbildung 3.22 dargestellt einordnen. Wie dort zu erkennen, sind MOF und ISO/IEC 20000 der ITIL also nicht nur inhaltlich, sondern auch im Schwerpunkt ihrer allgemeinen Ausrichtung am nächsten. Alle anderen Rahmenwerke legen ihren Fokus woanders, was bedeutet, dass kaum damit zu rechnen ist, dass diese in absehbarer Zukunft eine konkrete Alternative zu ITIL in deren primären Anwendungsfällen darstellen werden. Auch bei einer weiteren Angleichung von Begriffen und sich überlappenden Teilkonzepten zwischen ITIL und anderen Standards (wie sie sich bei CobiT anbahnt) steht daher nicht zu erwarten, dass sich Werkzeuge (oder Konzepte zur Werkzeugunterstützung) zwischen diesen übertragen lassen. Hierfür unterscheiden sich, trotz oberflächlicher Ähnlichkeit, die Zielsetzungen der Rahmenwerke im Detail zu stark – sie wenden sich an separate Adressatenkreise mit jeweils ihrem eigenen Blickwinkel auf die Prozesse, was letztlich auch in unterschiedliche fachliche Anforderungen an ein Werkzeug münden muss.

*Schwerpunkte liegen unterschiedlich*

### 3.11.3 Zusammenfassung und Ausblick

Neben der ITIL existieren einige weitere prozessorientierte Rahmenwerke für Dienstmanagement. Unglücklicherweise besteht zwischen diesen kein Konsens über die zentralen Begriffe, Aufgabenbereiche oder Konzepte des ITSM, weswegen sie schwer miteinander zu vergleichen oder voneinander abzugrenzen sind. Entsprechend existieren bislang keine kritische Vergleiche in denen mehr als zwei Frameworks betrachtet werden – und selbst dies geschieht in fast allen Fällen

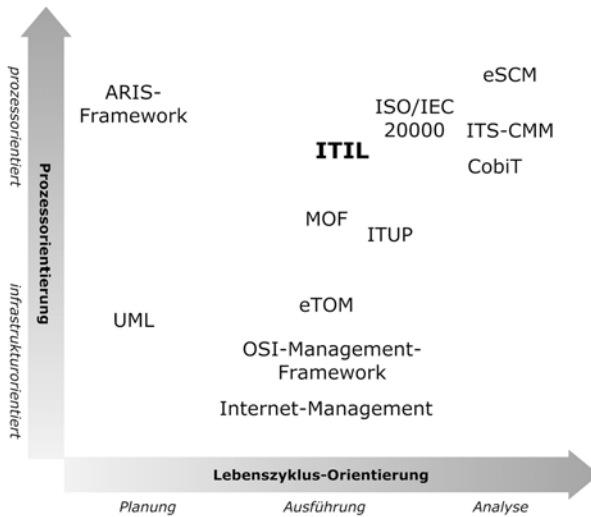


Abbildung 3.22: Einordnung der Schwerpunkte ITSM-relevanter Rahmenwerke

nicht von unabhängiger Seite, sondern vom Eigentümer eines der beiden untersuchten Rahmenwerke. Es ist daher auch schwierig klare Empfehlungen für oder gegen den Einsatz der einzelnen Standards auszusprechen. Auch ist ihr Nutzen, obwohl intuitiv einsichtig, nur vereinzelt und für bestimmte Szenarien überzeugend belegt (z.B. für eine kleine Teilmenge der MOF-Prozesse in [ER06]). Die große Verbreitung und, auch relativ zu den anderen Standards, weiter zunehmende Bedeutung von ITIL (vgl. [Blu04] und [Blu06]) ist daher wohl nicht auf objektive Qualitätskriterien zurückzuführen sondern durch andere Faktoren bedingt.

*Status-Quo bei ITSM-Frameworks bleibt unübersichtlich*

Es bleibt aber festzuhalten, dass, wenn man einer Alternative zur ITIL mit der gleichen schwerpunktmäßigen Zielrichtung sucht, nur die ITIL-basierten MOF, ITUP und ISO/IEC 20000 in Frage kommen. Die beiden ersten sind allerdings proprietäre Standards deren zukünftige Entwicklung allein in der Hand der herausgebenden Firma liegt. ISO/IEC 20000 ist aufgrund seiner extremen Kürze ohne die ITIL als „Sekundärliteratur“ wohl kaum sinnvoll zu interpretieren. Entsprechend ist auch nicht lohnend in Bezug auf eine Werkzeugunterstützung auf Entwicklungen im Kontext anderer Rahmenwerke zu setzen. Auch wenn es für eTOM ein vielversprechendes Projekt (NGOSS, vgl. Abschnitt 3.2.1) gibt – die Unterschiede in Zielsetzung und Ansatz sind zu groß für eine in absehbarer Zeit zu realisierende Integration der beiden Standards, welche die Voraussetzung für eine gemeinsame Werkzeuglandschaft wäre.

*Framework-übergreifende Tool-Konzepte nur schwer möglich*

### *3 ITIL im Kontext der ITSM-Rahmenwerke*

Im folgenden Kapitel werden daher nur die Möglichkeiten zur Tool-Unterstützung für die einzelnen Dienstmanagementprozesse der ITIL genauer untersucht.



## 4 Möglichkeiten und Grenzen der Werkzeug-Unterstützung

Wie im vorhergehenden Kapitel gesehen, sind liegen die Schwerpunkte der verschiedenen Rahmenwerke im IT-Management in einigen Aspekten recht weit auseinander – für einen bestimmten Ansatz entwickelte IT-Werkzeuge lassen sich daher nicht ohne weiteres für andere ITSM-Ansätze verwenden. Da, wie im folgenden Abschnitt 4.1 dargelegt, ein ITIL-basiertes Dienstmanagement ohne Unterstützung durch geeignete IT-Tools nicht effizient umsetzbar erscheint, ist die Entwicklung eines entsprechenden allgemeinen Konzeptes zum Werkzeug-Einsatz notwendig.

In Abschnitt 4.2 erfolgt zunächst eine Werkzeug-Klassifikation. Ein tragfähiges Tool-Konzept ist kaum möglich, wenn keine Klarheit über die grundlegenden Arten einsetzbarer Werkzeuge besteht. Bestehende Klassifikationen von IT-Management-Werkzeugen sind aber entweder historisch gewachsen und somit wenig strukturiert oder sie berücksichtigen die Unterstützung von Dienstmanagementprozessen nicht ausreichend (siehe Abschnitt 4.2.1). In Abschnitt 4.2.2 wird daher ein Einordnungsschema entwickelt, das eine Klassifikation der Werkzeuge anhand der gleichen grundlegenden Aspekte erlaubt, die bereits für die Einordnung der Rahmenwerke des Dienstmanagements (vgl. Abschnitt 3.11.2) herangezogen wurden. Dies erlaubt die Benennung der allgemeinen Werkzeug-Kategorie, die für eine möglichst umfassende Unterstützung der in der ITIL beschriebenen Dienstmanagementprozesse am geeignetsten ist („Prozess-Betriebswerkzeuge“).

Auch diese Gruppe ist aber noch relativ weit gefasst, weswegen es für eine genaue Identifizierung der geforderten Werkzeug-Eigenschaften Sinn macht, das Einordnungsschema noch weiter zu verfeinern (Abschnitt 4.2.3). Dies ist auch deswegen geboten, weil, wie in Abschnitt 4.3 gezeigt wird, die Dienstmanagementprozesse der ITIL keineswegs eine homogene Menge bilden. Um die Abweichungen in den grundlegenden Prozesseigenschaften klar herauszuarbeiten, reichen aber wieder bestehende Klassifikationsschemata (vgl. Abschnitt 4.3.1) nicht aus. Untersucht man jedoch die ITIL-Prozesse anhand Dienstmanagement-spezifischer und Werkzeugunterstützung-relevanter Kriterien, wie sie in Abschnitt 4.3.2 erarbeitet werden, so lässt sich (unter Annahme der für größere IT-Organisationen typischen Rahmenbedingungen, vgl. Abschnitt 4.3.3) aufzeigen, dass sie sich in ihrer Natur zum Teil grundlegend unterscheiden.

Unter Rückgriff auf die Ergebnisse dieser Untersuchung (Abschnitt 4.3.5) lässt

sich ein Klassifikationsschema für Dienstmanagementprozesse entwickeln (Abschnitt 4.3.6), welches eine Zuordnung von ITIL-Prozessen zu den in Abschnitt 4.2.3 gefundenen Werkzeugkategorien erlaubt. Diese Klassifikation von Dienstmanagementprozessen bildet die Grundlage, um im folgenden Kapitel 5 einen zielgerichteten Modellierungsansatz entwickeln zu können.

### 4.1 Motivation für spezialisierte ITIL-Werkzeuge

#### *ITIL-Prozesse brauchen Tools*

Im Allgemeinen gilt, auch wenn bei Aussagen hierzu in der Regel auf Produktions- und Verwaltungsprozesse Bezug genommen wird, dass ein modernes, effizientes Prozessmanagement erst durch entsprechende IT-Unterstützung möglich wird (siehe z.B. [HC93]). Die Einführung von ITIL in eine IT-Organisation, die bislang dem Management der Prozesse keine oder nur wenig Beachtung geschenkt hat, wird also in der Regel mit der Einführung neuer IT-Werkzeuge einhergehen müssen – oder zumindest mit einer umfassenden Anpassung bestehender Werkzeug-Infrastrukturen. Im Kontext von ITIL-Projekten wird von Consultants zwar häufig vor einer zu großen Fixierung auf den Tool-Aspekt gewarnt: Der Spruch „a fool with a tool is still a fool“ findet sich auf Folien zahlreicher Vorträge zu ITIL. Aber für eine Erfolg versprechende Durchführung eines organisatorischen Wandel mit sich bringenden Projektes, wie es eine ITIL-Einführung zweifelsohne darstellt, müssen auch Fragen der Werkzeugunterstützung ausreichend und rechtzeitig adressiert werden [Mar04].

#### *Ansätze für andere Frameworks nicht geeignet*

An die Frage der Werkzeuge kann prinzipiell auf zwei Arten herangegangen werden. Durch den Einsatz spezialisierter Tools oder die Verwendung bestehender, ursprünglich für einen anderen Einsatzzweck entwickelter Werkzeuge. Wie in Kapitel 3 gesehen, unterscheiden sich die Ansätze und Methoden der verschiedenen Frameworks für das IT-Management. Entsprechend steht nicht zu erwarten, dass Werkzeugansätze optimal greifen, welche für nicht prozessorientierte Standards oder für auf andere Lebenszyklusphasen ausgerichtete Rahmenwerke entwickelt wurden. So wird auf absehbare Zeit auch das die eTOM-Prozesse integrierende NGOSS-Projekt keine Tools hervorbringen können, welche sich ohne weiteres für ITIL-basiertes Management einsetzen lassen.

#### *keine konkreten Tool-Richtlinien von der OGC*

ITIL-Unterstützung wird von vielen Produkten versprochen. Wie bereits in Abschnitt 3.1.4 dargelegt, gibt es allerdings in der aktuellen Version der ITIL, außer vereinzelt vagen Aussagen, keine konkrete Richtlinien für ITIL-Werkzeuge. Das OGC selbst bietet auch weder Tests oder Kriterien, mit denen sich die Eignung oder „ITIL-Kompatibilität“ von Tools bestimmen ließe (siehe auch ITIL-FAQ der OGC [OGC07]). Auch eine allgemeine Benennung von Ansätzen für mögliche unterstützende Werkzeugen existiert nur ansatzweise – jenseits der Nennung von Datenbanken (insbesondere der CMDB, vgl. Abschnitt 3.1.3) finden Werkzeuge in der ITIL nur vereinzelt, und in wenigen, mehr oder weniger unstrukturierten

Stichpunkten, Erwähnung.

Im kommerziellen ITIL-Umfeld gibt es durchaus Einteilungen, die sich auf ITIL-unterstützende Tools beziehen. Eine solche kann beispielsweise auf der Webseite *Toolselector* [Inf06] gefunden werden. Werkzeuge werden hier aber, neben allgemeinen nicht-funktionalen Kriterien wie Preis oder unterstützte Betriebssysteme, nur danach unterschieden, für welche ITIL-Prozesse Unterstützung versprochen wird.

*Toolselector*

Ähnlich teilt auch das *PinkVerify*-Schema von *Pink Elephant* [Pin07] (vgl. Abschnitt 3.1.4) Werkzeuge nach den von ihnen unterstützten Prozessen ein. Insgesamt ist die Unterstützung von acht Prozessen überprüfbar, *IT Service Continuity Management* und *Financial Management for IT Services* werden, ohne Begründung, nicht erfasst. Für einen zertifizierbaren Prozess werden zwischen 14 und 30 (im Schnitt 20) Kriterien angegeben, welche als Ja/Nein-Fragen formuliert sind. Für die Zertifizierung eines Werkzeuges muss es für den jeweiligen Prozess 100% der obligatorischen Kriterien sowie 80% der Kriterien insgesamt erfüllen [Pin06]. Man kann allerdings einfach nachrechnen, dass die zweite Bedingung – insofern die obligatorischen Kriterien erfüllt sind – für fünf der acht Prozesse automatisch erfüllt wird und für die restlichen drei nur Erfüllung eines einzigen zusätzlichen Kriteriums erfordert.

*PinkVerify*

In beiden Fällen werden also ITIL-Tools letztlich danach unterschieden, ob sie einen bestimmten ITIL-Prozess unterstützen oder nicht. Während *Toolselector* sich dabei auf Herstellerinformationen verlässt, bietet *Pink Verify* immerhin für die meisten Prozesse einige Kriterien an, mit denen sich die Aussage „unterstützt ITIL-Prozess X“ erhärten lassen soll. Diese Kriterien bleiben allerdings sehr allgemein und lassen sich von einem großen Spektrum von Software-Produkten mit unterschiedlichsten Eigenschaften prinzipiell erfüllen. Zudem bleibt völlig offen, wie die Kriterien zustande kommen.

*kaum  
Abgrenzung  
verschiedener  
ITIL-Tools*

Ähnliche wenig belastbar sind auch die Aussagen der Hersteller über die ITIL-Unterstützung der auf dem Markt befindlichen Software-Werkzeuge. Auch wenn zahlreiche Hersteller ihre Produkte als „ITIL-konform“ oder als „ITIL-Tool“ verkaufen – wie erwähnt besteht auch im Markt zur Zeit kein Konsens darüber, was diese Begriffe aussagen, welche Typen von ITIL-Werkzeugen es gibt (oder geben kann) und welche Funktionalitäten sie besitzen sollten. Bei Netzmanagement-Werkzeugen beispielsweise kann ein Tool-Hersteller darauf verweisen, welche Spezifikationen (für Protokolle, Datenmodelle usw.) von seinem Produkt in welcher Version und in welchem Umfang erfüllt werden. Für ITIL-Werkzeuge ist derlei nicht möglich, weil keine entsprechenden allgemein anerkannten oder auch nur bekannten Spezifikationen existieren. Die Prozessbeschreibungen selbst sind zu informell und vage als dass sich konkrete Anforderungen oder gar eine Tool-Spezifikation aus ihnen trivial ableiten ließe. In Konsequenz bleibt die Aussage, ein Produkt sei ITIL-konform, weder beleg- noch widerlegbar. Auch aussagekräftige Vergleiche zwischen verschiedenen Werkzeugen können mangels geeigneter

*mangels  
Spezifikationen  
zahlreiche  
Fragen offen*

Einordnungs- und Bewertungsmaßstäbe kaum angestellt werden.

### 4.2 Klassifikation von Werkzeugen

Ein erster Schritt hin zu einer systematischen Tool-Unterstützung der ITIL-Prozesse ist die Klärung, welche grundlegenden Kategorien von IT-Management-Werkzeugen für ITIL-Prozesse zum Einsatz kommen können. Nach einem kurzem Überblick im folgenden Abschnitt 4.2.1 über bestehende Werkzeugklassifikationen im IT-Management allgemein, folgt in Abschnitt 4.2.2 die Vorstellung einer grundlegenden, ähnlich dem Schema zur Einordnung von ITSM-Frameworks (vgl. Abschnitt 3.11.2) aufgebauten Klassifikationen von Werkzeugen. Für die innerhalb dieser Klassifikation hinsichtlich ITIL-Werkzeugen vorrangig interessanten Kategorie der (ITSM-) Prozess-Betriebswerkzeuge wird schließlich eine verfeinernde Taxonomie bezüglich funktionaler Grundaspekte (Unterstützung von Kommunikation und Unterstützung von Ablaufsteuerung) in Abschnitt 4.2.3 abgeleitet.

#### 4.2.1 Bestehende Klassifikationen

*informelle  
Werkzeug-  
Klassifikationen*

Auch im IT-Management allgemein besteht keine allseitig anerkannte, genaue Typisierung von Werkzeugen. Weitgehende Einigkeit über Aufgaben und Funktionalitäten besteht nur für einige verbreitete Arten von Werkzeugen, für die sich im Laufe der Zeit eine gemeinsame Vorstellung entwickelt hat, wie beispielsweise für Protokollanalytoren. In den Produktportfolios der einzelnen Hersteller finden sich so Typenbezeichnungen für Management-Tools, die sich zumindest partiell im Laufe der Zeit angeglichen haben. Eine solche informelle, aber im großen und ganzen anerkannte Kategorisierung ist auch die Basis für die Unterteilung verschiedener Kategorien von IT-Management-Werkzeugen, wie sie in Fachzeitschriften zur Anwendung kommt – beispielsweise in Produkttests oder der Leserwahl für die beliebtesten Produkte.

*in Fachzeit-  
schriften  
verwendete  
Kategorien*

So werden bei der jährlichen Wahl der besten Werkzeuge in der Zeitschrift *Redmond* (die sich an Administratoren Windows-basierter Infrastrukturen wendet) diese in sechs Hauptkategorien eingeteilt: „Network and Systems Management“, „Administration“, „Installation and Deployment“, „Security“, „Storage and Backup“ und „Training and Certification“. Unter „Network and Systems Management“ existieren beispielsweise 14 Unterkategorien für Tools, die von „Performance Management Product“ über „Virtual Server Product“ bis zu „Browser“ reichen [Red06]. Eine Systematik hinter diesen Einteilungen ist aber weder dargelegt noch offensichtlich – die Einteilung wird wohl eher durch das Angebot am Markt, aktuelle Interessen der Leserschaft und die am stärksten beworbenen

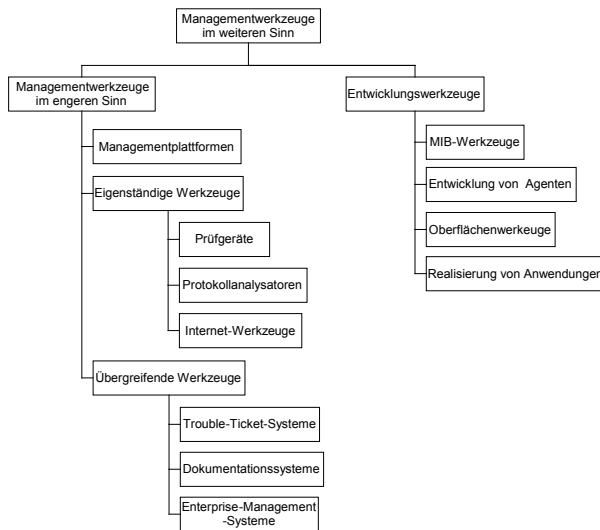


Abbildung 4.1: Klassifikation von Management-Werkzeugen nach [HAN99]

und diskutierten Begriffe bestimmt. Eine Bezugnahme auf ITIL-Werkzeuge wird bislang nicht vorgenommen.

Typisierungen von Werkzeugen für das IT-Management in der wissenschaftlichen Literatur adressieren meist eine vergleichsweise enge Subdomäne des IT-Managements. Eine der wesentlichen Ausnahmen bildet die Klassifikation in [HAN99], die in Abbildung 4.1 dargestellt ist. Eine Einordnung von ITIL-geeigneten Werkzeugen in diese Klassifikation ist allerdings nicht unmittelbar möglich. Zwar könnten die Unterkategorien von „Übergreifende Werkzeuge“ teilweise als Typen von Werkzeugen für für prozessorientiertes ITSM aufgefasst (Trouble-Ticket-Systeme). Prozessorientierte Managementmittel werden aber in [HAN99] als getrennt von Management-Werkzeugen gesehen und in einer separaten Klassifikation in Mittel zur „Informationshilfe“, Mittel zur „Kommunikationshilfe“, Mittel zur „Verarbeitungshilfe“ und Mittel zur „Ausführungshilfe“ eingeteilt – diese sind jedoch eher als Komponenten einer integrierten Lösung denn als eigenständige Werkzeugtypen zu sehen.

*Werkzeug-Klassifikation in der Literatur*

Diese bestehenden Schemas zur Klassifikation machen in ihrem jeweiligen Kontext Sinn. Sie erlauben aber nicht, Werkzeuge intuitiv danach einzuteilen, ob sie die Umsetzung eines der in Kapitel 3 vorgestellten Rahmenwerke prinzipiell unterstützen können. Für dieses Ziel wird im Folgenden ein geeignetes Klassifikationsschema entwickelt.

## 4.2.2 Einordnung anhand Lebenszyklus- und technischer Orientierung

In den IT-Management-Frameworks unterscheiden sich mit der schwerpunktmäßigen Orientierung auch die eingesetzten (nicht-technischen) Mittel und Methoden. Es liegt nahe, dass sich die funktionalen Anforderungen an ein unterstützendes Werkzeug in einem gewissen Maß analog hierzu verändern. Das folgende Klassifikationsschema nimmt daher die beiden in Abschnitt 3.11.2 zur Einordnung von IT-Management-Standards eingeführten Unterscheidungskriterien auf und verwendet sie auch für die Werkzeugklassifikation.

### *Werkzeuge zur Unterstützung der Planung*

So kann man Modellierungswerkzeuge, analog zur Einordnung von Modellierungsstandards als planungsorientierte Rahmenwerke in Abschnitt 3.11.2, als Planungswerkzeuge interpretieren. Für die Planung von Prozessen werden beispielsweise häufig Werkzeuge für graphische Modellierung verwendet, welche die Erstellung von Prozessmodellen (vgl. Abschnitt ??) in ARIS-Modelltypen oder anderer Notationen erleichtern. Es existieren auch Planungswerkzeuge für infrastrukturorientiertes Management (z.B. Software für Netzplanung), wenn sich hier auch keine standardisierten Modellierungsmethoden in großer Breite etabliert haben. UML-Tools können, sofern sie die entsprechenden Diagrammtypen unterstützen, für Infrastruktur- (z.B. mittels *Deployment Diagrams*) wie auch Prozessplanungen (z.B. mittels *Activity Diagrams*) eingesetzt werden. Neben dem Erleichtern der Erstellung von Plänen ermöglichen einige Tools auch die Simulationen oder das automatisierte Prüfen erwünschter Eigenschaften dieser auf Basis der erstellten Modelle.

### *Werkzeuge zur Unterstützung der Ausführung*

Für die Ausführungsphase lassen sich prinzipiell viele Sub-Typen von Betriebswerkzeugen bilden. Fast alle der vielfältigen Managementaufgaben im IT-Betrieb sind durch spezialisierte, eigenständige Werkzeuge unterstützbar (vgl. Werkzeugklassifikation nach [HAN99], Abbildung 4.1). Interessanter im ITSM-Zusammenhang sind allerdings integrierende oder übergreifende Werkzeuge. Ein Beispiel für ein entsprechendes prozessorientiertes Tool ist ein Trouble-Ticket-System, welches die Kommunikation und Koordination zwischen verschiedenen Parteien in einem Entstörungsprozess erleichtert. Mehr am technischen Management orientierte Werkzeuge können Aspekte der Steuerung von Infrastruktur-Komponenten (z.B. Load-Balancing, Failover) automatisieren, die sonst manuell zu erledigenden wären. Hinsichtlich des Grades der technischen Orientierung zwischen diesen beiden Werkzeugtypen liegen Management-Plattformen, die verschiedene eigenständige Tools des Infrastrukturmanagements für die Steuerung durch einen (einzelnen) Administrator unter einer gemeinsamen Benutzeroberfläche integrieren.

### *Werkzeuge zur Unterstützung der Analyse*

Die in der Analysephase eingesetzten Datenanalyse- und Reporting-Werkzeuge dienen vor allem der Auswertung der während der Ausführung gesammelten Daten. Im prozessorientierten Management erleichtert dies die Konsolidierung der prozessbezogenen Rohdaten (z.B. Durchlaufzeit einzelner Prozessinstanzen) zu aussagekräftigen KPIs (z.B. durchschnittliche Durchlaufzeit eines Prozesses)

und die Erkennung von Trends (z.B. Entwicklung der durchschnittliche Durchlaufzeit). Auch eine graphische Aufbereitung der Ergebnisse und die Generierung von Reports gehören hier zu den Aufgaben, die durch IT-Tools erheblich erleichtert werden können. Werkzeuge die entsprechende Aufgaben erfüllen, werden im infrastrukturorientierten Management häufig als Performance-Management-Tools bezeichnet.

Abbildung 4.2 zeigt zusammenfassend die sechs grundlegenden Werkzeugarten, die sich aus dem Kontext dieser Arbeit relevanten Ausprägungen der beiden Unterscheidungskriterien ergeben. Die Kategorien sind aber, wie schon auch bei der Einordnung der Frameworks in Abschnitt 3.11.2, nicht scharf voneinander abgegrenzt. Es gibt Arten von Werkzeugen, die sich mehr als einem Bereich zuordnen lassen. Protokollanalytoren können beispielsweise als eigenständige Werkzeuge nicht nur für die Entstörung (die der Ausführungsphase zuzuordnen ist), sondern auch für Analyse- und Optimierungszwecke genutzt werden. Dennoch lässt sich mit den analog aufgebauten Einordnungsschemata für Frameworks und Werkzeuge (vgl. Abbildung 4.2 mit Abbildung 3.22) eine grobe Zuordnung von Werkzeugen zu den durch diese prinzipiell unterstützbaren ITSM-Ansätzen finden. Als ein ITIL-Werkzeug ist dementsprechend in erster Linie ein Prozess-Betriebswerkzeug für ITSM zu sehen – also ein Tool, welches die Ausführung beziehungsweise den täglichen Betrieb des prozessorientierten Dienstmanagements unterstützt.

*Abgrenzung  
nicht scharf*

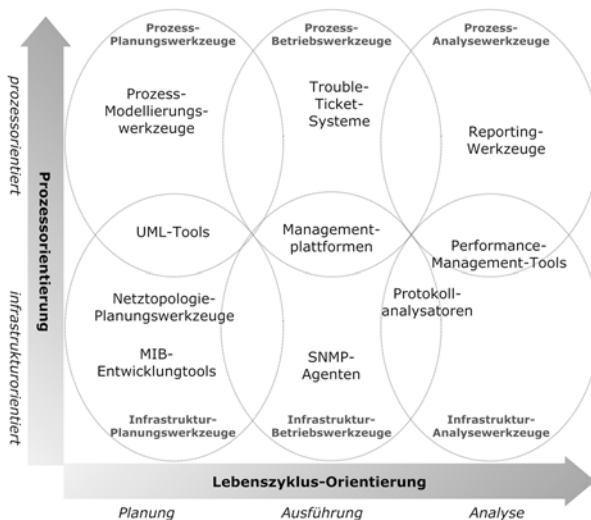


Abbildung 4.2: Einordnung der Schwerpunkte gängiger Management-Werkzeuge

### 4.2.3 Verfeinerung der Einordnung für prozessorientierte ITSM-Betriebswerkzeuge

„process aware“  
Werkzeuge für  
ITIL

Für die Unterstützung einzelner Aktivitäten eines ITSM-Prozesses können prinzipiell viele Arten von IT-Werkzeugen eingesetzt werden. So können neben klassischen Werkzeugen des Netz- und Systemmanagements (also z.B. SNMP-Tools, Protokollanalytoren usw.) selbst viele Standardapplikationen wie Textverarbeitung, E-Mail oder Tabellenkalkulationen nutzbringend zur Anwendung kommen. Ein Prozess-Betriebswerkzeug für einen ITIL-Prozess *X*, also ein „*X*-Management-System“<sup>1</sup> sollte nicht nur einzelne Aspekte oder Aktivitäten erleichtern, sondern eine möglichst umfassende Unterstützung des Prozesses als Ganzem liefern – also „process aware“ sein.

Kommunikation  
zwischen  
Menschen mehr  
im Mittelpunkt

Gegenüber den technischer orientierten Ansätzen zum IT-Dienstmanagement unterscheidet sich das prozessorientierte ITSM vor allem durch zwei Aspekte: Zum einen verlagert sich der Schwerpunkt von der Frage, wie man es (einzelnen) Administratoren besser ermöglichen kann die Infrastruktur zu managen, hin zu der Fragestellung, wie das IT-Personal insgesamt beim dienstorientierten Management der IT kooperieren kann. Es rückt also der im klassischen IT-Management oft vernachlässigte Aspekt der Kommunikation zwischen Personen mehr in den Mittelpunkt. Zum anderen kennzeichnet Prozesse, dass sie verschiedene Aktivitäten umfassen, deren Ablauf über Systemgrenzen hinweg gesteuert werden muss.

Kommunikation  
und  
Ablaufsteuerung  
müssen  
unterstützt  
werden

In Anlehnung an das *3K-Modell* [Teu01] für Werkzeuge des *Computer-Supported Cooperative Work* (CSCW) kann man also sagen, dass prozessorientierte ITSM-Tools die beiden Aspekte Koordination (oder auch Ablaufsteuerung) und Kommunikation/Kooperation<sup>2</sup> unterstützen müssen. Im Vergleich zu den Typen prozessorientierter Managementmittel bei [HAN99] (vgl. Abschnitt 4.2.1) orientieren sich die beiden hier vorgeschlagenen Kriterien am Typ „Kommunikation“ und dem unter „Ausführungshilfe“ eingeordneten Subtyp „Automatisierung“.

Kategorien  
prozess-  
orientierter  
ITSM-Betriebs-  
werkzeuge

In Abbildung 4.3 werden daher vier entsprechende Kategorien festgelegt, in die Werkzeuge je nach ihrer Unterstützung dieser beiden Aspekte fallen. Genau genommen sind es für prozessorientierte ITSM-Betriebswerkzeuge nur drei Kategorien, denn die erste Kategorie I umfasst diejenigen Werkzeuge, die weder eine um-

<sup>1</sup>Der Begriff „Management-System“ kann unterschiedliche Bedeutungen haben. Hier ist mit „Management-System“ nicht, wie sonst häufig im Kontext des Qualitätsmanagements, ein Verbund von Prozessen, Prozeduren, Richtlinien usw. gemeint. Vielmehr wird im Folgenden der Begriff „Management-System“ oder Zusammensetzungen hiermit (z.B. „Incident-Management-System“), für IT-Systeme verwendet – im gleichen Sinne wie, im Kontext des technischen IT-Managements, Begriffe wie „Network-Management-System“ oder „Fault-Management-System“ als IT-Systeme verstanden werden (siehe z.B. [Mis04].)

<sup>2</sup>Hiermit ist die Kommunikation und Kooperation zwischen menschlichen Akteuren gemeint. Zur Vereinfachung hier darauf verzichtet, wie es eigentlich beim 3K-Modell der Fall ist, zusätzlich zwischen Kommunikation und Kooperation unterstützenden Werkzeugen zu differenzieren.



fassende Kommunikationsunterstützung noch eine weiter gehende Ablaufsteuerung anbieten und daher nicht als prozessorientiert zu bezeichnen sind. Hierunter fallen die meisten klassischen IT-Management-Werkzeuge. Wie diese, sind auch die Werkzeuge der Kategorie II eher auf die Benutzung durch einzelne Personen ausgerichtet. Sie bieten allerdings eine Ablaufsteuerung, wenn diese Abläufe auch zwischen IT-Systemen stattfinden – im Unterschied zu übergreifenden Werkzeugen der Kategorie I realisieren sie also nicht nur eine informationsorientierte oder Portal-orientierte, sondern eine workflow- bzw. prozessorientierte Integration von IT-Systemen (vgl. die Typen von *Application Integration* in [Lin03]). Sowohl Werkzeuge der Kategorie III wie auch der Kategorie IV erleichtern die Kommunikation und Kooperation innerhalb von Gruppen. Erstere können als Workgroup-Computing-Werkzeuge bezeichnet werden (vgl. wieder 3K-Modell, [Teu01]). Werkzeuge der Kategorie IV bieten aber eine weitergehende Automatisierung der Koordination der verschiedenen Aktivitäten. So sind beispielsweise einfachere Trouble-Ticket-Systeme, welche die Ablage von Tickets in verschiedenen je einem Mitarbeiter zugeordneten „Queues“ und das Weiterleiten dieser im Sinne einer gemeinsamen Bearbeitung erlauben, bei denen diese Weiterleitung manuell gesteuert werden muss, in Kategorie III einzuordnen. Workflow-orientierte Trouble-Ticket-Systeme, die Tickets nach in einer Prozessdefinition festgelegten Kriterien (z.B. Kategorie, Priorität, momentane dokumentierte Arbeitslast bei den verschiedenen möglichen Adressaten) automatisch richtig zuweisen, die Bearbeitungs-Queues nach Ticket-Priorität sortieren oder Tickets je nach Priorität auch zeitgesteuert eskalieren, sind Tools der Kategorie IV.

Ein Tool der Kategorie IV kann auch als IT-Management-spezifisches *Workflow-Management-System* (WfMS) bezeichnet werden. Workflow-Management-Systeme ermöglichen eine weitgehend automatische Steuerung von Workflows mittels einer Software, die eine rechnerorientierte Repräsentation der Workflow-Logic interpretieren kann [Hol95]. Diese Systeme ermöglichen also, zumindest im Prinzip, eine umfassende Kontrolle über den Prozess, wie sie, insbesondere in Hinblick auf zeitkritische Aufgaben (z.B. wenn zur Einhaltung des *Service Level* die Lösung eines *Incident* innerhalb einer gewissen Maximalzeit notwendig ist) in ITSM-Prozessen wünschenswert ist. Außerdem können genaue Werte für viele der Prozesskennzahlen (z.B. Bearbeitungs- und Wartezeiten für die verschiedenen Schritte im Prozess *Incident Management*) am besten erhoben werden, wenn die Steuerung der Prozessschritte durch ein IT-System erfolgt, so dass Basisdaten für die Kennzahlen ohne großen zusätzlichen Aufwand mit erfasst werden können.

Allerdings fallen die Vorteile des Einsatzes von WfMS unterschiedlich stark aus, je nach Charakteristik des umzusetzenden Prozesse [zM03]. Außerdem sind – wenn man einen etwas weiteren Prozessbegriff annimmt – wie offensichtlich bei der ITIL der Fall (vgl. Abschnitt 4.3.6) – nicht alle Arten von Prozessen gleichermaßen für den WfMS-Einsatz geeignet [HEN01]. Um zu bestimmen, für welche ITIL-Prozesse die Entwicklung von Werkzeugen besonders lohnenswert wäre

*WfMS fallen in Kategorie IV*

*Nicht jeder Werkzeug-Typ passt zu jedem Prozess*

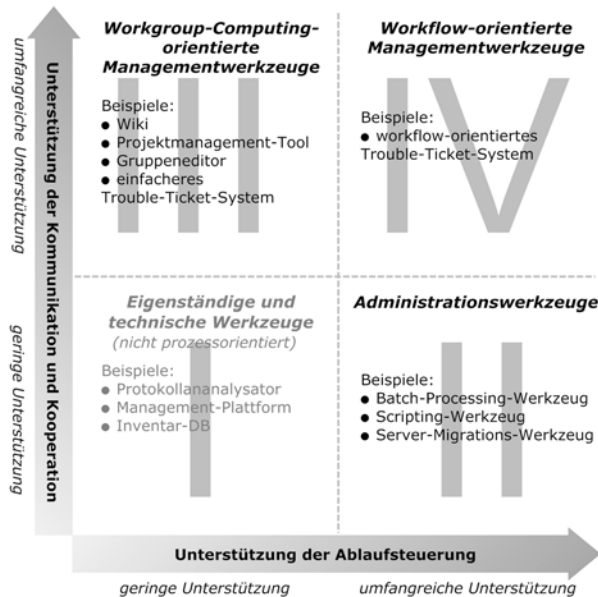


Abbildung 4.3: Grundlegende Arten von prozessorientierten ITSM-Werkzeugen

bzw. welche Prozesse hierfür für welche Typen von Werkzeugen geeignet oder ungeeignet sind, ist also eine nähere Untersuchung der Charakteristika der in der ITIL beschriebenen Dienstmanagementprozesse notwendig.

### 4.3 Klassifikation von ITIL-Prozessen

Um zu bestimmen welche Arten von Werkzeugen für welche ITIL-Prozesse geeignet sind, aber auch wie lohnend der Einsatz von Tools für die verschiedenen Prozess erscheint, ist es notwendig die in ITIL definierten Dienstmanagementprozesse entsprechend zu klassifizieren. Da existierende Ansätze zur Prozessklassifikation (siehe folgender Abschnitt 4.3.1) nicht darauf ausgelegt sind, abstrakte ITSM-Prozesse, wie sie mit den ITIL-Prozessen vorliegen, anhand von Werkzeugbezogenen Kriterien einzuordnen, werden (auf Basis von den in [Bre06] vorgestellten Ergebnissen) in Abschnitt 4.3.2 geeignete Kriterien entwickelt. Wie sich unter gewissen Grundannahmen (Abschnitt 4.3.3) die Kriterien für die ITIL-Prozesse bestimmen lassen wird in Abschnitt 4.3.4 diskutiert. Die zusammengefassten Ergebnisse (Abschnitt 4.3.5) dienen dazu auf Basis der Ausprägungen zweier Kriterien vier „Prozess-Quadranten“ zu definieren (Abschnitt 4.3.6), in

welche die ITIL-Prozesse platziert und somit in ihrer Eignung zu einer der im vorherigen Abschnitt 4.2 vorgestellten Kategorien von prozessorientierten ITSM-Betriebswerkzeugen zugeordnet werden können.

### 4.3.1 Prozessklassifikation in der Literatur

Die in der ITIL selbst vorgenommene Einteilung von Prozessen unterscheidet nur die Kategorien *Service Support* und *Service Delivery*. Auch die Einordnungen von Prozessen in der ITIL-Sekundärliteratur halten sich, wenn auch teilweise unter Verwendung einer anderen Terminologie (vgl. „taktische und operationelle Ebene“ in [Vog00]), an die hierdurch getroffene Aufteilung. Dies spiegelt zwar wider, welche der Prozesse einen größeren Zeithorizont haben und eher planerischen Natur sind (die *Service-Delivery-Prozesse*), trifft aber keine Aussage über die Unterstützbarkeit durch IT-Werkzeuge.

*Einteilung in der ITIL*

Auch allgemeine, nicht auf IT-Management beschränkte, Klassifikationsschemata für Prozesse helfen in diesem Punkt nicht weiter. Sie berücksichtigen in der Regel nur eine grobe Einteilung aus Sicht des strategischen Unternehmensmanagements, indem beispielsweise Prozesse in Kernprozesse (*Primary Processes*), unterstützende Prozesse (*Support Processes*) und Entwicklungsprozesse aufgliedert werden. Diese grobe funktionale Einteilung wird meist noch um eine Ebene verfeinert, also beispielsweise unterstützende Prozesse in Buchhaltung, Personalwesen usw. unterteilt (wie z.B. in [And00]). Für ITSM-Prozesse, die aus dieser Sicht – sofern überhaupt erfasst – wohl nur eine der nicht weiter verfeinerten Unterkategorie bilden würden, erlaubt dies also auch keine genauere Klassifikation als die durch ITIL gegebene.

*Allgemeine Klassifikations-schemata*

Hinsichtlich der Auswahl von Prozessen für Projekte zur Einführung von Workflow-Management-Systemen existieren aber immerhin Arbeiten, die den Einsatz entsprechender Kriterienkataloge vorschlagen [zM03]. Als die drei grundlegenden Hauptkriterien wird grundsätzliche Eignung des Prozesses für Workflowmanagement, die organisatorischen Rahmenbedingungen und potentielle Nutzeffekte einer Workflow-Automatisierung genannt. Dieser nicht ITSM-spezifische Kriterienkatalog ist allerdings nur auszugsweise veröffentlicht und dafür gedacht auf Ist-Prozesse, d.h. bereits existierende Prozessumsetzungen angewendet zu werden. Ebenfalls für die Anwendung auf Ist-Prozesse ausgelegt ist die Unterscheidung von Prozessen nach ihrer Strukturiertheit in [HEN01] – eine sehr strukturierte und vorhersagbare Interaktion zwischen den an einem Prozess beteiligten Personen ist beispielsweise für manche Produktionsprozesse ideal, aber tritt bei der Lösung kreativer Aufgaben seltener auf. Die Unterscheidung verschiedener Grade der Strukturiertheit, von ad-hoc über semi-strukturiert bis zu strukturiert, erlaubt die Platzierung von Prozessen in einem so genannten *Workflow-Kontinuum*.

*WfMS-bezogene Bewertung von Ist-Prozessen*

### 4.3.2 Kriterien für die Klassifikation von Dienstmanagementprozesse

Die folgende Klassifizierung erfolgt unter dem Aspekt der Werkzeugunterstützung – wie geeignet sind die einzelnen ITIL-Prozesse für den Einsatz bestimmter Tools (insbesondere Workflow-Management-Systeme) und wie wichtig bzw. lohnend wäre der Einsatz solcher Werkzeuge? Andere Gesichtspunkten, wie z.B. Reihenfolge bei der Einführung der Prozesses in eine Organisation, könnten zu völlig anderen Kriterien und resultierenden Einordnungen führen und werden nicht berücksichtigt.

**Fallfrequenz** Eine vergleichsweise offensichtliche Quelle von möglichen Kriterien für die Klassifikation, sind allgemeine Prozesskennzahlen, wie sie im Prozessmanagement typischerweise überwacht werden (vgl. Abschnitt 2.8). Ein Wert, dessen Erhebung praktisch immer nützlich ist, ist die Anzahl der Fälle bzw. Prozessinstanzen pro Zeit, also beispielsweise für das *Incident Management* die Anzahl bearbeiteter *Incidents* pro Woche. Im Folgenden sei diese Zahl im Allgemeinen *Fallfrequenz* genannt. Diese ist nicht nur für die Prozesskontrolle interessant, sondern auch für die Entscheidung für oder gegen einen Tool-Einsatz. Je öfter ein Prozess ausgeführt wird, desto öfter kommt auch ein ihn unterstützendes Werkzeug zum Einsatz. Unter ansonsten gleichen Rahmenbedingungen wird sich die Beschaffung (bzw. Entwicklung) und Einführung eines Werkzeuges also bei einem Prozess mit vielen Fällen pro Tag schneller amortisieren als bei einem Prozess, der vielleicht nur wenige Male pro Jahr ausgeführt wird.

**Durchlaufzeit** Eine weitere klassische Prozesskennzahl ist die *Durchlaufzeit*, also die Zeit, die vom auslösenden Ereignis (z.B. Anwender meldet ein *Incident*) bis zum Abschluss des Falles (z.B. *Incident Record* wird geschlossen) vergeht. Der Zusammenhang zwischen dem potentiellen Nutzen eines Tool-Einsatzes und der Durchlaufzeit ist nicht so eindeutig wie bei der Fallfrequenz. Zwar lässt eine lange Durchlaufzeit den Einsatz von Werkzeugen intuitiv weniger dringlich erscheinen, andererseits können bei langen Durchlaufzeiten und gleichzeitig hoher Fallfrequenz Tools von besonderem Nutzen sein, da Mitarbeiter ohne sie sonst rasch den Überblick über die resultierende hohe Zahl an gleichzeitig offenen Fällen verlieren können. Für eine klare Einteilung der Prozesse in solche, für die ein Tool-Einsatz besonders lohnend und solche, für die er weniger lohnend wäre, taugt die Durchlaufzeit also nicht ohne weiteres. Im Design von Werkzeugen kann sie aber durchaus Beachtung finden. Beispielsweise sind Werkzeuge, die zur Realisierung von ACID-Transaktionen Ressourcen blockieren für so genannte *long running processes* eher ungeeignet [Kay03]. Eine geringere Durchlaufzeit spricht letzten Endes aber meist von einer höheren Zeit-Kritikalität des Prozesses, so dass hier Automatisierungen von Aktivitäten und Eskalationsmechanismen (z.B. automatische Alarmierung des *Incident Manager*, wenn eine bestimmte Zeit seit der Meldung des *Incident* vergangen ist) tendenziell lohnenswerter erscheinen.

Wenn an einem Prozess viele Akteure beteiligt sind, wird dessen Implementie-

rung stärker von einem IT-Werkzeug profitieren, welches die Zusammenarbeit in dieser Gruppe unterstützt. Dies wird um so mehr gelten, falls außerhalb des Kontextes des Prozesses keine reguläre Kommunikation und Kooperation zwischen den Prozessbeteiligten gegeben ist, also beispielsweise, wenn die am Prozess Beteiligten auf mehrere Standorte verteilt sind oder unterschiedlichen Organisationen bzw. Organisationseinheiten angehören. Auch im Tool-Design sollten entsprechende Rahmenbedingungen berücksichtigt werden, da nicht alle Arten von Systemen gleichermaßen für einen organisationsübergreifenden Einsatz geeignet sind. Das zusammengesetzte Maß hierfür, also wie ausgeprägt und komplex die Arbeitsteilung innerhalb des Prozesses ist, wird im Folgenden *Organisatorische Komplexität* genannt.

*Organisatorische  
Komplexität*

Wie direkt ein Prozesses Einfluss auf die Erreichung von Unternehmenszielen hat, ist selbstverständlich auch ein Kriterium dafür, ob sich der Einsatz einer unter Umständen aufwändigen und kostspieligen Tool-Lösung rentiert (siehe auch Kriterien bei [zM03]). Wie bereits in Abschnitt 1.1.2 dargestellt, wird im IT-Dienstmanagement versucht den Einfluss, den die Erbringung der IT-Dienste auf die Unternehmensziele hat, auf konkrete Service-Parameter abzubilden. Von Interesse ist also hier primär die Einhaltung vereinbarter *Service Levels*. Auf diese wird tatsächlich jeder der in ITIL definierten Dienstmanagementprozesse eine potentielle Auswirkung haben – sonst müsste schließlich sein Sinn in Frage gestellt werden. Da die Dienstmanagementprozesse in der ITIL sich alle nicht direkt mit der Dienstleistung an sich befassen (wie es z.B. die Fulfillment-Prozessgruppe in der eTOM tut, vgl. Abschnitt 3.2.2), sondern die Erbringung des IT-Dienstes nur indirekt unterstützen, ist es schwer zu bestimmen, welche Prozesse nun darauf einen größeren Einfluss ausüben als andere. Bezüglich der Unmittelbarkeit des Einflusses kann aber durchaus unterschieden werden – es gibt hier also unterschiedliche Grade der Indirektheit bei der Unterstützung der Dienstleistung. Es gibt ITSM-Prozesse, deren ineffektive Ausführung einen direkten und sofortigen negativen Einfluss auf *Service Levels* haben würde, z.B. wenn eine Störungsmeldung verloren geht und ein kritischer *Incident* schlicht nicht behoben wird. Bei anderen Prozessen ist bei einem ähnlichen Vorfall die Auswirkung auf den *Service Level* mittelbarer. Insbesondere diejenigen Prozesse, deren Ziel in der Unterstützung anderer Dienstmanagementprozesse in planerischen Aufgaben besteht, haben mehr Einfluss auf die Effizienz als auf die Effektivität der Dienstleistung. Wenn also beispielsweise ein Teil der ermittelten Kapazitätsanforderungen im Rahmen des *Capacity Management* verloren geht, so muss dies keine Auswirkung auf die Dienstqualität haben – dies kann durch ein massives Overprovisioning an der oberen Grenze der nun ungenaueren Bedarfschätzung vermieden werden, allerdings eben unter Verlust an Kosteneffizienz. In der weiteren Diskussion soll, auch wenn dieser Punkt vom Kontext eines Einsatzszenarios stark abhängig ist, die Einhaltung von *Service Levels* als im Zweifelsfall wichtiger angesehen werden als die Erreichung der optimalen Kosteneffizienz.

*Unmittelbarkeit  
der  
Service-Level-  
Auswirkung*

Die Strukturiertheit ist ein Kriterium, das bereits im Workflow-Kontinuum (vgl.

### Strukturiertheit

Abschnitt 4.3.1) für die Identifizierung geeigneter Werkzeuge zur Prozessunterstützung herangezogen wird. Das Workflow-Kontinuum ist zur Einordnung von in der Praxis angetroffenen Ist-Prozessen gedacht, das grundsätzliche Konzept lässt sich aber für die abstrakten ITIL-Prozesse adaptieren. Zu beachten ist dabei aber, dass bei implementierten Prozessen ein geringes Maß an Strukturierung auch auf ein nicht durchgängiges Prozessdesign oder einen geringen Reifegrad der Prozessimplementierung zurückzuführen sein kann – der Begriff „Ad Hoc“ in Bezug auf die Prozessausführung wird nicht nur im Workflow-Kontinuum, sondern auch in Reifegradmodellen häufig zur Benennung des niedrigsten Reifegrades verwendet (beispielsweise in CobiT, vgl. Abschnitt 3.5). Ein Ad-Hoc-Prozess kann oftmals im Laufe der Zeit zu einem strukturierten Prozess reifen [HEN01]. Will man aber nicht konkret implementierte Dienstmanagementprozesse betrachten, sondern die abstrakten Prozessbeschreibungen der ITIL, so macht es keinen Sinn, wenn hierbei die Prozessreife (zumindest so wie sie in CMM, eSCM oder CobiT verstanden wird) einen Einflussfaktor darstellt. Um für ITIL-Prozesse unterschiedliche Stufen der Strukturierung zu unterscheiden, ist es daher notwendig, für alle betrachteten Prozesse die gleiche, also beispielsweise die höchste erreichbare Prozessreife anzunehmen. Unterschiede in der Strukturiertheit liegen dann in der generellen Natur des Prozesses, unabhängig von einer konkreten Umsetzung.

### Zusammenfassung der Kriterien

Zusammenfassend lassen sich also fünf grundlegende Kriterien für eine Klassifizierung von ITIL-Prozessen hinsichtlich einer Werkzeugunterstützung finden:

- A. Fallfrequenz** Die durchschnittliche Frequenz, mit der neue Prozessinstanzen auftreten, d.h. die Zahl der bearbeiteten Fälle pro Zeit. Falls ein klar strukturierter Workflow für den Prozess definiert ist, entspricht dies der Frequenz, in der auslösende Ereignisse (Startereignisse des Prozesses) auftreten.
- B. Durchlaufzeit** Durchschnittliche Lebensdauer einer Prozessinstanz, d.h. Zeit vom auslösenden Ereignis bis zum Abschluss des Falles.
- C. Organisatorische Komplexität** (kurz: Org. Komplexität) Ein Maß dafür, wie viele verschiedene Parteien am Prozess beteiligt sind, wie ausgeprägt die Arbeitsteilung ist und inwieweit erschwerende Faktoren die Kommunikation und Kooperation beeinflussen (beispielsweise geographische Distanz oder organisatorische Grenzen).
- D. Service-Level-Auswirkung** (kurz: SL-Auswirkung) Unmittelbarkeit der Auswirkung der Prozess-Performanz auf typische (insbesondere verfügbarkeitsbezogene) Service-Parameter.
- E. Strukturiertheit** Maß, inwieweit dem Prozess ein konkreter Workflow mit eindeutigem Kontrollfluss zugrunde gelegt werden kann.

Natürlich ließen sich diese fünf Kriterien bei Bedarf verfeinern oder auch ergänzen, insbesondere wenn die Prozesse hinsichtlich anderer Zielsetzungen als eine

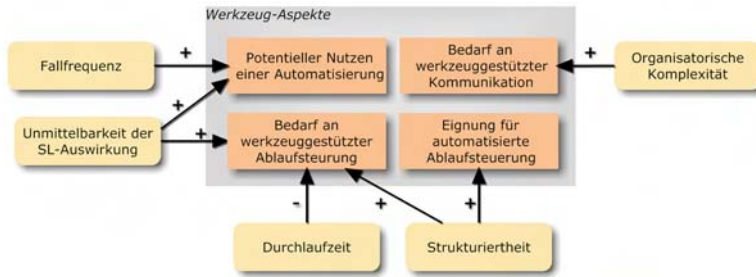


Abbildung 4.4: Einfluss von Kriterien zur Prozessklassifikation auf Aspekte der Werkzeugunterstützung

Werkzeugunterstützung untersucht werden sollten. Die vorliegenden Kriterien sind unter den vielen möglichen zur Prozessklassifikation allerdings diejenigen, welche die größte Auswirkung auf die beim Werkzeug-Einsatz zu treffenden Entscheidungen haben. Jedes der Kriterien beeinflusst entweder direkt, inwieweit sich ein Werkzeug-Einsatz lohnt oder welche Art von Werkzeug für die Unterstützung des jeweiligen Prozesses geeignet ist. Die Zusammenhänge sind in Abbildung 4.4 dargestellt und im folgenden noch einmal zusammengefasst.

*für andere Zwecke eventuell auch andere Kriterien sinnvoll*

Der Fallfrequenz und die Unmittelbarkeit der Auswirkung auf *Service Levels* bestimmen generell, wie groß der potentielle Nutzen einer Effektivitäts- oder Effizienz-Verbesserungen des Prozesses insgesamt ist. Beispielsweise wird Arbeitersparnis mittels einer Teilautomatisierung (also eine Effizienzverbesserung) für einen Prozess, der täglich hunderte Male ausgeführt wird, einen größeren Nutzen bringen als für einen Prozess, für den nur alle paar Monate ein Fall auftritt. Ebenso wird beispielsweise ein Werkzeugeinsatz lohnender, wenn die dadurch verbesserte Einhaltung von Prozesszielen einen unmittelbaren Einfluss auf die Einhaltung von *Service Levels* hat.

*Einfluss auf allgemeinen Werkzeug-Nutzen*

Da diese Ziele sich meist um die Einhaltung bestimmter Zeiten bei der Bearbeitung von Aufgaben drehen, beeinflusst die Unmittelbarkeit der Service-Level-Auswirkung auch den spezifischen Bedarf nach einem Werkzeug, das die Steuerung von Abläufen und dabei die Service-Level-orientierte Zuteilung von Ressourcen erleichtert. Eine sehr hohe Durchlaufzeit verringert hingegen wiederum in der Regel die Notwendigkeit, die Abläufe des Prozesses unbedingt automatisiert steuern zu müssen. Wo sich eine Prozessinstanz über Wochen hinzieht, kann meist die Zuteilung von Personal und anderen Ressourcen noch eher manuell gesteuert werden wie bei Prozessen, die innerhalb von wenigen Stunden abgeschlossen werden müssen und dabei häufig voraussetzen, dass gewisse Teilaufgaben nicht mehr als einige Minuten unbearbeitet bleiben. Wie strukturiert ein Prozess ist, hat ebenfalls Auswirkungen auf die Anforderungen in Bezug auf

*Einfluss auf Eignung für Werkzeugtypen*

Ablaufsteuerung – bei sehr strukturierten Prozessen wird meist auch gesteigerter Wert auf die Einhaltung des genauen Ablaufes gelegt, was durch ein geeignetes Werkzeug erleichtert (oder auch erzwungen) werden kann. Ein Mindestmaß an Struktur ist aber auch überhaupt die Voraussetzung, dass sich der Ablauf sinnvoll automatisiert steuern lässt. Die organisatorische Komplexität resultiert in einer Art von Anforderungen – sind die an einem Prozess beteiligten Parteien auf mehrere Orte verteilt und ihre Zahl hoch, so steigert dies den Bedarf die Kommunikation zwischen ihnen werkzeuggesteigert zu unterstützen.

### 4.3.3 Annahmen zum Einsatzszenario

*Keine  
Betrachtung  
konkreter  
Prozessumset-  
zungen*

Wie oben dargestellt, befassen sich bisherige Arbeiten zur Klassifikation von Prozessen, zumindest wo es um mehr als eine sehr allgemeine funktionale Einteilung geht, mit der Betrachtung konkreter, implementierter Prozesse. In diesen Fällen ist die Bestimmung der Prozesseigenschaften an konkreten Sachverhalten, möglicherweise sogar schon auf Basis bereits erhobener Daten, wie Prozesskennzahlen, möglich. Ein entsprechendes Vorgehen ist für eine allgemeine Betrachtung der ITIL-Prozesse nicht möglich. Das Problem beschränkt sich nicht darauf, dass ITIL-Prozesse abstrakter Natur sind und daher immer gewisse Freiheitsgrade für die konkrete Implementierung bleiben, die es unmöglich machen exakte Werte für bestimmte Prozessparameter zu bestimmen. Die ITIL ist auf eine enorme Variationsbreite (bezüglich Größe, Sektor usw.) von IT-Organisationen anwendbar, sie enthält daher zahlreiche Praktiken, deren Anwendung szenariospezifisch abgewogen werden sollte.

*Beschränkung  
der Betrachtung  
auf typischen  
IT-Service-  
Provider*

Um einigermaßen fundierte Aussagen über die Charakteristik einzelner ITIL-Prozesse machen zu können, müssen daher einige Grundannahmen zu den Rahmenbedingungen, in denen die Prozesse umgesetzt werden, getroffen werden. Da im folgenden eine Betrachtung möglichst vollständiger ITIL-Prozesse angestrebt wird, bleiben kleinere Einsatzszenarien in denen typischerweise vereinfachte Versionen der Prozesse zur Anwendung kommen, unberücksichtigt. Das grundlegende Einsatzszenario für die zu klassifizierenden ITIL-Prozesse wird daher wie folgt festgelegt:

*Eigenschaften  
des typischen  
IT-Service-  
Providers*

IT-Service-Provider, deren Organisation und technische Infrastruktur hinreichend komplex sind, dass die Umsetzung nahezu aller in ITIL *Service Support* [OGC00] und *Service Delivery* [OGC01] dokumentierten Praktiken sinnvoll und lohnenswert erscheint. Solche IT-Service-Provider haben folgende Charakteristiken:

- Unterstützung von geschäftlichen Endnutzern, d.h. ein signifikanter Anteil der Anwender nutzt die IT-Dienste zur Unterstützung von Geschäftsprozessen (nicht IT-Prozessen)
- mindestens mehrere hundert unterstützte Nutzer



- mindestens zweistellige Zahl von IT-Mitarbeitern
- mehrere angebotene IT-Dienste (bzw. unterstützte Applikationen)
- mehr als ein Kunde (d.h. für interne IT-Abteilungen, mehr als eine unterstützte Fachabteilung)
- eine zumindest partielle Softwareentwicklung oder Softwareanpassung, welche dediziert für die Kunden des IT-Service-Provider erfolgt<sup>3</sup>
- mindestens ein externer Sub-Provider

Auch wenn die bisherigen Umfragen zum ITIL-Einsatz wohl nicht im engeren Sinne repräsentativ waren, so weisen ihre Ergebnisse doch darauf hin, dass solche mittelgroßen bis großen IT-Organisationen als die typischeren und lohnenderen Szenarien für einen ITIL-Einsatz angesehen werden können (vgl. z.B. Diskussion von ITIL-Studien und Befragungen in [Buh05] oder „Respondent Demographics“ der INS-Studie [Blu04]).

#### 4.3.4 Bestimmung der Kriterienausprägung für ITIL-Prozesse

Im folgenden wird die Methode zur Bestimmung der Ausprägungen der Kriterien (sozusagen der Wertebelegungen) für die in ITIL festgelegten Dienstmanagementprozesse diskutiert. Auch unter den oben getroffenen Annahmen können natürlich keine präzisen numerischen Werte bestimmt, aber durchaus die Relationen zwischen den betrachteten Prozessen hinsichtlich der definierten Kriterien herausgearbeitet werden. So lässt sich bei einem betrachteten Paar von Prozessen für jedes Kriterium (mit wenigen, weiter unten noch diskutierten Ausnahmen) vergleichsweise klar sagen, ob dieses Kriterium für den einen Prozess im Vergleich zum anderen stärker, schwächer oder ungefähr gleich ausgeprägt ist (bzw. der entsprechende Wert höher, niedriger oder ungefähr gleich ist). Beispielsweise lassen sich *Incident Management* und *Problem Management* anhand der fünf festgelegten Kriterien wie folgt vergleichen:

*kriterienweises  
Vergleichen je  
zweier Prozesse*

- A. Fallfrequenz** Höherer Wert für *Incident Management*. Nach der Definition des Begriffes in ITIL ist ein *Problem* die „einem oder mehreren *Incidents* zugrunde liegende Ursache“<sup>4</sup> [OGC00]. Alleine hierdurch kann man davon ausgehen, dass es mehr *Incidents* als *Problems* gibt und dementsprechend im Regelfall auch mehr Instanzen des Prozesses *Incident Management* pro Zeit. Man kann natürlich argumentieren, dass im Laufe des so genannten *Proactive Problem Management* (vgl. Abschnitt 3.1.3 bzw. 6.8 in [OGC00]) ja auch Probleme bearbeitet werden, bevor sich *Incidents*

*Zahl der  
Incidents  
übersteigt die  
der Problems  
bei weitem*

<sup>3</sup>Nur für *Release Management* relevant, da ein Fehlen zu einem stark vereinfachten Prozess führen würde.

<sup>4</sup>Im Original: „unknown underlying cause of one or more incidents“

manifestieren können, es also durchaus *Problems* ohne entsprechende *Incidents* gibt. Diese Zahl wird aber um sehr viel niedriger sein als die Anzahl der *Incidents*, die nicht zur Erstellung eines neuen Problem-Records führen. Für Endnutzer-Services wird häufig, unter der impliziten Annahme einer ausreichend stabilen Infrastruktur und eines etablierten *Service Desk*, eine Erstlösungsquote (für *Incidents*) von 85% als ein typischer Wert genannt (siehe z.B. Kapitel zu *Service Desk* in [OGC00]). Da Lösungen von *Incidents* im *Service Desk*, mit wenigen Ausnahmen, keinen Anlass zum Erstellen neuer Problem-Records geben, kann davon ausgegangen werden, dass aufgrund der insgesamt hohen Anzahl von am *Service Desk* registrierten *Incidents*<sup>5</sup>, die Anzahl der durchschnittlich pro Zeit neu angelegten Problem-Records mindestens eine Größenordnung unter der der neu angelegten Incident-Records (pro Zeit) liegt.

*Incidents  
müssen schneller  
abgeschlossen  
werden als  
Probleme*

**B. Durchlaufzeit** Niedrigerer Wert für Incident Management. Von allen in der ITIL definierten Dienstmanagementprozessen ist das *Incident Management* der zeitkritischste. Die Mehrzahl der *Incidents* sollte, wie erwähnt, unmittelbar am *Service Desk* gelöst werden, ihre Bearbeitung also nur wenige Minuten in Anspruch nehmen. Für den Rest müssen häufig in einem SLA definierte Lösungszeiten eingehalten werden. Diese Vereinbarungen variieren in der Praxis, bewegen sich aber doch typischerweise für die garantierte maximale Incident-Lösungszeit in der Bandbreite von 2 Stunden bis 48 Stunden. Im Kontrast hierzu liegt die Aufgabe des *Problem Management* hauptsächlich in der Verbesserung der Qualität und Zuverlässigkeit und Infrastruktur – konsequenterweise muss im *Problem Management* die Gründlichkeit einen höheren Stellenwert einnehmen als die Schnelligkeit. Die Trennung des *Incident Management* von der Problembehandlung hat gerade das Ziel, dem IT-Fachpersonal die notwendige Zeit und Ruhe zur Analyse von Ursachen und möglichen Lösungsansätzen zu geben. Dementsprechend wird die Zeit bis zum Abschluss eines Problems im Durchschnitt sehr viel höher liegen als die Zeit zum Abschluss eines *Incident*.

*Typischerweise  
mehr  
Prozessbeteiligte  
im Incident  
Management*

**C. Organisatorische Komplexität** Stärkere Ausprägung (höherer Wert) für *Incident Management*. In der Ausführung des *Incident Management* wie des *Problem Management* sind, aufgrund der in beiden Prozessen vorkommenden, komplexen Infrastrukturanalyse-Aktivitäten, potentiell zahlreiche Stellen innerhalb des IT-Service-Providers sowie externer Sub-Provider involviert. Insgesamt müssen dennoch im *Incident Management* die größeren Herausforderungen hinsichtlich organisatorischer Komplexität bewältigt werden. *Incidents* werden in den meisten Fällen von Nutzern mit-

---

<sup>5</sup>Die Anzahl der am *Service Desk* gemeldeten *Incidents* ist einer der wenigen Parameter für die sich eine quantitative Schätzung, wenn auch nur der Größenordnung nach, abgeben lässt. Es scheint realistisch, wiederum bei einem gut etablierten *Service Desk* und einer reifen Dienstleistung, von einem bis wenigen *Incidents* pro Monat und Nutzer auszugehen. So rechnet beispielsweise das *Anwender-Service-Zentrum (ASZ)* der *BMW Group* bei 50.000 unterstützten Nutzern mit etwa 800.000 *Incidents* pro Jahr [FAS04].

tels Kontaktaufnahme mit dem *Service Desk* gemeldet. Dies sind bereits zwei im Prozess zu berücksichtigende Stellen, welche außerhalb der eigentlichen Kern-IT liegen und im *Problem Management* nicht auftauchen. Hinzu kommt, dass die Zeitkritikalität des *Incident Management* innerhalb einer Fallbearbeitung häufig zusätzliche Aktoren-Wechsel notwendig macht. Bleibt nämlich ein Incident mehrere Stunden ungelöst, so wird zum Ende der Arbeitszeit des aktuellen Incident-Bearbeiters oder dem Zeitpunkt eines Wechsels des aktiven *Service Desk* innerhalb einer *Virtual Service Desk Infrastruktur* (vgl. Abschnitt 4.2 in [OGC00]) in der Regel die Bearbeitung des *Incidents* nicht für beinahe einen Tag eingestellt werden, sondern eine Übergabe des Falles an einen anderen Mitarbeiter erfolgen.

**D. Service-Level-Auswirkung** Stärkere Ausprägung (höherer Wert) für *Incident Management*. Beide Prozesse üben einen starken Einfluss auf die Dienstverfügbarkeit aus: Die MTTR ist durch die Effektivität des *Incident Management* bestimmt, während *Problem Management* durch sein Ziel der Infrastrukturverbesserung die MTBF (bzw. *mean time between incidents*) maßgeblich beeinflusst (vgl. Diskussion zur Verfügbarkeit in Abschnitt 1.1.2). Der Effekt des *Problem Management* ist allerdings indirekter (wenn auch nachhaltiger) als der des *Incident Management*. Ein Versagen innerhalb des Incident-Managements, beispielsweise das Verlieren eines aktuellen Incident-Records, führt beinahe unweigerlich zu einer direkten Verletzung vereinbarter *Service Levels*.

*Auswirkung des Incident Management direkter*

**E. Strukturiertheit** Leicht stärkere Ausprägung für *Incident Management*. Zu *Incident Management* wie auch zu *Problem Management* lässt sich ein klarer, alle wesentlichen Aktivitäten umfassender Workflow bestimmen. Letztendlich ist aber, nach den Vorgaben der ITIL, für das *Incident Management* die Koordination zwischen den am Prozess beteiligten Parteien (beispielsweise durch den Mechanismus der funktionalen und hierarchischen Eskalation) klarer und spezifischer bestimmt.

*Incident Management etwas strukturierter*

Entsprechende Vergleiche lassen sich für alle Paarungen von ITIL-Prozessen durchführen. Für eine vollständige Einordnung der Prozesse ist es dabei nicht notwendig alle Kombinationen zu untersuchen. Auch wenn das in einigen Fällen gefundene Ergebnis „ungefähr gleich starke Ausprägung“ keine Äquivalenzrelation darstellt, so sind doch die „stärkere/schwächere Ausprägung“-Ergebnisse transitiv. Wenn beispielsweise bestimmt wird, dass die Durchlaufzeit im *Release Management* höher ist als im *Problem Management*, so folgt Zusammenhang mit obigem Ergebnis (Durchlaufzeit im *Problem Management* höher als im *Incident Management*) unmittelbar, dass im *Release Management* der Wert für dieses Kriterium auch höher als im *Incident Management* ist.

*wiederholtes Vergleichen führt zu vollständiger Einordnung*

Als Ergebnis kann für jedes der fünf Kriterien eine Ordnung der ITIL-Prozesse erstellt werden, wie sie beispielhaft für das Kriterium *Strukturiertheit* in Abbildung 4.5 dargestellt ist. Die unterschiedlich breiten und zu den Seiten hin

*Ordnung der ITIL-Prozesse*

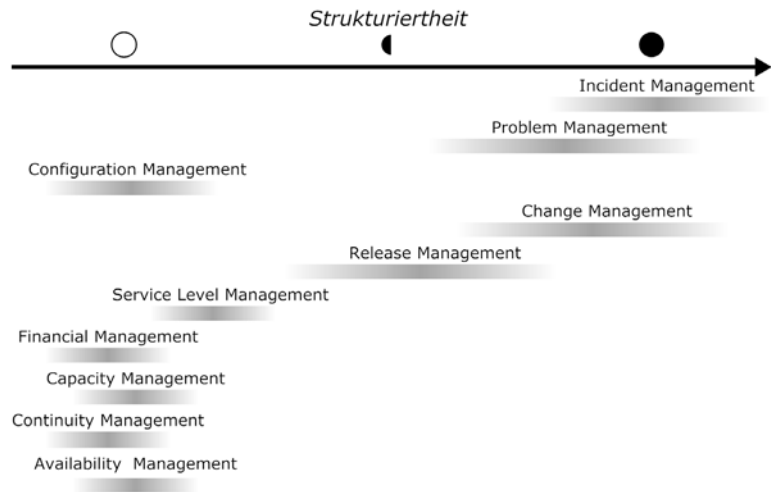


Abbildung 4.5: Relative Einordnung der Strukturiertheit der ITIL-Prozesse

auffhellenden Balken für die einzelnen Werte sollen dabei die Unschärfe wiedergeben, die auch unter den oben getroffenen Annahmen bei der Bestimmung der Werte unabhängig von einer konkreten Prozessimplementierung unvermeidbar ist.

### 4.3.5 Zusammenfassung der Ergebnisse

*Darstellung der Kriterienausprägung mittels dreier möglicher Wertebelegungen*

Um eine übersichtlichere Darstellung zu erleichtern, werden die Ergebnisse der Wertebestimmung in der folgenden Aufstellung für jedes Kriterium auf drei mögliche Werte eingeschränkt: Hoher Wert (starke Ausprägung) („●“), niedriger Wert (schwache Ausprägung) („○“) und mittlerer Wert (mittlere Ausprägung) („◐“). Es ist zu beachten, dass diese Werte nur als relativ im Spektrum der hier untersuchten Dienstmanagementprozesse zu sehen sind. Beispielsweise werden kreative Ad-Hoc-Prozesse (wie z.B. in [HEN01] diskutiert) ein geringeres, und industrielle Fertigungsprozesse ein höheres Maß an Strukturiertheit besitzen als irgendein ITSM-Prozess. Nach der Bestimmung der Werte für alle vorgeschlagenen Kriterien, stellt sich die Klassifikation der ITIL-Prozesse wie in Tabelle 4.6 gezeigt dar.

Werte, die nicht bestimmt werden können, sind in der Tabelle mit „na“ (=„nicht anwendbar“) gekennzeichnet. Dies ist bei den Kriterien *Fallfrequenz* und *Durchlaufzeit* für das *Configuration Management* wie auch für alle Prozesse der *Service*

	<i>Incident Management</i>	<i>Problem Management</i>	<i>Configuration Management</i>	<i>Change Management</i>	<i>Release Management</i>	<i>Service Level Management</i>	<i>Financial Management</i>	<i>Capacity Management for IT Services</i>	<i>IT Service Continuity Management</i>	<i>Availability Management</i>
Fallfrequenz	●	◐	na	◐	○	○	na	na	na	na
Durchlaufzeit	○	◐	na	◐	●	●	na	na	na	na
Org. Komplexität	●	◐	○	●	●	◐	○	○	○	○
SL-Auswirkung	●	◐	○	◐	◐	○	○	○	○	○
Strukturiertheit	●	●	○	●	◐	○	○	○	○	○

Abbildung 4.6: Charakteristiken der ITIL-Prozesse

*Delivery* mit Ausnahme des *Service Level Management* der Fall. Dies hängt mit dem Umstand zusammen, dass für diese Prozesse keine klare Workflow-Struktur definiert werden kann, also das Kriterium *Strukturiertheit* schwach ausgeprägt ist. So wie diese Prozesse in ITIL beschrieben sind, ähneln sie weniger einem Prozess im klassischen Sinne, sondern eher einer Sammlung von kleineren Prozessen und Aktivitäten, die zur Erreichung eines bestimmten Zieles (z.B. fundierte Planung unter Verfügbarkeitsaspekten) beitragen sollen. Abgesehen von diesem gemeinsamen Ziel, ist der Zusammenhang zwischen diesen Teilprozessen und Aktivitäten aber eher lose.

*keine klare  
Workflow-  
Struktur für fast  
alle Prozesse der  
Service  
Delivery...*

Gleiches gilt für das *Configuration Management*. Ähnlich wie bei einigen anderen Prozessen werden in der ITIL hierfür, neben dem was man normalerweise als die wiederkehrenden Prozessaktivitäten ansehen würde, auch mehr oder weniger einmalige Prozessplanungs- und Einführungsschritte als Prozessaktivitäten definiert (z.B. *Configuration Management Planning*, vgl. Abschnitt 7.6 in [OGC00]) oder periodische Überprüfungsaufgaben (z.B. *Configuration Verification and Audit*, vgl. Abschnitt 1 7.6 in [OGC00])<sup>6</sup>. Diese Prozesse folgen also keinem klaren Workflow, mit einem auslösenden Ereignis und einem oder mehreren möglichen

*...ebenso wie für  
Configuration  
Management*

<sup>6</sup>Hier werden in der ITIL also, ohne dass dies explizit gemacht oder kommentiert würde, in die Beschreibung des Prozessablaufs bzw. -ausführung auch Elemente hinzugenommen die eigentlich eher der *Design*- oder *Analyse*-Phase des Prozessmanagements zuzuordnen wären.

Ergebnissen bzw. Endereignissen. Vielmehr bestehen sie aus einer Ansammlung verschiedener Aktivitäten, Prozeduren und Richtlinie, zwischen denen sich kaum Zusammenhänge bezüglich gemeinsamer auslösender Ereignisse oder der relativen zeitlichen Abfolge herstellen lassen.

*Fallfrequenz und Durchlaufzeit für schwach strukturierte Prozesse nicht klar bestimmbar*

Dementsprechend lässt sich für diese Prozesse nicht sagen, was nun eine Prozessinstanz (oder Fall) ist. Dieses lässt sich höchstens für die einzelnen Teilprozesse und Aktivitäten festlegen, nicht aber für den Gesamtprozess, wie er in der ITIL beschrieben wird. Da man nicht definieren kann, was eine Prozessinstanz (beispielsweise des *Configuration Management*) ist, macht es auch keinen Sinn über die Häufigkeit oder die Durchlaufzeit von Instanzen Aussagen zu treffen. Auch zu versuchen, einen Durchschnitt über die einzelnen Teilprozesse zu bilden, wird, aufgrund der teils sehr unterschiedlicher Natur, kaum aussagekräftige Resultate liefern – beispielsweise ist im *Configuration Management* die *Configuration Identification* ein langwieriger Planungsprozess, während die meisten unter *Control of CIs* zusammengefassten Aktivitäten vergleichsweise simpel und schnell vollzogen sind (z.B. ein Update eines CI-Records in der CMDB vornehmen).

*Fallfrequenz der Teilprozesse als Zwischenergebnis nutzbar*

Allerdings kann bei diesen schwach strukturierten Prozessen die Fallfrequenz für eine Gewichtung der einzelnen Teilprozesse untereinander, wie sie für die Bestimmung der anderen Kriterien sinnvoll ist, dienen. So lässt sich für das *Configuration Management* prinzipiell sowohl für eine hohe wie auch eine niedrige *organisatorische Komplexität* argumentieren: *Configuration Identification* wird in der Regel praktisch alle IT-Abteilungen sowie Beteiligte aller anderen Dienstmanagementprozesse involvieren, während *Status Accounting* und *Control of CIs* in der Regel nur eine gut formalisierbare Kooperation mit dem *Change Management* und *Release Management* erfordern. Da letztere Aktivitäten aber ständig im täglichen Betrieb auftreten, während eine einmal zufriedenstellend abgeschlossene *Configuration Identification* unter Umständen für Monate oder sogar Jahre nicht wiederholt werden muss, werden die beiden erstgenannten Aktivitäten höher gewichtet und dem *Configuration Management* eine relativ geringe organisatorische Komplexität zugeschrieben (wenn diese auch höher ausfällt als bei den meisten Prozessen der *Service Delivery*).

*Bewertung der Fallfrequenz und Durchlaufzeit beim SLM auf „On-going Process“ beschränkt*

Eine Ausnahme unter den Prozessen der *Service Delivery* bildet das *Service Level Management* – es ist der einzige Prozess, für den trotz einer vergleichsweise schwach ausgeprägten Struktur Aussagen zu Fallfrequenz und Durchlaufzeit getroffen werden werden können. Zwar sind auch für das *Service Level Management* keine klaren Start- und Endereignisse definierbar und es daher nur schwer möglich einen sinnvollen Workflow, beispielsweise mittels einer EPK, zu definieren (vgl. [Has04]). Allerdings lässt sich der zentrale, mit der Pflege bestehender SLA-Beziehungen befasste Anteil *Service Level Management* (*On-going Process*, vgl. Abschnitt 4.5 [OGC01]) als ein zyklischer Prozess auffassen, dem eine Zykluszeit (also Durchlaufzeit)<sup>7</sup> und eine Fallfrequenz (die Zahl der gepflegten SLAs geteilt

<sup>7</sup>hier die Zeit zwischen zwei, ein SLA betreffendes *Service Review Meetings*

durch die durchschnittliche Zeit zwischen zwei *Service Review Meetings*) zugeordnet werden kann. Die angegebenen relativen Werte für diese beiden Kriterien beziehen sich also nur auf diesen einen, aber dominanten Teil des Prozesses.

### 4.3.6 Einteilung in Quadranten der Werkzeugeignung

Basierend auf obigen Ergebnissen sowie des Einordnungsschemas für prozessorientierte Dienstmanagement-Betriebswerkzeuge aus Abschnitt 4.2.3 kann man die ITIL-Prozesse hinsichtlich ihrer Bedarfs an Werkzeugunterstützung und der hierfür am geeignetsten erscheinenden Werkzeugtypen einordnen.

#### Einordnung der ITIL-Prozesse nach ihren Werkzeug-relevanten Charakteristika

Wie in Abbildung 4.4 zusammengefasst, lohnt sich eine Unterstützung mittels IT-Werkzeugen vor allem für jene Dienstmanagementprozesse, die eine hohe Fallfrequenz haben. Auch eine unmittelbare Auswirkung der Prozesseffektivität auf *Service Levels* lässt die Automatisierung bestimmter Prozessaspekte besonders attraktiv erscheinen. Eine prozessorientierte Werkzeugunterstützung sollte also zunächst für *Incident Management*, *Change Management* und *Problem Management* überlegt werden.

*für welchen  
Prozess  
Tool-Einsatz  
besonders  
lohnend?*

Dies sagt aber noch nichts über die Eignung dieser Prozesse für bestimmte Arten von prozessorientierten Tools, wie beispielsweise Workflow-Management-Systeme aus. Um dies zu bestimmen, lohnt sich ein Blick auf die Ausprägungen der restlichen Kriterien für die ITIL-Prozesse. Möchte man alle ITIL-Prozesse betrachten, so lässt sich allerdings die Durchlaufzeit hierfür nur bedingt heranziehen, da ihre Einschätzung für viele Prozesse nicht möglich ist – somit bleiben die beiden Kriterien Strukturiertheit und organisatorische Komplexität.

*welcher Typ von  
Werkzeug?*

Anhand dessen, wie stark diese beiden Kriterien jeweils ausgeprägt sind, lassen sich die ITIL-Prozesse, wie in Abbildung 4.7 zu sehen, in vier Quadranten einteilen, welche im Großen und Ganzen den in Abbildung 4.3 dargestellten Tool-Kategorien zugeordnet werden können. Die Platzierung der Prozesse innerhalb der Quadranten gibt die Tendenz ihrer Zuordnung wieder, welche sich durch die Ordnung der Prozesse nach den beiden Kriterien ergibt – beispielsweise ist *Release Management* strukturierter als *Service Level Management* (vgl. Abbildung 4.5) und ließe sich daher eher als letztgenanntes auch Quadrant IV zuweisen. Die Eigenschaften der Prozesse in den verschiedenen Quadranten lässt sich wie folgt zusammen fassen:

*vier Quadranten*

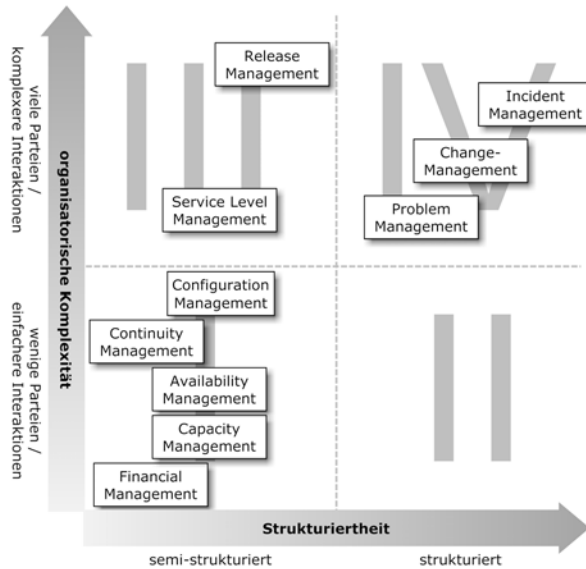


Abbildung 4.7: Einordnung von ITIL-Prozessen in vier Quadranten

### Quadrant I

*geringe  
Strukturiertheit  
/ geringe  
organisatorische  
Komplexität*

Prozesse in Quadrant I haben keine klare, den ganzen Prozess umfassende Ablauf-Struktur und eine vergleichsweise geringe organisatorische Komplexität mit wenigen beteiligten Parteien. Bei den in diesem Quadranten platzierten ITIL-Prozessen handelt es sich in der Hauptsache um Planungs- und Kontrollprozesse, die unterschiedliche, weder durch einen klaren Ablauf noch einen gemeinsamen zeitlichen Rahmen gekoppelte Aktivitäten umfassen. Vergleicht man die Natur dieser Prozesse mit der Prozessdefinition der DIN (vgl. Abschnitt 2.3), so kann man sagen, dass die Wechselbeziehungen zwischen den Mitteln und Tätigkeiten hier sehr schwach ausgeprägt sind und sich teilweise auf den Beitrag zu einem gemeinsamen Ziel beschränken.

*Einordnung  
Configuration  
Management*

Die Aktivitäten in den hier zu findenden ITIL-Prozessen verarbeiten zwar Informationen, die ihnen zum wesentlichen Teil aus anderen ITSM-Prozessen geliefert werden – allerdings fließt die Information in der Regel nur in eine Richtung, so dass dies keine Kooperation im engeren Sinne darstellt. Dies gilt, selbst wenn es noch am stärksten zu einer Einordnung in Quadrant 3 tendiert, auch für das *Configuration Management*. Es ist allerdings eine Frage der Interpretation, ob man die Bereitstellung von CMDB-Information als Prozessaktivität (was für eine



höhere organisatorische Komplexität spräche) oder als Funktion der von diesem Prozess gepflegten, aber eben nicht verkörperten CMDB sieht – hier wird letztere Sicht eingenommen und *Configuration Management* daher in Quadrant I eingeordnet. Die einzelnen Aktivitäten dieser Prozesse können natürlich sehr wohl von einem Einsatz geeigneter Tools<sup>8</sup> profitieren, für eine im eigentlichen Sinne prozessorientierte Werkzeugunterstützung fehlen aber die Voraussetzungen.

### Quadrant II

In Quadrant II einzuordnende Prozesse sind strukturiert, haben aber eine geringe organisatorische Komplexität. In Betriebshandbüchern definierte, von einem einzelnen Administrator abarbeitbare Abläufe (z.B. für das Patch-Management eines Server-Produktes) wären ein Beispiel für solche Prozesse. Derartige, im IT-Betrieb nicht selten zu beobachtende Prozesse finden sich allerdings nicht in der ITIL – diese legt ihren Schwerpunkt woanders. Zahlreiche IT-Management-Tools, die eine Teilautomatisierung von typischen Aufgaben des IT-Operations (z.B. Server-Migration, Patch-Management, Stapel-Verarbeitung von Routine-Aufgaben) anstreben, können als Werkzeuge für Prozesse des Quadranten II angesehen werden.

hohe  
Strukturiertheit  
/ geringe  
organisatorische  
Komplexität

### Quadrant III

Die Aktivitäten der Prozesse in Quadrant III involvieren vergleichsweise viele Parteien, teilweise über organisatorische Schranken – allerdings ohne übergreifenden, strukturierten Ablauf. Sowohl *Service Level Management* wie auch *Release Management* haben strukturierte Anteile, diese ergeben aber insgesamt keinen klaren Prozessablauf. Sie umfassen aber Aktivitäten, die eine Kooperation und wechselseitigen Informationsaustausch zwischen zahlreichen beteiligten Parteien erfordern, und scheinen daher, wenn auch nicht für WfMS, aber für den Einsatz von Workgroup-Computing-Werkzeugen geeignet (vgl. Abbildung 4.3).

geringe  
Strukturiertheit  
/ hohe  
organisatorische  
Komplexität

### Quadrant IV

Nur die in diesem Quadranten zu findenden Prozesse entsprechen einem „klassischen“ Prozessbild mit einem klaren Ablauf, welcher eine nennenswerte Anzahl unterschiedlicher Parteien einbindet. Nur drei der zehn in der ITIL definierten Dienstmanagementprozesse gehören hierzu: *Incident Management*, *Change Management* und *Problem Management*. Auch die vergleichsweise geringen Durchlaufzeiten sprechen für den Einsatz von Workflow-Management-Systemen für die-

hohe  
Strukturiertheit  
/ hohe  
organisatorische  
Komplexität

<sup>8</sup>An der entsprechenden Stelle in Abbildung 4.3 finden sich keine konkreten Beispiele für ITIL-bezogene Tools dieser Gattung – die *Capacity Management* Capacity Database (vgl. Abschnitt 3.1.3) wäre aber beispielsweise hierzu zu zählen.

se drei Prozesse, welche aufgrund ihrer hohen Fallfrequenz und SL-Auswirkung bereits als die lohnendsten Kandidaten für eine Werkzeugunterstützung identifiziert worden sind.

### Diskussion

*Einordnung bleibt relativ* Es muss beachtet werden, dass auch diese Einordnung relativ und nicht absolut ist. Ist ein Szenario groß und komplex, so steigt generell für alle Prozesse die organisatorische Komplexität. In solch einem Fall würden sich die in Quadrant I eingeordneten Prozesse in Richtung Quadrant III verschieben. Prinzipiell kann unter bestimmten Umständen also auch für die hier dem Quadranten I zugeordneten Prozesse ein Einsatz von Workgroup-Computing-Werkzeugen sinnvoll sein.

*Sonderfall CMDB* Einen oft missverstandenen Sonderfall nimmt die *CMDB* ein. Die *CMDB* taucht in der ITIL häufig als Begriff auf – dabei sind ihre Konzepte und Aufgaben aber so vage formuliert, dass ein erheblicher Interpretationsspielraum bleibt. So kann sie als eine Datenbank für ein Infrastruktur-Modell verstanden werden, als eine Plattform für eine informationsorientierte Integration zwischen den ITIL-Prozessen oder eine Kombination und Integration von beidem (vgl. [BGSS06]). Sie stellt aber kaum ein prozessorientiertes Werkzeug zur Unterstützung des *Configuration Management* an sich dar. Die ITIL liefert keine Anhaltspunkte, wie die Funktionalität der *CMDB* zu der Steuerung und Koordination der diversen Aufbau-, Erfassungs- und Pflegeaktivitäten (vgl. Abschnitt 3.1.3) beitrüge.

## 4.4 Zusammenfassung und Fazit

*ITIL-Prozesse unterschiedlicher als unmittelbar offensichtlich* Auch wenn es nicht immer auf den ersten Blick sofort ersichtlich ist, so unterscheiden sich die ITIL-Prozesse doch nicht nur in ihrer funktionalen beziehungsweise fachlichen Ausrichtung (z.B. Managen von Infrastruktur-Änderungen versus Managen von SLAs), sondern auch in ihrer Natur als Prozess. Viele von ihnen entsprechen nicht dem typischen, klassischen Bild eines strukturierten Prozesses und folgen nicht einem einzelnen, klar umreißbaren Ablauf.

*Unterstützung für Incident Management am lohnendsten* Eine Automatisierung bringt auch nicht für alle Dienstmanagementprozesse den gleichen Nutzen. Betrachtet man die Ergebnisse der Untersuchung der ITIL-Prozesse (vgl. Abbildung 4.6 auf Seite 119), so zeigt sich, dass eine Werkzeugunterstützung des *Incident Management* aufgrund seiner hohen Fallfrequenz und der unmittelbaren Auswirkung der Prozesseffektivität auf *Service Levels* (vgl. dazu Abbildung 4.4 auf Seite 113) am lohnendsten erscheint. Auch für *Problem Management* und *Change Management* würde aber wohl auch ein vergleichsweise hoher Aufwand bei der Umsetzung eines Werkzeugkonzeptes rentieren können.

Es ist intuitiv einsichtig, dass für eine umfassende und optimale Unterstützung der ITIL eine Vielzahl verschiedenster IT-Tools erforderlich wäre. Die vielfältigen Aufgaben der zehn ITIL-Prozesse, die unterschiedlichste Bereiche von Infrastruktur-Fehlerdiagnose bis zur Rechnungsstellung berühren, bieten zahlreiche Ansatzpunkte für den Einsatz diverser, funktional spezialisierter Werkzeuge. Die meisten dieser in der ITIL dargelegten Teilaufgaben sind dabei nicht als originär anzusehen, auch wenn sie oft erst mit der Popularisierung der ITIL in das Blickfeld des Managements mancher IT-Betriebsorganisation gekommen sind. Entsprechend existieren für viele Teilaufgaben der Prozesse und Aktivitäten durchaus geeignete Werkzeuge.

*Prinzipiell viele  
Tools von  
Nutzen*

Die Herausforderung bei der Werkzeugunterstützung für ITIL liegt jedoch nicht nur im Füllen der noch vorhandenen funktionalen Lücken bei der Erleichterung und Automatisierung einzelner Arbeitsschritte, welche die an sich zahlreichen Management-Tools noch offen lassen. Hierfür gibt es aus den Forschungsabteilungen einiger großer Hersteller von IT-Management-Werkzeugen vielversprechende Arbeiten (z.B. [BS04] oder [Kel05]), die aber, auch wenn ihre Titel manchmal etwas anderes suggerieren, immer nur isolierte Aspekte der Gesamtprozesse adressieren.

*Herausforderungen nicht nur  
in einzelnen  
Teilaktivitäten*

Welcher der Dienstmanagementprozesse überhaupt besonders dringend unterstützt werden muss, welche Aspekte oder Prozessteile adressiert werden und mit welcher Art von Werkzeug dies versucht werden soll – das folgte bislang keinem umfassenden, kohärenten Ansatz. Bisher gibt es auch kein allgemeines, herstellerunabhängiges und nachvollziehbares Konzept für ein Management-System, das einen ITIL-Prozess als Ganzes unterstützt – also die Steuerung seines Ablaufes erleichtert und die verschiedenen, für die einzelnen Aktivitäten einsetzbaren Werkzeuge prozessorientiert integrieren kann. Grundvoraussetzung hierfür wäre unter anderem ein gemeinsames Verständnis von Werkzeugtypen und wie sich diese in ein Rahmenkonzept zur Werkzeugunterstützung für ITIL-Prozesse einbauen lassen. Die in diesem Kapitel vorgestellten Klassifizierungen und Einordnungen können hierfür jedenfalls eine Basis bilden.

*Klassifizierung  
notwendige  
Basis für  
effektive  
Werkzeug-  
Konzepte*



## 5 Modellbasierter Ansatz

Trotz der großen Allgemeinheit und Unverbindlichkeit der ITIL-Richtlinien existieren verschiedene Software-Produkte, die von ihren Herstellern als ITIL-Werkzeuge bezeichnet werden. Doch es gibt hierbei weder hinsichtlich der für ein ITIL-Tool geforderten Eigenschaften noch in Bezug auf Konkretisierungen der ITIL-Inhalte, wie sie für eine zielgerichtete Werkzeug-Entwicklung unumgänglich wären, einen allgemeinen Konsens. Im folgenden Abschnitt (5.1) wird zunächst dargelegt, wie ein modellbasierter Ansatz für die Werkzeugunterstützung von ITIL-Prozessen einen Ausweg aus dieser verfahrenen Situation bieten könnte.

Im Rahmen dieser Arbeit soll ein entsprechender Ansatz zunächst nur für eine ausgewählte Klasse von Prozessen und Werkzeugen (siehe Abschnitt 5.2) entwickelt und beispielhaft umgesetzt werden. Die in Abschnitt 5.3 vorgestellten Ziele bei der Modellerstellung und die daraus entwickelten Richtlinien für die Erstellung von Referenzmodellen für ITIL-Prozesse sind aber für alle Dienstmanagementprozesse der ITIL gültig. In Abschnitt 5.4 wird ein auf dem *Zachman Framework* basierender Ordnungsrahmen für Modelle von Dienstmanagementprozessen und entsprechenden prozessorientierten Werkzeugen sowie ein an diesen Rahmen angepasstes Vorgehen für die Entwicklung eines entsprechenden Gesamtmodells vorgestellt.

### 5.1 Motivation

Was ist nun vorteilhaft daran, die gewünschten Werkzeuge nicht mehr oder weniger direkt zu programmieren, sondern den Umweg über Modelle zu gehen? Es gibt viele stichhaltige Argumente für die Verwendung von Modellen in der Software-Entwicklung allgemein, wie beispielsweise die Erleichterung einer späteren Anpassung oder Integration des zu entwickelnden Systems (für eine Diskussion der Vorzüge einer modellbasierten Entwicklung siehe z.B. [OMG03] oder fast jedes Buch zu UML).

*Allgemeine  
Vorteile*

Im Falle der Entwicklung von ITIL-Tools gelten aber darüber hinaus noch weitere Vorteile. Wie bereits in Abschnitt 4.1 dargelegt, ist es praktisch völlig unklar, welche Eigenschaften ein ITIL-Werkzeug haben sollte. Der Weg zu einem ITIL-Tool umfasst nicht nur den System- und Programmentwurf, mit all seinen Designentscheidungen und Konzeptentwicklungen, sondern auch einen die ITIL konkretisierenden Prozessentwurf. All diese Designs und Konzepte bleiben

*Situation bei  
ITIL-Tools*

aber undokumentiert (oder zumindest die Dokumentation unveröffentlicht). Der Vergleich zweier unterschiedlicher ITIL-Werkzeuge, die reklamieren den gleichen ITIL-Prozess zu unterstützen, fällt daher oft so schwer: sie gehen wahrscheinlich von unterschiedliche Interpretationen des Prozesses aus, diese lassen sich aber höchstens sehr mühsam mittels „Reverse Engineering“ auf Basis der Funktionalität der fertigen Tools nachvollziehen.

### *Vorteile von Modellen für Entwicklung von ITIL-Tools*

Ein modellbasierter Ansatz, also die Dokumentation (mittels Modellen) der Zwischenergebnisse auf dem Weg zu einem ITIL-Werkzeug, von der ITIL-Konkretisierung bis zur technischen Spezifikation, kann hier einen Ausweg aus der bislang reichlich verfahrenen Situation in der ITIL-Toolandschaft bieten. Das Fassen von Zwischenergebnissen der Entwicklung in Modelle bietet mehrere Vorteile: Neben dem klassischen Vorzug der verbesserten Anpassbarkeit und Integrierbarkeit des fertigen Systems kann dieses auch in seiner Funktionalität vom Anwender besser eingeschätzt werden. Für den Anwender wird beispielsweise nachvollziehbar, wie die Prozesse, die er mit dem Werkzeug unterstützen will, in den Grundzügen denjenigen Vorstellungen von ITIL-Prozessen entsprechen, die als Basis für den Werkzeugentwurf dienen. Auch können die Zwischenergebnisse an sich eine Referenz für eine Prozessimplementierung, den Neuentwurf oder die Weiterentwicklung eines Systems bilden (letzterer Umstand dürfte aber Hersteller kommerzieller Werkzeuge natürlich von der Veröffentlichung von wiederverwendbaren Modellen eher abhalten).

### *Basis für die Entwicklung eines allgemeinen Tools?*

Es stellt aber die Frage, von welcher Basis man bei den zu unterstützenden ITIL-Prozessen ausgehen kann, also wie Modelle der ITIL zu erstellen sind. Die Inhalte der ITIL reichen für sich genommen kaum für die Entwicklung eines Werkzeug-Konzeptes aus, sie müssen konkretisiert werden. Es wird von Beratern oft argumentiert, dass diese Konkretisierung im Sinne eines für die ITIL gepredigten „adopt and adapt“ für jedes Einsatzszenario individuell vorgenommen werden muss. Dies würde bedeuten, dass sich konkretere Modelle nur für jeweils eine Zielorganisation anfertigen ließen. Aber es liegt nicht nur an Spezifika konkreter Szenarios, wenn die ITIL auf unterschiedliche Art interpretiert wird – dies ist auch zu einem guten Teil der unstrukturierten und teils widersprüchlichen Darstellung ihrer Inhalte geschuldet. Unter ITIL-Beratern herrschen nicht selten unterschiedliche Meinungen über die Interpretation von ITIL-Inhalten, die spezieller sind als die allgemeinen, in den Zertifizierungen geprüften Fragestellungen.

### *Referenzmodelle als Lösungsansatz*

Es lässt sich nicht vermeiden, dass ITIL-Inhalte von Person zu Person und abhängig vom Einsatzszenario immer wieder anders interpretiert werden. Es ist aber frustrierend, dass diese verschiedenen Interpretationen kaum konkret miteinander (oder der ITIL selbst) vergleichbar oder vernünftig diskutierbar sind – für beides fehlen einheitliche Darstellungsmittel und Referenzpunkte. Eine effiziente Abhilfe könnte hier nur die Schaffung klarer, beispielhafter Referenzmodelle der Prozesse schaffen. Unter einem Referenzmodell wird allgemein ein Modell

verstanden, welches für eine Klasse von Entscheidungsproblemen oder Design-Fragestellungen, Hilfestellung in der Form von beispielhaften Lösungsschemata und generellen Rezepten liefert. Referenzmodelle vermitteln somit Wissen über bestimmte Problemomänen mittels eines allgemeinen, repräsentativen, manchmal auch normativen Modells. Ein Referenzmodell kann als Basis für das Design eines speziellen Modells oder Vergleichsobjekt für die Einordnung und Beurteilung existierender spezieller Modelle verwendet werden [Pro03].

## 5.2 Abgrenzung

Im folgenden wird zunächst festgelegt, für welche primäre Anwendungsdomäne der Ansatz gelten soll. Zunächst soll er natürlich die Entwicklung von Werkzeugen zur prozessorientierten Unterstützung eines ITIL-orientierten Dienstmanagements ermöglichen. Auch wenn ein für ITIL geeigneter Ansatz zumindest partiell wohl ebenso für in der grundlegenden Ausrichtung verwandte Rahmenwerke (vgl. Abschnitt 3.11) anwendbar wäre, sind diese nicht im Fokus. Innerhalb der ITIL soll die Anwendung dabei auf die Richtlinien zur Gestaltung der Prozessausführung beschränkt sein, auf der auch der Schwerpunkt der ITIL liegt (vgl. Abschnitt 3.11.2).

*nur ITIL im Fokus*

Da die ITIL-Prozesse sich nicht nur in ihren Aufgabenstellungen, sondern auch in ihren grundlegenden Eigenschaften zum Teil erheblich voneinander unterscheiden, ist es nicht sinnvoll, Tools für alle Dienstmanagementprozesse gleichzeitig entwickeln zu wollen. Es ließe sich kaum ein Modellrahmen finden, der zur Beschreibung aller Prozesse gleich geeignet wäre, und es müssten grundlegend unterschiedliche Arten von Werkzeugen adressiert werden. Wie in Abschnitt 4.3 gesehen, ist eine (Teil-)Automatisierung aber nicht für alle Dienstmanagementprozesse der ITIL gleich wichtig. Am insgesamt lohnendsten erscheint die Unterstützung für das *Incident Management*. Aber auch die Ausführung von *Change Management* und *Problem Management* würde vom Einsatz prozessorientierter Werkzeuge profitieren. Innerhalb des in Abschnitt 4.3.6 vorgestellten Einordnungsschemas finden sich diese drei Prozesse alle im Quadrant IV. Dank dieses Umstandes kann für alle drei das prinzipiell gleiche Vorgehen angewandt werden: Da es sich um Prozesse ähnlicher Natur handelt, kann ihre Modellierung unter identischen Aspekten erfolgen und für alle drei ist auch der gleiche grundlegende Typ Werkzeug einsetzbar (Workflow-orientierte Management-Werkzeuge). Die grundlegende Abgrenzung der primären Anwendungsdomäne des weiter unten vorgestellten Ansatzes ist in Abbildung 5.1 noch einmal bildlich dargestellt.

*Beschränkung auf Prozesse im Quadrant IV*

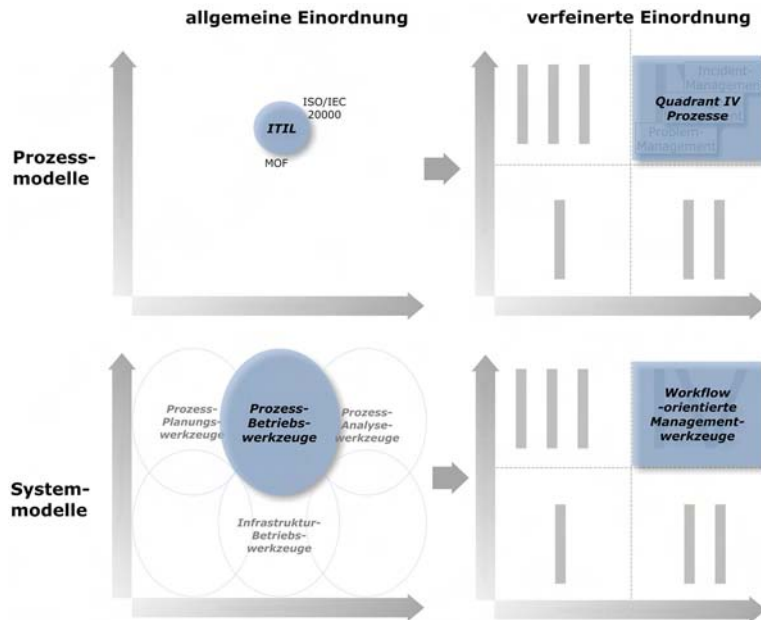


Abbildung 5.1: Anwendungsdomäne des Ansatzes

### 5.3 Ziele

Referenzmodelle für ITIL-Prozesse sollten gewisse Voraussetzungen erfüllen, damit sie tatsächlich als Basis für die Entwicklung allgemein anwendbarer Werkzeuge dienen können. Zunächst gelten für sie im Prinzip die gleichen Ansprüche wie für Referenzmodelle im allgemeinen. Eine Sammlung entsprechender Anforderungen für die Geschäftsprozessmodellierung sind die *Grundsätze ordnungsgemäßer Modellierung* (GoM) [RSD03]:

*Grundsätze ordnungsgemäßer Modellierung*

**Grundsatz der Richtigkeit** Verlangt sowohl die semantische (korrekte Wiedergabe des zu modellierenden Sachverhaltes) wie auch eine syntaktische (Einhaltung von Notationsregeln) Richtigkeit des Modells

**Grundsatz der Relevanz** Alle für die jeweilige Sicht relevanten Sachverhalte müssen im Modell dokumentiert sein – irrelevante Informationen (d.h. solche, welche ohne Verlust an Wert für den Adressaten aus dem Modell entfernt werden können) sollten nicht enthalten sein

**Grundsatz der Wirtschaftlichkeit** Die Kosten einer Modellierung sollten in einem



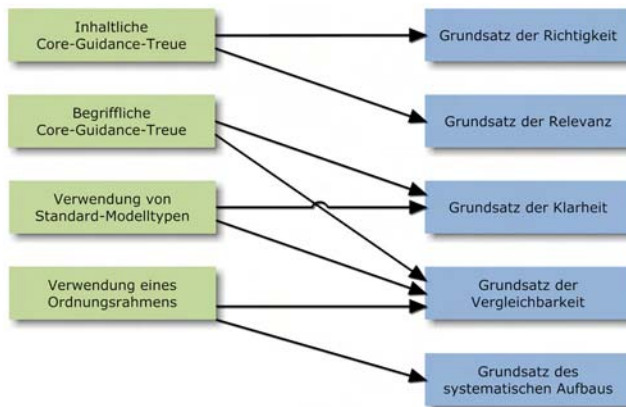


Abbildung 5.2: Richtlinien für ITIL-Referenzmodelle und ihr Einfluss auf die Grundsätze ordnungsgemäßer Modellierung

angemessenen Verhältnis zu ihrem Nutzen stehen<sup>1</sup>

**Grundsatz der Klarheit** Das Modell sollte vom Adressaten möglichst intuitiv begreifbar sein, d.h. so weit wie möglich nur die bei einem typischen Adressaten bereits vorhandenen Methodenkenntnisse voraussetzen

**Grundsatz der Vergleichbarkeit** In größeren Modellierungsprojekten sollte durch Definition und Durchsetzung von Modellierungsempfehlungen (z.B. Festlegung der zu verwendenden Modelltypen, Layoutkonventionen) eine Vergleichbarkeit unabhängig voneinander erstellter Modelle gewährleistet werden

**Grundsatz des systematischen Aufbaus** Die Schnittstellen zwischen Modellen, die einen bestimmten Sachverhalt aus verschiedenen Sichten beleuchten, sollten wohldefiniert und dokumentiert sein

Für die Modellierung der ITIL-Prozesse lassen sich konkretere Richtlinien bestimmen, die erleichtern, dass die erstellten Referenzmodelle diesen Grundsätzen entsprechen (der Grundsatz der Wirtschaftlichkeit bleibt dabei unberücksichtigt). Die im folgenden vorgestellten vier Richtlinien sind, zusammen mit den Grundsätzen der Modellierung, die sie primär adressieren, in Abbildung 5.2 dargestellt.

*Richtlinien für Referenzmodelle von ITIL-Prozessen*

**Inhaltliche Core-Guidance-Treue** Im Sinne des Grundsätze der Richtigkeit und der Relevanz sollte das Referenzmodell eines ITIL-Prozesses so eng wie

<sup>1</sup>Für eine effektive praktische Umsetzung dieses Grundsatzes fehlen aber bisher entsprechende Methoden der Kosten- und Leistungsrechnung [Pro03]

möglich am Inhalt der *ITIL Core Guidance* (vgl. Abschnitt 3.1.1) orientiert sein – das zu erstellende Modell sollte also „werkgetreu“ sein. Das bedeutet, das Modell sollte keine unnötigen Abänderungen, Vereinfachungen oder Ergänzungen der Prozesse, so wie sie in den Original-Büchern beschrieben sind, enthalten.

**Begriffliche Core-Guidance-Treue** Über die inhaltliche Treue zur Core-Guidance hinaus sollten auch die Begriffe der ITIL so weit wie möglich unverändert Eingang in das Modell finden. Dies dient sowohl dem Grundsatz der Klarheit (Adressaten mit ITIL-Kenntnissen wird das Verständnis erleichtert) wie auch dem der Vergleichbarkeit (z.B. wenn unabhängig voneinander entstandene Modelle des gleichen Prozesses verglichen werden). Unter dem Aspekt der Vergleichbarkeit ist ebenfalls anzumerken, dass ohne eine enge Orientierung an den Original-Begrifflichkeiten auch eine inhaltliche Core-Guidance-Treue nur schwer nachweis- oder widerlegbar ist.

**Verwendung von Standard-Modelltypen** Den gleichen Grundsätzen kann zusätzlich durch die Verwendung von Standard-Modelltypen gedient werden. Da die Modelle der ITIL-Prozesse als allgemeine Referenz verwendbar sein sollen, sind der Adressatenkreis und dessen Methodenkenntnisse nicht, wie z.B. in einem unternehmensinternen Entwicklungsprojekt, von vornherein bekannt. Die Verwendung standardisierter und verbreiteter Modellierungssprachen und -notationen erleichtert das Verständnis der Modelle für einen größeren Adressatenkreis. Im gleichen Sinne erhöht sie auch die Chance, dass unabhängig voneinander erstellte Modelle, ohne die komplexe und selten verlustfrei mögliche Abbildung zwischen Modellierungssprachen, miteinander verglichen werden können.

**Verwendung eines Ordnungsrahmens** Die Verwendung eines Rahmens zur Ordnung der Modelle dient in unmittelbarer Weise dem Grundsatz des systematischen Aufbaus. Zugleich hat er aber auch, indem er die Zuordnung äquivalenter Teilmodelle zueinander erleichtert, einen positiven Einfluss auf die Vergleichbarkeit von Gesamtmodellen.

Die beiden letztgenannten Richtlinien sind auf die Erstellung des Gesamtmodells (das neben den Prozessmodellen auch Teilmodelle zum Systementwurf beinhaltet) anwendbar, die beiden erstgenannten beziehen sich hingegen speziell auf die Erstellung der Prozessmodelle (also den unten in Abschnitt 5.4.1 festgelegten *Owner View*).

### 5.4 Methode

Wie kann nun ein solcher modellbasierter Ansatz umgesetzt werden? Die grundlegende Methode, die im folgenden hierfür vorgeschlagen wird, basiert auf der

Definition eines Ordnungsrahmens für die zu entwickelnden Modelle (nächster Abschnitt 5.4.1) und einem an diesen Rahmen angepassten Vorgehen bei der Modellerstellung (Abschnitt 5.4.2).

### 5.4.1 Ordnungsrahmen

Bei der Erstellung eines Modells muss immer eine Abwägung getroffen werden zwischen der Detaillierung, die erforderlich ist damit das Modell möglichst alle für seine Adressaten wesentlichen Informationen enthält, und der mit einem hohen Detaillierungsgrad einhergehenden Komplexität des Modells, welche das Verständnis erschweren kann. Aufgrund des Umfangs und der Komplexität der Aufgabenstellung bei der Entwicklung eines ITIL-Tools ist es schwerlich möglich, hierbei einen geeigneten Kompromiss für ein einzelnes Modell zu finden – man verletzt entweder den Grundsatz der Relevanz oder den Grundsatz der Klarheit. Eine modellbasierte Ansatz für ITIL-Tools wird vielmehr mit der Erstellung einer ganzen Reihe von Teilmodellen einhergehen müssen, welche die Sachverhalte und Designs unter jeweils verschiedenen Blickwinkeln wiedergeben. Bei der Erstellung einer großen Menge von Teilmodellen besteht aber die Gefahr, dass der Überblick verloren geht und die Teilmodelle unnötig redundant ausfallen oder aber in ihrer Gesamtheit nicht alle wichtigen Aspekte abdecken. Es ist also ein Ordnungsrahmen vonnöten, der es erlaubt das Gesamtmodell dergestalt in Teilmodelle zu zerlegen, dass diese weitgehend redundanzfrei und in ihrer Einordnung in das Gesamtmodell nachvollziehbar bleiben. Damit kann auch dem Grundsatz des systematischen Aufbaus und dem Grundsatz der Vergleichbarkeit gedient werden.

*Warum ein Ordnungsrahmen?*

#### Bestehende Ansätze zur Ordnung von Modellen

Für Aufteilungen in Teilmodelle existieren in der Informatik mehrere Ansätze, die im Sinne eines „separation of concerns“ (vgl. [Dij82]) versuchen die Komplexität von Informationssystemen durch Verwendung verschiedener Modellsichten besser beherrschbar zu machen. Auch in der Modellierung von Geschäftsprozessen wird auf dieses Konzept zurückgegriffen und ein Prozess durch verschiedene Teilmodelle beschrieben, bei denen jedes eine Beschreibungssicht bzw. einen Abstraktionsaspekt des Prozesses widerspiegelt (vgl. [RSD03]). Dementsprechend gibt es eine Anzahl von Ansätzen zur Abgrenzung der verschiedenen Sichten auf ein System oder einen Prozess.

*Separation of Concerns*

#### Ansätze zur Beschreibung verteilter Systeme und Netzmanagement-Architekturen

Einige ältere Standards aus der Informatik enthalten Konzepte zur Betrachtung komplexer Systems unter verschiedenen Blickwinkeln, welche bis heute Ein-

*RM-ODP und OSI-Management-Teilmodelle* fluss ausüben, auch wenn die Standards in ihrer Gesamtheit kaum noch Anwendung finden. Hierzu gehört zum Beispiel das *Reference Model for Open Distributed Processing* (RM-ODP) [ITU95][Put00], dessen Definition von *Viewpoints* im Grundsatz häufig aufgegriffen wird. Auch die Definition der vier Teilmodelle im *OSI Management Framework* [ITU92] leistet bis heute zur Einordnung von Management-Architekturen gute Dienste (vgl. z.B. [HAN99]).

*Viewpoints in Anlehnung an RM-ODP* **MDA Viewpoints** Im Bereich der Softwareentwicklung ist in Bezug auf eine modellbasierte Entwicklung die *Model Driven Architecture* (MDA) [OMG03] der OMG einer der prominentesten Ansätze. MDA übernimmt von RM-ODP das Konzept der Viewpoints und definiert einen *Computation Independent Viewpoint*, einen *Platform Independent Viewpoint* und einen *Platform Specific Viewpoint*. Diese Information unter diesen Blickwinkeln wird entsprechend in einem *Computation Independent Model* (CIM), einem *Platform Independent Model* (PIM) und einem *Platform Specific Model* (PSM) festgehalten.

*CIM, PIM und PSM* Dabei entspricht das CIM dem so genannten *Domain Model* in der Software-Entwicklung, aus dem die Anforderungen an das zu entwickelnde System abgeleitet werden können. Im Kontext dieser Arbeit könnte es also dem Modell des zu unterstützenden Prozesses entsprechen. Das PIM ist hingegen ein Modell des Systems, allerdings unter einem Blickwinkel der plattformunabhängig, wenn auch nicht notwendigerweise unabhängig vom generellen Typ der Plattform ist. Ein PIM kann also einen komponentenorientierten Systemaufbau beschreiben, abstrahiert aber von Plattformdetails wie zum Beispiel, ob die Komponenten als Enterprise JavaBeans oder .NET-Komponenten realisiert werden sollen – dies ist erst für das PSM relevant. Eines der wesentlichen Ziele der MDA ist das Erreichen einer automatisierten Modelltransformation von einem PIM in (verschiedene) PSM, so dass ein Systemdesign weitgehend automatisiert auf verschiedene Plattformen (Technologien) übertragen werden kann.

*Die Sichten des ARIS-Hauses* **ARIS-Haus** Andere Ansätze, die sowohl Modelle der Geschäftsprozesse als auch Modelle der diese Prozesse unterstützenden IT-Systeme in einen gemeinsamen Rahmen ordnen möchten, unterscheiden explizit zwei Arten des „separation of concerns“, also verschiedene Arten der Sichtenbildung. So teilt zum Beispiel die im deutschsprachigen Raum verbreitete Prozessmodellierungsmethode des ARIS-Rahmenwerks Modelle zunächst in vier *Beschreibungssichten* auf: Die *Organisationssicht*, die *Datensicht*, die *Funktionssicht* und schließlich die *Steuerungssicht*. Diese Zerlegung wird im so genannten ARIS-Haus aber auf drei *Beschreibungsebenen* angewendet, dem Fachkonzept, dem DV-Konzept und der Implementierung [Sch97] (siehe Abbildung 5.3 auf der nächsten Seite), wobei man das ARIS-Fachkonzept und MDA-CIM sowie ARIS-DV-Konzept und MDA-PIM als

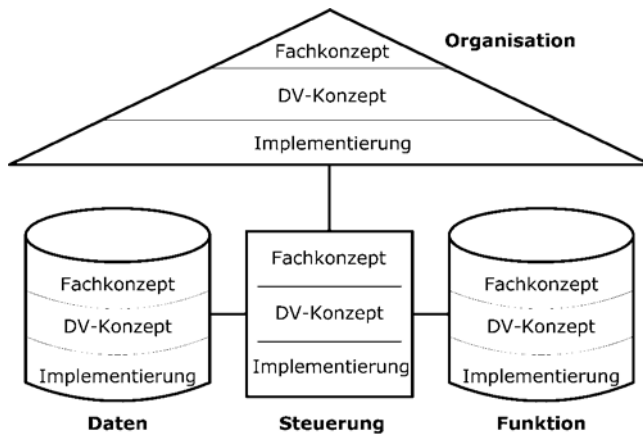


Abbildung 5.3: ARIS-Konzept: Das „ARIS-Haus“ [Sch97]

in etwa äquivalent auffassen kann<sup>2</sup>. Durch diese zweifache Unterteilung existieren also insgesamt zwölf Modellierungssichten (wenn man die Implementierungsebene mitrechnet), die zusammen das so genannte ARIS-Haus bilden (siehe Abbildung 5.3).

**Zachman Framework** Noch etwas allgemeiner und umfassender ist das *Zachman Framework*. Von diesem existieren zwei Varianten: die 1987 vorgestellten ursprünglichen Fassung [Zac87] und eine erweiterten Version von 1992 [ZS92]. Die erweiterte Variante ändert aber nichts am grundlegenden Aufbau, sondern fügt nur neue neue Abstraktionstypen (s.u.) hinzu. Die stellt somit mehr eine Ergänzung als eine fundamentale Verbesserung des ursprünglichen Frameworks dar (siehe auch [Har03]).

*Zwei Varianten*

Das *Zachman Framework* ordnet Modelle<sup>3</sup> in einer Matrix an – horizontal gegliedert in Perspektiven bzw. Blickwinkel (*perspectives, views*) und vertikal unterteilt

*Perspektiven und Abstraktionstypen*

<sup>2</sup>Die Implementierung wird in [Sch97] nur am Rande behandelt, daher lässt sich an dieser Stelle nicht exakt sagen, inwieweit sie dem PSM in MDA entspricht – vergleicht man aber ARIS und MDA mit dem *Zachman Framework* (siehe nächster Abschnitt), so scheint die Implementierungs-Ebene aus ARIS eher deren *Subcontractor Perspektive* zu entsprechen, während PSM der *Builder Perspektive* zuzuordnen ist.

<sup>3</sup>Bei Zachman „architectural representations“ genannt. Der Begriff „Model“ wird im *Zachman Framework* verwendet für eine Sammlung von „architectural representations“, die einen gemeinsamen Adressaten haben – also alle verwendeten „architectural representations“ einer Ebene im Framework. Da eine „architectural representations“, außer auf der untersten Ebene, aber letztlich kaum etwas anderes als ein Modell ist, wird dieser Begriff hier nicht weiter verwendet.

	Data	Function	Network	People	Time	Motivation
<b>Planner View</b> <i>Scope</i>	list of things important to the business	list of processes the business performs	list of Locations in which the business operates	list of organizations important to the business	list of events significant to the business	list of business goals / strategies
<b>Owner View</b> <i>Business Model</i>	e.g. semantic model	e.g. process flow diagram	e.g. logistics network	e.g. organization chart	e.g. master schedule	e.g. business plan
<b>Designer View</b> <i>System Model</i>	e.g. data model	e.g. data flow diagram	e.g. distributed system architecture	e.g. human interface architecture	e.g. processing structure	e.g. knowledge architecture
<b>Builder View</b> <i>Technology Model</i>	e.g. data design	e.g. structure chart	e.g. system architecture	e.g. human / technology interface	e.g. control structure	e.g. knowledge design
<b>Subcontractor View</b> <i>Detailed Representations (out-of-context)</i>	e.g. data definition description	e.g. program	e.g. network architecture	e.g. security architecture	e.g. timing definition	e.g. knowledge definition

Abbildung 5.4: Zachman Framework nach [ZS92]

in Beschreibungsarten bzw. Abstraktionstypen (*types of description, types of abstraction*).

**Perspektiven** Es werden fünf Perspektiven festgelegt: Der Blickwinkel des *Planner* ist auf den allgemeinen Kontext des Gesamtunternehmens gerichtet, während der des *Owner* ein konzeptueller Blick auf die Geschäftsvorgänge ist – beide sind IT-unabhängig. Der *Owner View* entspricht im großen und ganzen dem *Computation Independent Viewpoint* aus MDA. Die anderen drei Perspektiven *Designer*, *Builder* und *Subcontractor* betreffen IT-Systeme, wobei die *Designer View* als äquivalent zum *Platform Independent Viewpoint* in MDA und die *Builder View* als dem *Platform Specific Viewpoint* entsprechend gelten kann (siehe auch Vergleich von *Zachman Framework* und MDA in [FHM<sup>+</sup>03]).

**Typen von Abstraktion** Orthogonal zu diesen Perspektiven wurden im ursprünglichen Framework [Zac87] drei Abstraktionstypen definiert – *Data*, *Function* und *Network* – die 1992 um drei weitere – *People*, *Time* und *Motivation* – erweitert wurden. Somit ergibt sich eine Matrix mit 30 Teilmodellen<sup>4</sup> welche in Abbildung 5.4, zusammen mit Beispielen möglicher Modellinhalte oder -typen, dargestellt ist.

Das *Zachman Framework* ist dabei ein reiner Ordnungsrahmen – es macht kei-

<sup>4</sup>in der Nomenklatur von Zachman fünf Modelle mit je sechs „architectural representations“ – die Bezeichnungen der Modelle bei Zachman können Abbildung 5.4 entnommen werden

nerlei Vorgaben oder Annahmen über die für die Teilmodelle verwendeten Modellierungssprachen oder -notationen. **Zachman** weist selbst darauf hin, dass sich bislang nicht für jede seiner Zellen (insbesondere des erweiterten Frameworks) eine geeignete Modellierungsmethode etabliert hat – für andere Zellen gibt es wiederum mehrere etablierte Modelltypen, die sich in sie einordnen ließen [Zac03]. Dass es daher kaum möglich ist in einer praktischen Anwendung alle Zellen des Frameworks sinnvoll zu füllen, hat wohl auch zur relativ geringen Verbreitung im praktischen Einsatz beigetragen.

keine  
vorgegebenen  
Modellierungs-  
methoden

### Entwicklung eines Ordnungsrahmens

Von den vorgestellten Ordnungsrahmen ist der des *Zachman Framework* also der universellste und bietet daher eine gute Basis für die Festlegung eines Ordnungsrahmens für die Entwicklung prozessorientierter ITSM-Betriebswerkzeuge. Das *Zachman Framework* steckt allerdings auch einen sehr weiten Bereich von zu modellierenden Aspekten ab, der nicht in seiner Gänze für das hier verfolgte Vorhaben unbedingt notwendig ist. Daher ist es nicht effizient, für eine Beschreibung der ITIL und eines unterstützenden Systems Modelle für alle Zellen zu entwickeln.

Ausgangspunkt  
ist das *Zachman  
Framework*

Die ITIL beschreibt bereits Dienstmanagementprozesse, bewegt sich also inhaltlich ungefähr auf der Ebene des *Owner View*. Die zusätzliche Beschreibung eines *Planner View*, also der allgemeinen, für die Unternehmensführung relevanten Rahmenbedingungen würde kaum einen zusätzlichen Erkenntnisgewinn erbringen. Ein *Planner View*, der für alle oder nahezu alle IT-Service-Provider gültig ist, könnte höchstens die zur Klassifikation von ITIL-Prozessen vorgenommenen Annahmen wiedergeben (vgl. Abschnitt 4.3.3).

Verzicht auf  
*Planner View*

Auch auf den *Subcontractor View* kann verzichtet werden. Er erlaubt keine kohärente Sicht des Gesamtsystems mehr (er enthält die „out-of-context representations“ [Zac87]) – die eigentlich interessanten Designentscheidungen sind daher schon in den darüber liegenden Sichten getroffen. Zudem stellen die „architectural representations“ des *Subcontractor View* in der Regel keine eigentlichen Modelle mehr dar (außer in dem Sinn, wie der Quellcode eines Programms ein Modell des Ablaufverhaltens ist).

Verzicht auf  
*Subcontractor  
View*

Auch eine Berücksichtigung aller Abstraktionstypen ist nicht unbedingt sinnvoll. Für die betrachteten Dienstmanagementprozesse definiert die ITIL keine „business events“, im *Owner View* kann also auf den Abstraktionstyp *Time* verzichtet werden. Aber auch im *Designer View* und *Builder View* stehen Zeitaspekte selten für sich alleine, sondern sind meist eng mit den Funktionen des Systems verknüpft. Sie lassen sich daher in einem Modell zum Abstraktionstyp *Function* integrieren und rechtfertigen in der Regel nicht die Erstellung eines für sich alleine genommen wenig aussagekräftigen Modells.

Verzicht auf  
*Abstraktionstyp  
Time*

	Data	Function	Network	People
Owner View	ITIL-Dokumente und Entitäten	Workflow ITIL-Prozess	Verteilung organisatorischer Einheiten	Rollen im ITIL-Prozess
Designer View	Datenmodell des ITSM-Systems	Nutzfälle des ITSM-Systems	Aufteilung auf Komponenten bzw. Services	Rechte- bzw. Zugriffskonzept
Builder View	Datenspezifikation in XML	Services-Workflow in BPEL	Web-Services-Schnittstellen	User-Interface-Navigation

Abbildung 5.5: Ordnungsrahmen für modellbasierten Ansatz (beispielhafte Modellinhalte)

*Verzicht auf Abstraktionstyp Motivation*

Auch auf *Motivation* kann verzichtet werden. Da die ITIL kaum Aussagen trifft, warum ihre Richtlinien nun genau so und nicht anders gestaltet sind, kann die Motivation für den *Builder View* eh kaum geliefert werden. Beim Design der Modelle im *Designer View* und *Builder View* scheint es für einen ersten Entwurf eines solchen Systems pragmatischer, die allgemeinen Designziele in der Beschreibung der Modelle darzulegen, als eine formale Darstellung in einem Modell anzustreben – die Motivation für ein Systemdesign formal zu fassen und zu modellieren ist keine triviale Aufgabe (vgl. z.B. die Forschung zu *Design Rationale* [DMMP06]).

*Resultierender Ordnungsrahmen*

Im Folgenden soll daher ein Ordnungsrahmen zum Einsatz kommen, der auf einem Ausschnitt des *Zachman Framework* basiert. Klammert man die oben genannten Perspektiven und Abstraktionstypen aus, so bildet sich ein Ausschnitt, der einem um den Abstraktionstyp *People* erweiterten, aber auf die drei den MDA-Viewpoints entsprechenden Perspektiven (also auf *Owner View*, *Designer View* und *Builder View*) beschränkten ursprünglichen *Zachman Framework* entspricht.

*Network im Owner View für ITIL ausgeklammert*

Der so entstehende Ordnungsrahmen mit zwölf Zellen ist in Abbildung 5.5 dargestellt. Es sei noch zu beachten, dass, aufgrund der Allgemeinheit der ITIL und ihrer weitgehenden Unabhängigkeit von organisatorischen Strukturen, kein Modell für den Abstraktionstyp *Network* im *Owner View* erstellt wird. Über die Verteilung der Prozessbeteiligten und Prozessressourcen auf verschiedene funktionale Organisationseinheiten oder gar Standorte lassen sich keine sinnvollen, allgemeingültigen Aussagen treffen, die ausreichen um ein entsprechendes Modell zu entwickeln.



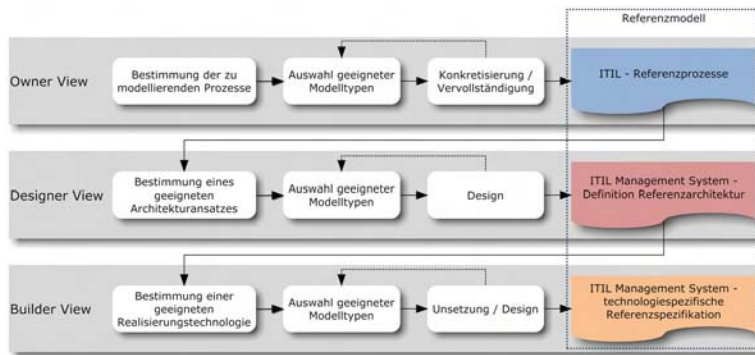


Abbildung 5.6: Vorgehen bei der Modellerstellung

### 5.4.2 Vorgehen

Das Vorgehen bei der Erstellung der Modelle ergibt sich weitgehend aus dem gewählten Ordnungsrahmen und wird in Abbildung 5.6 illustriert. Wie bei der modellbasierten Entwicklung üblich, wird ebenenweise von oben nach unten vorgegangen. Die fertigen Modelle einer Ebene bilden die Basis für die Anforderungen an die Ausprägung der nächst niedrigeren Ebene. Als erstes werden also die Modelle des *Owner View* (das Domänenmodell), in diesem Fall also die Prozessmodelle der ausgewählten ITIL-Prozesse erstellt. Da es sich hierbei primär um die Darstellung bestehender Prozessrichtlinien handelt, kann man sagen, dass diese Modelle in gewisser Weise „Istmodelle“ darstellen, während die für *Designer View* und *Builder View* entwickelten Modelle als „Sollmodelle“ angesehen werden können – allerdings ist der Unterschied aufgrund des hohen Aufwandes, der bei der Konkretisierung und Vervollständigung der ITIL-Richtlinien erbracht werden muss, nicht so klar wie in klassischen Modellierungsprojekten.

*Vorgehen startet im Owner View*

Auf jeder Ebene muss zunächst entschieden werden, was den Rahmen der Modellierung auf dieser Ebene bildet. Auf der Ebene *Owner View* ist also beispielsweise zu entscheiden, welche Prozesse überhaupt im Modell berücksichtigt werden sollen. Auf den unteren Ebenen muss eine ähnliche Grundsatzentscheidung getroffen werden, also beispielsweise ein zentraler oder verteilter Systemaufbau bzw. die Wahl einer Umsetzungsplattform oder -technologie.

*Vorgehen in jeder Ebene ähnlich*

Diese grundlegende Entscheidung beeinflusst auch, welche Modelltypen für die Modellierung überhaupt geeignet sind. Da der oben definierte Ordnungsrahmen zwölf Zellen des *Zachman Framework* umfasst, können hier im Prinzip viele unterschiedliche Notationen zum Einsatz kommen. Im Sinne des Grundsatzes der Klarheit ist es allerdings wünschenswert, dass für das Verstehen des Gesamt-

*Auswahl des Modelltyps und eigentliche Modellierung*

modells kein übermäßig breites Wissen verschiedenster Modellierungssprachen erforderlich ist. So bietet es sich an, zumindest wo immer sinnvoll, hierbei auf die UML zurückzugreifen, welche für die meisten der definierten Zellen einen adäquaten Diagrammtyp anbieten kann [FHM<sup>+</sup>03]. Schließlich müssen die jeweiligen Modelle erstellt, also z.B. die ITIL-Prozesse konkretisiert, der Systemaufbau entworfen werden.

### *Anpassung des Ansatzes für weitere ITIL-Prozesse*

Der Ordnungsrahmen und das Vorgehen, wie sie hier für die in Abschnitt 5.2 festgelegten Prozesse entwickelt wurden, können prinzipiell auch für die anderen Prozesse der ITIL (aus Quadrant I und Quadrant III) angewendet werden. Da es sich um andere Arten von Prozessen handelt, die konsequenterweise auch andere Arten von Werkzeugen erfordern, werden allerdings die Schwerpunkte bei der Modellierung anders liegen. Workflows spielen eine untergeordnete Rolle, weswegen die Modelltypen und Modellumfänge unter dem Abstraktionstyp *Function* sicher andere sein werden. Auf der anderen Seite kann durchaus überlegt werden, ob nicht beispielsweise für *Service Level Management* und *Release Management* nicht doch der Abstraktionstyp *Time* ( zur Erfassung von Review- bzw. Release-Zyklen) dem Ordnungsrahmen hinzugefügt werden sollte.

Eine genauere Untersuchung muss aber einer späteren Arbeit vorbehalten bleiben – im nächsten Kapitel wird zunächst nur für die drei Prozesse des Quadranten IV eine beispielhafte Umsetzung des vorgestellten Ansatzes vorgenommen.

## 6 Umsetzung

Nachfolgend werden die Erfahrungen und Erkenntnisse bei der prototypischen Umsetzung des im vorherigen Kapitel vorgestellten Ansatzes beleuchtet. Die erstellten Modelle sind in Anhang A (ab Seite 163) gesammelt. Den Schwerpunkt des nachstehenden Kapitels bildet die Diskussion des *Owner View* in Abschnitt 6.1 – für den *Owner View* wurden Modelle der in den Quadranten IV fallenden ITIL-Prozesse (*Incident Management*, *Problem Management* und *Change Management* – vgl. Abschnitte 4.3.6 und 5.2) erstellt. Auf diesen baut eine beispielhafte Systemarchitektur für ein *Incident Management System* auf, dessen Darstellung in entsprechenden Modellen des *Designer View* in Abschnitt 6.2 erläutert wird. Nach der Darlegung von Überlegungen zu einer konkreten, plattform-spezifischen Umsetzung auf Basis von für den *Builder View* entwickelten Modellen (Abschnitt 6.3) schließt eine Zusammenfassung (Abschnitt 6.4) das Kapitel ab.

### 6.1 Owner View

Wie man schon an den vielfältigen und sich immer wieder unterscheidenden Darstellungen der ITIL-Prozesse in der Sekundärliteratur erahnen kann, ist eine präzise Modellierung der Dienstmanagementprozesse der ITIL nicht ohne weiteres in direkter Weise möglich. Die Herausforderung bei der Erstellung der Modelle des *Owner View* ist daher, die ITIL-Richtlinien hinreichend zu konkretisieren und in nachvollziehbare Modelle zu fassen, die als fachliche Basis für eine Systementwicklung dienen können, dabei aber nicht zu weit von eigentlichen ITIL-Inhalten abzuweichen („Core-Guidance-Treue“, vgl. Abschnitt 5.3). Aufgrund der geringen Strukturierung, Unübersichtlichkeit und nicht konsequenten Begriffsverwendung der ITIL-Richtlinien in *Service Support* ist die Herausarbeitung dieser Inhalte mit einem erheblichen Aufwand verbunden.

Abschnitt 6.1.1 wiederholt und präzisiert noch einmal die Auswahl der Prozesse und die Festlegung der allgemeinen Zielsetzung für die unter dem *Owner View* (*Business Model*) zu erstellenden Modelle. Der darauf folgende Abschnitt 6.1.2 diskutiert an einem Beispiel die praktischen Schwierigkeiten bei der Modellerstellung. Die erstellten Modelle sind, wie bereits oben erwähnt, in Anhang A gesammelt. Die verbleibenden Abschnitte dieses Kapitels haben nicht das Ziel, alle Modelle detailliert zu erläutern, sie konzentrieren sich vielmehr auf die Be-

leuchtung der methodischen Herausforderungen bei der Modellierung sowie die Diskussion ausgewählter, bemerkenswerter Modellinhalte. Wie bereits in Abschnitt 5.4.1 erwähnt ist für die allgemeinen ITIL-Richtlinien eine Erstellung eines Modells unter dem Abstraktionstyp *People* nicht sinnvoll. Die Diskussion beschränkt sich daher auf das *Data Model* (Abschnitt 6.1.4), das *Function Model* (Abschnitt 6.1.5) und das *People Model* (Abschnitt 6.1.6).

### 6.1.1 Abgrenzung und Zielsetzung

*Incident, Problem und Change Management*

Eine Werkzeugunterstützung erscheint für das *Incident Management* am lohnendsten, die Prozesse *Problem Management* und *Change Management* haben aber ähnliche grundlegende Eigenschaften (vgl. Abschnitt 4.3.6). Zudem wird der Ablauf des *Incident Management* auch vom Abschluss von Aktivitäten in diesen beiden Prozessen beeinflusst, weswegen sie im folgenden, wenn auch in einzelnen Aspekten weniger detailliert als das *Incident Management*, ebenfalls betrachtet werden. Die inhaltliche Abgrenzung entspricht also der bereits in Abschnitt 5.2 getroffenen für die Anwendungsdomäne des modellbasierten Ansatzes (vgl. auch Abbildung 5.1 auf Seite 130).

*keine Modellierung von KPIs*

Ebenso der Abgrenzung der Anwendungsdomäne entsprechend werden die, wenn auch nicht zentralen, so doch vorhandenen Richtlinien der ITIL zur Prozessplanung, Prozesseinführung und Prozessanalyse, nicht mitmodelliert – entsprechende Aktivitäten (z.B. *Planning the Implementation of Operational Processes* aus dem *Change Management*) oder Informationsobjekte (z.B. KPIs) bleiben also zunächst unberücksichtigt.

*Modellierungsziele*

Die übergeordneten Ziele bei der Erstellung der Modelle des *Owner View* sind bereits in Abschnitt 5.3 festgelegt worden: Inhaltliche und begriffliche Treue der Modelle zur *ITIL Core Guidance*. Eine Herausforderung ist es dabei einen geeigneten Kompromiss zwischen Core-Guidance-Treue, Widerspruchsfreiheit, Verständlichkeit und Detaillierung zu finden. Warum dies notwendig ist, wird im nächsten Abschnitt erläutert.

### 6.1.2 Erfahrungen bei der Erstellung von ITIL-Modellen

*unstrukturierte Darstellung auch von Kerninhalten*

Die ITIL ist in vielen Teilaspekten lückenhaft und widersprüchlich – wo die Lücken und Widersprüche liegen, kann im Detail aber oft nur in mühsamer Kleinarbeit herausgearbeitet werden. Dies ist letztlich auf die bereits mehrfach angesprochene, nicht-formale und schlecht strukturierte Darstellung zurückzuführen. Die Information zu einem bestimmten Aspekt (z.B. „Welche Rollen gibt es in diesem Prozess?“) ist häufig über viele Abschnitte oder sogar Kapitel verteilt – dabei versteckt sich ein wichtiges Stück Information oft als einzelner Stichpunkt in einer Aufzählung von Banalitäten oder in einer Klammer, die hinter einem selbst

eigentlich wenig aussagekräftigen Satz gesetzt ist. Manche vermeintliche Lücke in der Darstellung eines bestimmten Prozess-Aspektes kann durch Information geschlossen werden, die sich erst in thematisch völlig anders ausgerichteten Abschnitten oder sogar in anderen Kapiteln findet (z.B. zu *Change Management* in Annex A des *Configuration Management* gewidmeten Kapitels 7 in *Service Support* [OGC00]). Ebenso können manche Widersprüche für einen Aspekt aufgelöst werden, sobald man die ITIL in der Breite unter diesem untersucht – andere Widersprüche aber zeigen sich erst dann. Da dies auch Kernkonzepte und -begriffe betrifft, die zudem kaum als solche gekennzeichnet werden, garantiert eine gründliche Analyse der ITIL-Inhalte nicht immer ein detaillierteres Modell, sondern oftmals nur ein begrifflich oder inhaltlich genaueres.

Um dies an einem kleinen Ausschnitt zu verdeutlichen, sei im folgenden kurz die Modellierung der beiden zentralen Konzepte *Problem* und *Known Error* aus dem *Problem Management* diskutiert. Für beide Begriffe, wenn auch nicht für die entsprechenden *Records*, finden sich im Glossar der ITIL Einträge:

- „Known Error: an Incident or Problem for which the root cause is known and for which a temporary Work-around or a permanent alternative has been identified“
- „Problem: unknown underlying cause of one or more Incidents“

*Begriffsdefinitionen im Problem Management*

Diese Definitionen unreflektiert in das Modell zu übertragen, bringt aber gewisse Widersprüche mit sich, wenn man zusätzlich den Workflow des *Problem Management* betrachtet. Der grundlegende Prozessablauf im *Problem Management* ist so, dass, sobald ein neues *Problem* erkannt ist, beispielsweise wenn sich die Symptomatik eines gemeldeten *Incident* weder mit einem dokumentierten *Problem* noch einem *Known Error* Einklang bringen lässt (vgl. Abbildung 6.2 in [OGC00]), der Teilprozess *Problem Control* angestoßen wird. Nach der Dokumentation des *Problem* und der Klassifikation wird schließlich (abhängig davon, welche Priorität dem *Problem* beigemessen wird) früher oder später damit begonnen zu untersuchen, welche *Root Cause* den Symptomen zugrunde liegt. Das Finden eines geeigneten Work-around ist noch Teil des *Problem Control*. Die Beseitigung der *Root Cause* wird erst im Rahmen des Teilprozesses *Error Control*, meistens durch Stellen eines RFC, angegangen.

*Prozessablauf Problem Management*

Was ist aber, wenn in *Problem Control* zwar die *Root Cause* gefunden wird, aber kein Work-around? Offensichtlich ist dies möglich. Ist die *Root Cause* ein Fehler an einer für einen bestimmten Dienst zentralen, nicht ohne weiteres umgeh- oder ersetzbaren Komponente, ist unter Umständen ein Work-around im eigentlichen Sinne nicht möglich. Nach der obigen Definitionen kann das *Problem* dann aber nicht zu einem *Known Error* werden und somit würde nie der Teilprozess *Error Control* gestartet. Dies macht natürlich keinen Sinn, das *Problem* bliebe dann ja ungelöst. Nicht diese, aber eine ähnlich gelagerte Problematik wird tatsächlich in Abschnitt 6.6.3 in [OGC00] diskutiert, wo nahe gelegt wird, in gewissen Fällen

*Begriffe schwer mit sinnvollem Ablauf in Einklang zu bringen*

einen leeren *CI Record* zu erstellen, damit ein *Problem* den formalen Status eines *Known Error* erreichen kann (die Zuordnung der *Root Cause* zu einem *Configuration Item* ist also anscheinend eine Voraussetzung für einen *Known Error*).

*gibt es einen „unknown error“?*

Vielleicht ist für den Beginn des Teilprozesses *Error Control* ja aber auch die Existenz eines *Known Error* keine Voraussetzung und er beginnt schon früher, also unter Umständen, bevor die dazugehörige Instanz von *Problem Control* endgültig abgeschlossen ist. Immerhin heißt es in der Beschreibung zu *Error Control* in Abschnitt 6.7.1 von [OGC00]: „An error is identified when a faulty CI (a CI that causes, or may be likely to cause, Incidents) is detected. A Known error status is assigned when the root cause of a Problem is found and a Work-around has been identified.“. Sind also der Ablauf von *Error Control* und *Problem Control* in bestimmten Fällen nebenläufig und ein *Known Error* nur eine Spezialisierung eines „Error“? Dann müsste es wohl auch einen „Error Record“ geben – dieser Begriff taucht aber in der ITIL nicht auf. Im gleichen Abschnitt (6.7.1) heißt es schließlich auch „the Problem record forms the basis of the Known Error record (indeed, it really involves only a change of status)“ – danach wird aus einem *Problem* nun wieder doch direkt ein *Known Error*.

*Auflösen aller Widersprüche unpraktikabel*

Es ist leicht einzusehen, dass das Auflösen all dieser Widersprüchlichkeiten nur durch eine unnötig komplexe Gestaltung des Prozessablaufs und des Datenmodells erfolgen könnte. Will man den eigentlich sinnvollen sequentiellen Ablauf von *Problem Control* und *Error Control* beibehalten, kommt man kaum umhin, im Widerspruch zur ITIL die Existenz eines bekannten Work-around nicht als notwendigen Teil des Konzeptes *Known Error* zu modellieren. Auch dass ein *Incident*, wie in der Definition im Glossar suggeriert, ein *Known Error* werden (oder sein) kann, ist nicht notwendigerweise zweckmäßig: Es heißt, das *Problem Record* bilde die Basis für das *Known Error Record* – ein *Incident Record* ist diesem vielleicht ähnlich, dient aber doch einem etwas anders gelagerten Zweck und wird von einem anderen Prozess verwaltet. Generell drängt sich zudem die Frage auf, ob es Sinn macht *Problem Record* und *Known Error Record* überhaupt als zwei unterschiedliche Konzepte zu modellieren. Für diese, durchaus noch ergänzbaren Fragestellungen zu den beiden Entitäten *Problem* und *Known Error* müssen also bei der Modellierung Designentscheidungen getroffen werden<sup>1</sup>.

*Kompromisse bei der Modellierung notwendig*

Ähnliche Herausforderungen ergeben sich für zahlreiche andere Aspekte in den ITIL-Prozessen – eine detaillierte Diskussion würde erheblich mehr Raum einnehmen als die betrachteten ITIL-Richtlinien selbst und den Rahmen dieser Arbeit sprengen. An etlichen Stellen sind auf dem Weg von den Texten in der ITIL hin zu einer konsistenten Prozessspezifikation Details zu konkretisieren und aufeinander

<sup>1</sup>Im entwickelten Modell kann ein *Known Error* nur aus einem *Problem* entstehen. Obwohl dies dann fachlich nicht notwendig wäre, wird aber der *Known Error Record*, da der Begriff häufig und prominent in den ITIL-Richtlinien genannt ist, als ein neben dem *Problem Record* eigenständig existierender *Record* modelliert (vgl. Abbildung A.5) – auch wenn dies bei den Modellen des *Designer View* keinen Sinn mehr macht (vgl. Abbildung A.20).

abzustimmen, was unter der Vorgabe einer engen Orientierung an die originalen ITIL-Begriffe und -Konzepte in vielen Fällen schwierig, manchmal auch praktisch unmöglich ist. Es muss also immer ein Kompromiss gefunden werden zwischen einem vielleicht ITIL-getreuen, aber widersprüchlichen oder unmäßig komplexen Modell und einem Modell, das vielleicht widerspruchsfrei und kohärent ist, aber die ITIL-Inhalte kaum noch widerspiegelt.

### 6.1.3 Allgemeine Darstellung der Modelle

Wie in Abschnitt 5.3 argumentiert, sollte die Darstellung der Modelle, im Sinne einer besseren Klarheit und Vergleichbarkeit, mittels Standard-Modelltypen erfolgen. Da die neueren XML-basierten Prozessbeschreibungssprachen (vgl. Abschnitt 3.10) primär auf eine maschinelle Verarbeitung und weniger auf eine leichte Interpretierbarkeit durch Menschen ausgelegt sind, erscheinen sie weniger geeignet. Wünschenswert ist eine Modellierungsmethode, welche die Verwendung möglichst intuitiv verstehbarer grafischer Modellierungsnotationen erlaubt. Zur Modellierung des *Owner View* bieten sich dabei primär die im ARIS-Rahmenwerk verwendeten Modelltypen (EPKs, Entity-Relationship-Diagramme, Organigramme) und die Diagrammtypen der UML an. Im vorliegenden Fall wird der UML der Vorzug gegeben, da sie auch für die Modelle des *Designer View* sowie, wenn auch je nach gewählter Umsetzungstechnologie nur eingeschränkt, für die des *Builder View* verwendet werden kann und so die Methodenkenntnisse eingeschränkt werden können, die vom Leser (bzw. Adressaten der Modelle) gefordert werden müssen.

*Entscheidung für UML als Modellierungssprache*

Allgemein werden in den folgenden Modellen Begriffe und Konzepte aus der ITIL, wo immer möglich, direkt und ohne Namensänderung als Modellobjekte erfasst. Wie weiter unten dargelegt, ist aber für eine vollständige, verständliche und konsistente Darstellung bisweilen auch die Aufnahme von zusätzlichen Elementen, die in genau dieser Form in der ITIL nicht definiert oder beschrieben werden, in die Modelle notwendig. Diese Ergänzungen werden in den Modellen mit einer anderen Füllfarbe gekennzeichnet als jene Modellobjekte, die direkt aus der ITIL abgeleitet werden können. Abbildung 6.1 verdeutlicht dies: Der Begriff des *Service Level* ist in der ITIL weder beschrieben noch erklärt und wird daher in einer anderen Farbe dargestellt als das *Service Level Agreement*, welches in der ITIL definiert wird.

*Unterscheidung zwischen und ITIL-Begriffen und Ergänzungen*

### 6.1.4 Data Model

Das *Data Model* im *Owner View* versucht die grundlegenden Zusammenhänge zwischen den verschiedenen in der ITIL definierten, beschriebenen oder an wichtigen Stellen erwähnten Entitäten, Dokumenten und Prozessartefakten (*Work Items*) herauszuarbeiten. Dieses Datenmodell ist bewusst einfach gehalten und

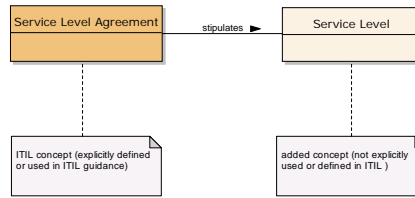


Abbildung 6.1: Farbliche Kennzeichnung direkt übernommener ITIL-Konzepte (vgl. Abbildung A.3)

kann, zusammen mit dem *People Model* (Abschnitt 6.1.6), auch als ein Fachbegriffsmodell (vgl. [RSD03]) der betrachteten ITIL-Prozesse verstanden werden, wobei allerdings rein begriffliche Beziehungen ohne inhaltlichen Bezug (wie „ $A$  ist synonym zu  $B$ “) nicht modelliert werden.

#### Darstellung als UML-Klassendiagramm

Als Modelltyp für die Darstellung der verschiedenen Entitäten und Dokumente, die in der ITIL für die drei betrachteten Prozesse definiert werden, fällt die Wahl auf UML-Klassendiagramme. Diese sind gut geeignet zur Darstellung der Beziehungen zwischen Entitäten auf Prozessebene (vgl. z.B. [EP98]) und erlauben dabei, vor allem dank der Möglichkeit Stereotypen zu definieren, für bestimmte Arten von Zusammenhängen eine prägnantere Wiedergabe als ER-Diagramme. Im Sinne einer einfachen, leicht erfassbaren und flexiblen Repräsentation werden die Entitäten und Prozessartefakte als Klassen ohne Attribute dargestellt.

#### Thing-Information Pattern

Die Darstellung in den Klassendiagrammen macht von einer Variation des *Thing-Information Pattern* aus [EP98] Gebrauch, d.h. es wird zwischen eigenständig existierenden Dingen einerseits und den sie beschreibenden bzw. repräsentierenden Dokumenten oder Datensätzen andererseits unterschieden. Dadurch ergeben sich unter den zu modellierenden ITIL-Entitäten mehrere grundlegende Typen, denen je ein eigener Stereotyp zugewiesen wird. Zunächst sind da die tatsächlichen „Dinge“, also Entitäten, welche, selbst wenn sie nicht notwendigerweise physisch greifbar sind, unabhängig von ihrer Dokumentation (im Rahmen eines Prozesses) existieren. Beispielsweise existiert ein *Problem* auch dann, wenn es noch nicht in Form eines *Problem Record* erfasst worden ist. Diese Entitäten werden im Modell mit dem Stereotyp `«thing»` versehen.

#### „Records“, „Documents“ und „Databases“

Vom Prozess geschaffenen Datenobjekte, die als eigenständige Datensätze aufzufassen sind, werden entweder als `«record»` oder als `«document»` gekennzeichnet<sup>2</sup>. ITIL verwendet „Record“ alleinstehend oder in zusammengesetzten Begriffen an vielen zentralen Stellen, liefert aber keine klare Definition. Die Unterscheidung im Modell erfolgt daher, soweit dies sinnvoll möglich ist, nach den Vorgaben der

<sup>2</sup>statt einem einheitlichen Stereotyp `«information»`, wie im ursprünglichen *Thing-Information Pattern*



ISO/IEC 20000: Ein *Record* dokumentiert Aktivitäten (z.B. *Problem Record*), während ein *Document* eher Pläne oder Absichten dokumentiert (z.B. SLA). *Records* und ihre Beziehungen untereinander werden laut ITIL häufig in Datenbanken gespeichert (wenn diese auch immer wieder als Teil der *CMDB* dargestellt werden), die entsprechenden Entitäten tragen im Modell daher den Stereotyp «*database*». Datenobjekte, die in der ITIL nicht als eigenständige Datensätze behandelt oder auch als Attribute von *Documents* oder *Records* auffassbar sind – beispielsweise die *Problem Category*, die zwar theoretisch individuell erfasst werden könnte, aber im Kontext des *Problem Management* nur als Teil eines *Problem Record* einen Sinn ergibt – werden als Klassen ohne Stereotyp dargestellt (könnten aber im Sinne des *Thing-Information Pattern*, ebenso wie die *Records* und *Documents*, den Stereotyp «*information*» tragen).

*Fokus auf grundlegenden Zusammenhängen*

Die ITIL beschreibt nicht nur die Dinge mit denen die Prozesse befasst sind – wie *Incidents*, *Problems* usw. – sondern ja auch die entsprechenden *Documents* und *Records*, wie eben beispielsweise den *Incident Record*. Wären all diese Informationskonzepte in der ITIL kohärent und vollständig dargelegt, so könnten sie in direkter Weise in die entsprechenden Modelle übernommen werden. Eine Modellierung der grundlegenden Entitäten (Dinge) mit ihren Relationen würde dann nicht notwendig sein, da alle wesentlichen Aspekte und Zusammenhänge ja bereits in den *Documents* und *Records* erfasst wären. Leider ist dies nicht der Fall: Für die *Problem Records* liefert die ITIL keine zusammenhängende Beschreibung der Inhalte (außer dass sie „very similar to Incident records“ sind, vgl. Abschnitt 6.6.1 in [OGC00]), für *Incident Records* und *RFCs* werden immerhin Stichpunktlisten geliefert (siehe Annex 5C und Abschnitt 8.3.1 in *Service Support* [OGC00]), die aber nicht als klare und vollständige Spezifikation verstanden werden können. Die weiter unten diskutierten Modelle der Datensicht im *Owner View* versuchen aber auch nicht eine vollständige Aufzählung der in den jeweiligen *Records* erfassten Daten, sondern die Klärung der grundlegenden Zusammenhänge zwischen den *Records*, *Documents* und anderen für das Verständnis der Prozesse wichtigen Entitäten.

*Configuration Management und Service Level Management*

Hierzu gehören mitunter auch Elemente, die eigentlich aus der Domäne eines anderen Prozesses stammen. Obwohl hier weder *Configuration Management* noch *Service Level Management* an sich betrachtet werden sollen, sind doch die von diesen Prozessen verwalteten Daten von so zentraler Bedeutung für alle anderen Dienstmanagementprozesse der ITIL, dass ihre zentralsten Dokumente und Entitäten mitmodelliert werden sollten. Entsprechende Ausschnitte der Datenmodelle zu diesen beiden Prozessen sind in Abbildung A.2 und A.3 wiedergegeben. Die Darstellung stützt sich dabei teilweise auf Vorarbeiten zu *Configuration Management* (siehe [BGSS06]<sup>3</sup>) und *Service Level Management* (siehe [BGN06]).

Stellvertretend für die *Data Models* der drei betrachteten Prozesse sollen zu-

<sup>3</sup>im Vergleich zu [BGSS06] wird im hier vorliegenden Modell der zentrale Datensatz mit dem Begriff *Configuration Record* statt *CI Record* bezeichnet

*Grundlegende  
Konzepte des  
Incident  
Management*

nächst die zentralen Aspekte des Datenmodells zum *Incident Management* erläutert werden. Abbildung 6.2 zeigt einen Ausschnitt aus dem Datenmodell des *Incident Management* (vgl. Abbildungen A.4 und A.3). Das zentrale „Thing“ ist im *Incident Management* natürlich der *Incident*, auf welchem auch der Fokus des Modells liegt. Im Sinne eines Datensatzes oder Dokumentes repräsentiert wird er im *Incident Record*, dem zentralen Datensatz (in diesem Fall *Record*) des Prozesses. Das Modell im *Owner View* liefert allerdings aus Gründen der Übersichtlichkeit keine vollständige Modellierung der Inhalte des *Incident Record*, der neben der Beschreibung des *Incident* sämtliche im Laufe einer *Incident*-Bearbeitung gesammelten Daten sowie die Dokumentation aller in diesem Rahmen unternommener Aktivitäten enthält.

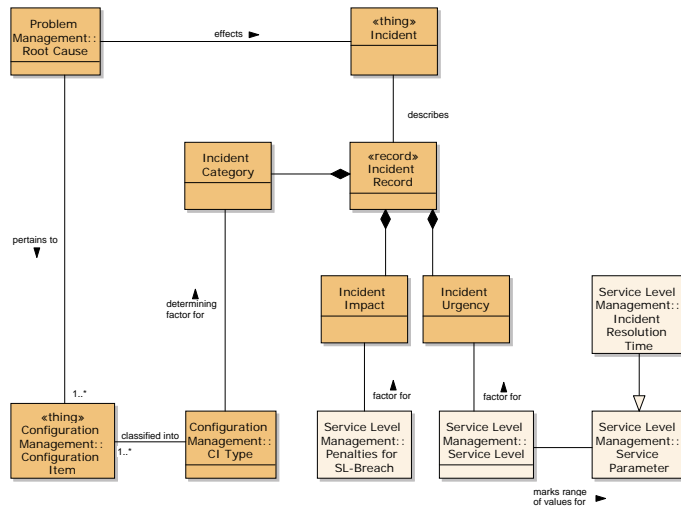


Abbildung 6.2: Ausschnitt aus dem Datenmodell für Incident Management (vgl. Abbildung A.4)

*Zuordnung von  
Entitäten und  
Dokumenten zu  
Prozessen*

Im Datenmodell sind die verschiedenen Entitäten und Dokumente den Prozessen zugeordnet, in deren Domäne sie fallen. Sie werden bei der Darstellung in den Modellen anderer Prozesse (hier *Incident Management*) entsprechend gekennzeichnet. So sind in Abbildung 6.2 beispielsweise auch einige Modellelemente aus dem *Service Level Management* – erkennbar am dem Klassennamen vorangestellten „Service Level Management:“ – erfasst. Die durch diese repräsentierten Informationen zu vereinbarten *Service Levels* und *Penalties* müssen zwar nicht in direkter Form im *Incident Record* gespeichert werden, es macht jedoch Sinn (ein institutionalisiertes *Service Level Management* vorausgesetzt) sie bei der Bestimmung von *Impact* und *Urgency* heranzuziehen (auch wenn dies in der ITIL

nicht explizit so dargestellt ist).

An den Modellen ebenfalls vergleichsweise gut nachzuverfolgen ist der Zusammenhang zwischen den Kategorien von *Incidents* und *Problems*. Die Kategorie eines *Incident* bestimmt sich ebenso wie die eines *Problem* durch den Typ von *Configuration Item*, dem die *Root Cause* des *Problem* zuzuordnen ist<sup>4</sup>. Insgesamt zeigen sich zwischen *Incident Management* und *Problem Management* weitgehende Ähnlichkeiten in Konzepten und verwendeten Begriffen. Im Vergleich hierzu fällt das *Change Management* in einigen Punkten aus dem Rahmen. Vergleicht man das Modell des *Change Management* mit denen des *Incident Management* und *Problem Management*, so fällt auf, dass ersteres merklich komplexer ausfällt. Wie in Abbildung 6.3 verdeutlicht, ist der Begriff der *Category* im *Change Management* ein völlig anderer als in *Incident Management* oder *Problem Management*. Die Kategorie eines *Change* bestimmt sich aus dessen erwarteter Auswirkung und benötigten Ressourcen zur Umsetzung.

Der Begriff  
„Category“ in  
ITIL

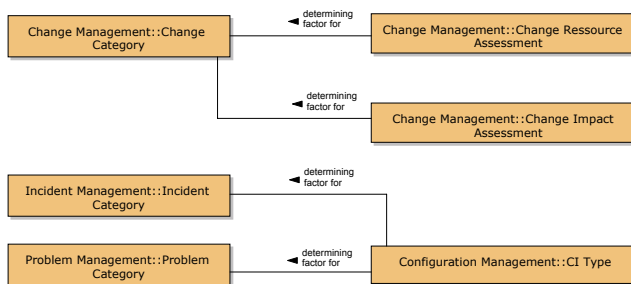


Abbildung 6.3: Haupteinflussgrößen für Einteilung und Zuweisung von Kategorien (vgl. Abbildungen A.4, A.5 und A.6)

Die ITIL bietet also nur für eine Handvoll grundlegender Konzepte eine kohärente Verwendung einheitlicher Begriffe. In der Konsequenz ist für eine detailliertere Darstellung von Aspekten eines Dienstmanagementprozesses häufig schwierig nachzuweisen, dass sie der ITIL entspricht – oder zu belegen, dass sie ITIL-Richtlinien widerspricht. Eine Modellierung der Entitäten und Begriffe der verschiedenen Prozesse ist nichtsdestotrotz durchaus möglich, muss sich aber auf die wesentlichen Kernkonzepte beschränken, will man nicht unmäßig viel Aufwand bei der Auflösung von Widersprüchen treiben, die ja zudem auch zum

Beschränkung  
auf  
Kernkonzepte

<sup>4</sup>In *Service Support*, Abschnitt 5.6.5, heißt es beispielsweise zu *Incident Closure*: „...classification is complete and accurate according to root cause“ – wobei in *Incident Management* und *Problem Management* die Begriffe *Classification* und *Category* meist synonym gebraucht werden. Nach dem ITIL-Glossar [OGC00] ist *Category* eine *Classification* von *Problems*, *Changes* und *Configuration Items* – in ITIL-Schulungen und Sekundärliteratur hat sich hingegen eine hiervon noch einmal leicht abweichende Interpretation durchgesetzt (nämlich dass sich die *Classification* aus *Category* und *Priority* zusammensetzt).

Nachweis der Korrektheit des Modells alle dokumentiert werden müssten.

### 6.1.5 Function Model

Im *Function Model* des *Owner View* steht bei der Modellierung eine ITIL-getreue, aber auch formal korrekte und sinnvolle Wiedergabe des Workflow der untersuchten Dienstmanagementprozesse im Mittelpunkt. Auch für die in Quarant IV einzuordnenden Dienstmanagementprozesse erfordert die Erstellung eines aussagekräftigen Modells einige vorangehende Überlegungen und eine Interpretation und Konkretisierung der in der ITIL zu findenden Texte und Diagramme zu den Prozessabläufen. Für korrekte und prägnante Modelle macht es Sinn, neben notwendigen inhaltliche Ergänzungen zu ITIL-Inhalten, auch die Modelltypen anzupassen.

#### *Ergänzung der im Modell zu erfassenden Aktivitäten*

Von der ITIL werden in Abschnitt 5.6 des Bandes *Service Support* („Incident Management activities“) sechs Aktivitäten des *Incident Management* erläutert<sup>5</sup> (vgl. Abschnitt 3.1.3 ). Diese können, wie in Abbildung A.7 auf Seite 169 zu sehen, mit ihren originalen Bezeichnungen in das Modell übernommen werden. Allerdings befasst sich schon der nächste Abschnitt der ITIL (5.7 in *Service Support*) mit *Handling of Major Incidents*. Dass sich, wie man bei einem Vergleich der entsprechenden Richtlinien erkennen kann, die Bearbeitung dieser *Major Incidents* zur der normaler *Incidents* prinzipiell ähnlich verhält, wie die Behandlung von *Urgent Changes* zum regulären Ablauf des *Change Management*, sollte sich auch in einer äquivalenten Darstellung widerspiegeln. Entsprechend wird auch *Handling of Major Incidents*, wie die entsprechenden Vorgänge im *Change Management*, als Aktivität interpretiert und in das Modell des Workflows integriert. Ebenso wird, da es letztlich doch einem einfacheren Workflow folgt, die Behandlung von Dienstanfragen als eigene Aktivität *Service Request Handling* hinzugefügt (farblich abgesetzt, da es sich hier um eine vom Autor ergänzte Aktivität handelt).

#### *Ausschließen von zu erfassenden Aktivitäten*

Im *Change Management* liegt der Fall hingegen teilweise umgekehrt. Hier sind in der Sektion „Activities“ (8.5 in *Service Support* [OGC00]) einige Punkte<sup>6</sup> aufgeführt, die sich nicht vernünftig in einen Workflow des Hauptprozesses integrieren, ja teilweise beim besten Willen noch nicht einmal überhaupt als Aktivitäten interpretieren lassen.

#### *kritische Beurteilung der ITIL-Inhalte*

Auch hier folgt die Darstellung der Inhalte in der ITIL also wieder keinem nachvollziehbaren Muster und somit kann auch keine allgemeine Methode für den Weg

<sup>5</sup> *Incident Detection and Recording, Classification and Initial Support, Investigation and Diagnosis, Resolution and Recovery, Incident Closure und Ownership, Monitoring, Tracking and Communication*

<sup>6</sup> *Planning the Implementation of Operational Processes, CAB meetings, Reviewing the Change Management Process for Efficiency and Effectiveness, Roles and Responsibilities, Establishing a Change Advisory Board*

von den ITIL-Richtlinien hin zum Modell beschrieben werden – auch für den Abstraktionstyp *Function* muss die Modellerstellung für jeden Prozess individuell nach den in Abschnitt 5.3 festgelegten Richtlinien und gesundem Menschenverstand erfolgen.

Neben der Interpretation der ITIL-Inhalte sind auch Fragen der Darstellung von ITIL-Prozessen in Workflow-Modellen eine kurze Diskussion wert. Eine Besonderheit des *Incident Management* ist, dass die ITIL explizit darauf hinweist, dass aus (fast) jeder Aktivität heraus eine hierarchische Eskalation (in der Regel zum *Incident Manager*) erfolgen kann. Dies trifft, wenn in der ITIL auch nicht mit dem Begriff *Escalation* ausgedrückt, so aber durch die ebenfalls in fast allen Aktivitäten potentiell notwendige Re kategorisierung von *Changes* in eine höhere Kategorie gegeben, auch auf das *Change Management* zu. Eine direkte Berücksichtigung dieses Umstandes in der Darstellung des Kontrollflusses würde aber das Modell stark verkomplizieren, müsste doch von jeder Aktivität hin zur Eskalationsaktivität verzweigt werden.

*Eskalationen  
schwierig zu  
modellieren*

Einen Ausweg bietet die Verwendung von stereotypisierten Aktivitäten. Wie in Abbildung 6.4 gezeigt, wird in den Modellen für *Change Management* und *Incident Management* eine Aktivität mit einem nach oben zeigenden Pfeil in der rechten oberen Ecke der Aktivität als ein grafischer Stereotyp zum Darstellen einer eskalierbaren Aktivität verwendet<sup>7</sup>. Wie man in den Abbildungen zu *Incident Management* (A.7) und *Change Management* (A.15) sieht, betrifft dies alle Aktivitäten<sup>8</sup> dieser beiden Prozesse außer denen, für die in einer Ausnahmesituation eh schon die höchste verfügbare *Authority* involviert ist (also die Aktivitäten zur Behandlung von *Major Incidents* und *Urgent Changes*).

*Stereotypen zur  
Kennzeichnung  
eskalierbarer  
Aktivitäten*

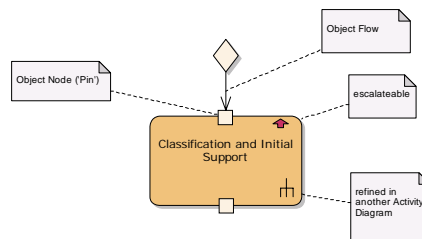


Abbildung 6.4: Darstellung des Workflows von ITIL-Prozessen mittels Aktivitätsdiagrammen (vgl. Abbildungen A.7, A.14 und A.15)

Nicht für die Darstellung des Ablaufes an sich notwendig, aber eine weitere

<sup>7</sup>Das Symbol in der rechten unteren Ecke, die so genannte „Gabel“, ist die gängige Kennzeichnung einer Aktivität, die in einem anderen Diagramm verfeinert dargestellt wird.

<sup>8</sup>in den verfeinerten Aktivitäten des *Incident Management* kommt dieser Stereotyp nicht mehr zum Einsatz

*Weiterreichen  
der Records als  
Objektfluss*

Besonderheit der drei betrachteten Prozesse schön illustrierend, ist die Erfassung eines Objektflusses in den Aktivitätsdiagrammen. In den drei betrachteten Prozessen werden alle die Prozessinstanz betreffenden Informationen<sup>9</sup> in einen zentralen, für das jeweilige *Incident*, *Problem* oder *Change* spezifischen *Record* gepackt. Ohne dass dies in der ITIL explizit so erklärt wird, ist die Idee dahinter offensichtlich, dass hiermit alle notwendigen Informationen zwischen den verschiedenen Prozessbeteiligten übermittelt werden. Diese Weitergabe des *Record* über den Prozessablauf hinweg ist, wie ebenfalls in Abbildung 6.4 illustriert, als Objektfluss modelliert, wie man an den *Object Nodes* (so genannten „Pins“) der Aktivitäten erkennen kann (oder das Objekt wird explizit dargestellt, wie beispielsweise in der Partition *Support Groups & External Support* in Abbildung A.7).

*Workflow-  
Integration*

Als letzter Punkt der Diskussion des *Function Model*, soll die Frage der Workflow-Integration herausgegriffen werden. Auch diese wird in der ITIL selbst wieder nur partiell und indirekt adressiert. Vergleicht man die Beschreibungen zu *Incident Management*, *Problem Management* und *Change Management*, so lassen sich unter Ergänzung sinnvoller Workflow-Element zum *Change Management* (Informieren des Antragstellers nach Ablehnung oder Implementierung eines *Change*, vgl. Abbildung A.15) Integrationspunkte von *Incident Management* und *Problem Management* mit dem *Change Management* identifizieren und im Modell aufzeigen. Beispielhaft ist dies in Abbildung 6.5 verdeutlicht: Die Verbindung der Workflows von *Incident Management* und *Change Management* entspricht dabei weitgehend dem Muster einer hierarchischen Beziehung im Sinne der Szenarios der *Workflow Interoperability* im *Workflow Reference Model* der *Workflow Management Coalition*[Hol95], allerdings endet der Prozess des *Change Management* nicht mit der Implementierung, sondern durchläuft danach noch die Aktivität *Change Review* (deren Abschluss man als synchronisiert mit dem Abschluss von *Incident Closure* bzw. *Error Closure* interpretieren kann).

*Integration von  
Incident  
Management  
und Problem  
Management*

Wie eine entsprechende Integration zwischen *Incident Management* und *Problem Management* sich gestalten ließe, lässt die ITIL allerdings völlig offen. Der ja durchaus mögliche Fall, dass während der Bearbeitung eines *Incident*, im *Problem Management* ein *Work-around* identifiziert oder gar die *Root Cause* behoben wird, findet in der ITIL keine Erwähnung. Auch die (außerhalb des Rahmens der hier betrachteten Prozesse) notwendige Workflow-Integration zwischen *Release Management* und *Change Management* bleibt in *Service Support* weitgehend unklar.

---

<sup>9</sup>im Sinne der Workflow Data Patterns [RtHEvdA04] das *Case Data*

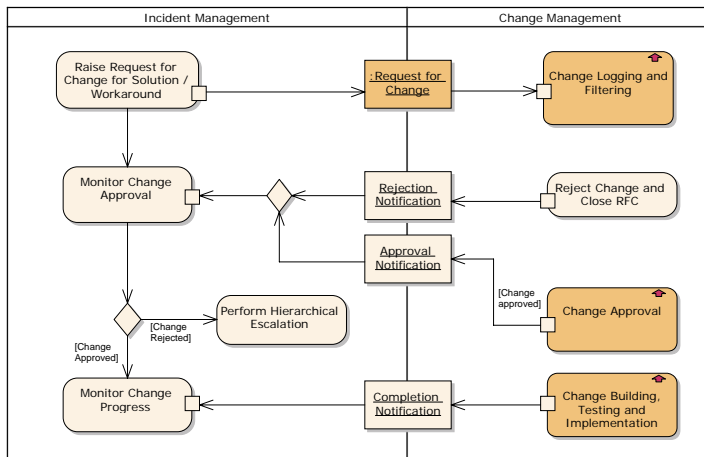


Abbildung 6.5: Integration zwischen Incident Management und Change Management (vgl. Abbildungen A.11 und A.15)

### 6.1.6 People Model

In *Service Support* werden im Abschnitt *Roles of the Incident Management Process* (5.8) nur zwei grundsätzliche Rollen definiert, der *Incident Manager* und der *Incident-Handling Support Staff*, welchem eine ganze Reihe von Verantwortlichkeiten zugeschrieben werden. Letzteres ist eine Rolle, die in aller Regel von mehreren Personen gleichzeitig ausgefüllt werden muss und im Normalfall weitgehend dem *Service Desk* zugewiesen ist. Allerdings tauchen an anderen Stellen in den Richtlinien zu *Incident Management* verschiedene andere Rollen auf, die ebenfalls Eingang in das Modell gefunden haben (vgl. Abbildung A.17). Die Darstellung der Modelle vom Abstraktionstyp *People* erfolgt wie beim *Data Model* in einem Klassendiagramm, in dem die in Nutzfalldiagrammen verwendeten Aktoresymbole als Stereotypen für Rollen verwendet werden. Der *Service Desk* ist mit einem anderen visuellen Stereotyp dargestellt, der häufig verwendet wird um außenorientierte Geschäftsmitarbeiter zu modellieren (vgl. [OWS<sup>+</sup>03]).

*Darstellung als  
Klassenmodell*

Dies schließt zunächst die Betrachtungen des *Owner View* ab. Bis hierhin waren die Betrachtungen unabhängig von eingesetzten IT-Systemen. Dies ändert sich im folgenden *Designer View*, in dem festgelegt wird, wie der Geschäftsprozess durch ein IT System, in diesem Fall ein *Incident Management System*, unterstützt werden kann und welche grundlegenden Eigenschaften bzw. welches prinzipielle Design dieses besitzen sollte.

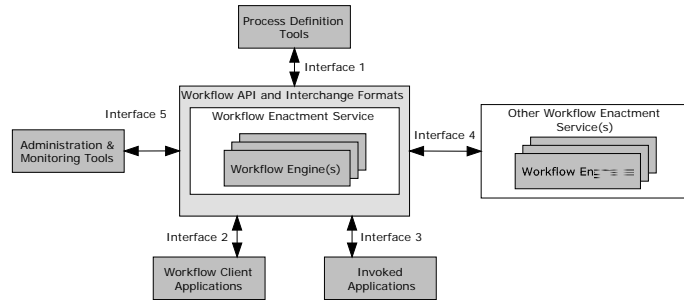


Abbildung 6.6: Architekturelemente im Workflow Reference Model [Hol95]

## 6.2 Designer View

Im *Designer View (System Model)*, also der Perspektive des IT-Architekten, sind bei einem, wie in Abschnitt 5.4.2 vorgeschlagenen Top-Down-Vorgehen durch den Ordnungsrahmen zum ersten Mal IT-spezifische Fragestellungen im Vordergrund.

### 6.2.1 Grundlegender Architekturansatz

#### *Orientierung am Workflow Reference Model*

Wie in Kapitel 4 gesehen, ist für eine Unterstützung des *Incident Management* ein Workflow-orientiertes Managementwerkzeug am besten geeignet. Ein entsprechendes Incident-Management-System wird also eine Art von Workflow-Management-System sein müssen. Für den Entwurf und die Beschreibung eines neuen Systems ist es natürlich vorteilhaft, wenn man sich an etablierte Vorgaben halten kann, die einige grundlegende Designentscheidungen in vernünftiger Weise vorwegnehmen und eine Vergleichbarkeit der entwickelten Architektur ermöglichen. Im Bereich der Workflow-Management-Systeme bietet sich hier natürlich das *Workflow Reference Model* der *Workflow Management Coalition* [Hol95] an, dessen grundlegende Architekturelemente in Abbildung 6.6 dargestellt sind.

#### *Architekturelemente des Workflow Reference Model*

Die Architektur unterscheidet verschiedene Systemkomponenten, in deren Mittelpunkt der *Workflow Enactment Service*, ein Software-Service, welcher mittels ein oder mehrerer *Workflow Engines* die Ausführung und Steuerung von Workflows ermöglicht, steht. Der Zugriff der verschiedenen Arten von anderen Systemkomponenten (Tools) auf diesen Service geschieht über Schnittstellen, die im *Workflow Reference Model* in sechs grundlegende Typen unterteilt werden (*Interface 1* bis *Interface 6*).



## 6.2.2 Modelle

Für das *Incident Management* wird eine Architektur vorgeschlagen, die als zentrale Komponente einen *Incident Management Workflow Service* definiert (welcher im Sinne des *Workflow Reference Model* einen *Workflow Enactment Service* darstellt). Die grundlegenden Systemkomponenten sind im *Network Model* in Abbildung A.24 dargestellt (der *Event Correlation Service* steht dabei nur beispielhaft für möglicherweise in den Workflow integrierbare Tools des infrastrukturorientierten Managements).

Man kann, gerade im Lichte der gegenwärtigen Popularität des Konzeptes, argumentieren, dass ein Incident-Management-System im Sinne einer besseren Anpassbarkeit und Integrierbarkeit nach SOA-Richtlinien entwickelt werden sollte. Eine allgemein anerkannte Definition dessen, was eine *Service Oriented Architecture* genau ausmacht, existiert allerdings bislang nicht (siehe auch Diskussion hierzu in [BDRH<sup>+</sup>07]). Nach den gängigen SOA-Vorstellungen lässt sich aber die verteilte Architektur des *Workflow Reference Model* durchaus als Ausprägung einer SOA interpretieren. Es ist durchaus möglich verschiedene Architekturelemente des *Workflow Reference Model* auf SOA-Konzepte, wie sie beispielsweise in [KBS05] definiert werden, abzubilden – ein Incident-Management-System nach dem *Workflow Reference Model* ließe sich so ohne weiteres in eine SOA integrieren<sup>10</sup>. Das *Network Model* der Systemarchitektur in Abbildung A.24 stellt daher die Komponenten (Teilsysteme) als Stereotypen der in [KBS05] definierten aktiven Elemente einer SOA dar.

Zuordnung zu  
SOA-Elementen

Die Darstellung im *Data Model* konkretisiert den Inhalt des *Incident Record*, wobei die Attribute, die im Sinne des *Workflow Reference Model* als reine *Workflow Application Data* gelten können, in der Klasse als **private** markiert (vorangestelltes „-“) sind<sup>11</sup>. Für den Abstraktionstyp *Function* bietet sich die Modellierung von Nutzfällen der verschiedenen Teilsysteme an, die im vorliegenden Modell nur exemplarisch in A.22 und A.23 erfolgt. Bei der Modellierung von Nutzfällen mit Aktoren ergibt sich quasi zwangsläufig auch eine teilweise Modellierung des *People Model*. Das explizite *People Model* (Abbildung A.25) kann daher vergleichsweise knapp ausfallen, wenn die Architektur wie im vorliegenden Fall vorsieht, für jede grundlegende Rolle im Prozess eine dedizierte *Workflow Client Application* zu schaffen. Dies erlaubt, wenn die Interaktion zwischen *Workflow*

„Thick“  
Workflow  
Clients

<sup>10</sup>Einige in [KBS05] genannte SOA-Schlüsselkonzepte, wie die Verwendung eines *Service Repository*, finden im *Workflow Reference Model* zwar keine Entsprechung, könnten aber ohne weiteres ergänzt werden (wenn dies im Kontext eines Incident-Management-Systems auch nicht unmittelbar lohend scheint).

<sup>11</sup>Dies ist noch nicht als konkreter Objektentwurf im Sinne einer objektorientierten Softwareentwicklung zu verstehen (dies würde ggf. erst im *Builder View* erfolgen) – als **private** sind hier alle Attribute markiert, die der *Incident Management Workflow Service* definitiv nicht interpretieren muss. In einem konkreten Objektentwurf könnten selbstverständlich alle Attribute als **private** deklariert werden und der Zugriff des *Incident Management Workflow Service* auf sie über Operationen geregelt werden.

*Enactment Service* und *Workflow Client Application* nach dem *Electronic Mail Model* erfolgt (also die *Worklist* beim *Client* vorgehalten wird), die Verlagerung eines großen Teils der Komplexität von Autorisierungs- und Authentifizierungsaufgaben in die *Workflow Client Applications*, was in Anbetracht der potentiell zahlreichen im *Incident Management* zu überwindenden Organisationsgrenzen als vorteilhaft angesehen werden kann.

### 6.3 Überlegungen zum Builder View

Die Modelle des *Builder View* (*Technology Model*) stellen eine technologiespezifische beziehungsweise plattformspezifische Umsetzung der im *Designer View* entworfenen Architektur dar. Bei einem Top-Down-Vorgehen, wie im vorliegenden Fall, muss daher zunächst entschieden werden, auf welcher technischen Basis die konkrete Umsetzung des Systemdesigns erfolgen soll.

*XML Web Services als zentrale Realisierungstechnologie*

Möchte man auch eine spätere Integration mit anderen Systemen erleichtern bzw. das zu entwickelnde System in eine SOA einbinden, so bietet sich die Realisierung eines verteilten Systems, wie es das im *Designer View* skizzierte Incident-Management-System darstellt, unter Einsatz von (XML-) Web-Services-Technologien an. Die Definition entsprechender Datentypen für das *Incident Management* lässt sich vergleichsweise leicht in einem XML-Schema realisieren (vgl. Abbildung A.26)

„*Basic Control-Flow Patterns*“  
*hinreichend*

Für die funktionalen Anforderungen an die *Workflow Engine* muss untersucht werden, ob sie alle Abläufe, die man zu definieren wünscht, auch ausführen kann. Dies ist letztlich davon abhängig, welche Ausdrucksstärke sie in den von ihr verarbeiteten Workflow-Definitionen erlaubt. Arbeiten im Bereich der *Workflow Patterns* [RtHvdAM06] erlauben die Bewertung von Workflow-Management-Produkten anhand der Kontrollfluss-Konstrukte (Patterns), die sie unterstützen. Die Auswahl ist im vorliegenden Fall aber nicht von entscheidender Bedeutung. Weder der eigentliche Prozess des *Incident Management* (vgl. Abschnitt 6.1.5), noch die Kontrollflüsse der einzelnen Nutzfälle des in Grundzügen entworfenen Incident-Management-System (vgl. Abschnitt 6.2) sind sonderlich komplex – alle lassen sich mit den *Basic Control-Flow Patterns*<sup>12</sup> und dem Pattern *Structured Loop* realisieren, welche von fast allen in [RtHvdAM06] untersuchten Produkten und Sprachen unterstützt werden.

*Orchestrierung mittels BPEL Engine*

Unter diesen, von einer *Workflow Engines* ausführbaren („executable“) Prozessbeschreibungssprachen finden sich mittlerweile auch herstellerunabhängige, XML-basierte (vgl. auch Abschnitt 3.10) Sprachen zur Koordination (*Orchestration*) von Web-Services. In einer ersten prototypischen Implementierung einer Workflow-Steuerung für ein Incident-Management-System wurde, primär aus

<sup>12</sup> *Sequence, Parallel Split, Synchronization, Exclusive Choice, Simple Merge*

Gründen der breiten Verfügbarkeit entsprechender Produkte eine Umsetzung einer Auswahl von im *Incident Management* auftretenden Workflow-Strukturen in BPEL 1.1 [ACD<sup>+</sup>03]<sup>13</sup> realisiert [Koy07].

Weder für die modellgetriebene Entwicklung von BPEL noch für die Darstellung von Web-Services-Schnittstellen und -Kommunikationsmustern hat sich bislang allerdings eine graphische Notation durchgesetzt. Es gibt zwar Ansätze für UML-basierte Darstellungsformen (z.B. mittels eines *UML Profile* [AGGI03]), wie sie auch für die Abbildungen A.28 und A.29 zum Einsatz kamen - auch BPMN wird hierfür eingesetzt. Produkte, die eine grafisch unterstützte Web-Services- oder BPEL-Entwicklung erlauben, setzen aber bislang weiterhin überwiegend auf proprietäre Darstellungsformen.

## 6.4 Erfahrungen

Die in diesem Kapitel dargelegten Erläuterungen zu den Modellen können deren Inhalte und die der Erstellung zugrunde liegenden Überlegungen aus Platzgründen nur ausschnittsweise wiedergeben. Im folgenden sollen aber noch einmal allgemeine Erfahrungen bei der Modellerstellung zusammengefasst werden.

Generell zeigt sich bei Verwendung des vorgeschlagenen Ordnungsrahmens, dass die Zuordnung von in einem bestimmten UML-Diagrammtyp erstellten Modellen zu einem Abstraktionstyp des Ordnungsrahmens nicht immer ganz eindeutig ist. Viele Modelltypen verknüpfen mehrere Abstraktionstypen miteinander, wie beispielsweise Nutzfalldiagramme (*Use Case Diagrams*) den Abstraktionstyp *Function* und *People*<sup>14</sup>. Dies spricht jedoch nicht gegen den Einsatz solcher Modelle, zeigen doch gerade sie oftmals die interessantesten Zusammenhänge auf. Der Ordnungsrahmen sollte als nützliches Mittel für die Gewinnung eines Überblicks über die im Gesamtmodell adressierten Aspekte, nicht als eine Einschränkung, welche nicht exakt platzierbare Modelle ausschließt, aufgefasst werden.

Die Modellierung von Dienstmanagementprozessen auf der Ebene des *Owner View* gelingt mit den Mitteln der UML recht gut – objektive Nachteile gegenüber einer Darstellung mit Modelltypen des ARIS-Frameworks lassen sich nicht feststellen, auch wenn die vielfältigeren Darstellungsmöglichkeiten der UML-Diagrammtypen zweifelsohne eine Eingewöhnung für mit der UML nicht vertraute Leser erfordern. Auf der Ebene des des *Designer View* ist die UML ohnehin „zu Hause“. Für die Illustration einer XML-, Web-Services- und BPEL-basierten Architekturumsetzung wünscht man sich bei der Modellierung des *Builder View* an einigen Stellen allerdings doch an diese Technologien besser angepasste Darstellungsmöglichkeiten.

*Erfahrungen bei Verwendung des Ordnungsrahmens*

*UML gut geeignet zur Modellierung von Dienstmanagementprozessen*

<sup>13</sup>ausgeführt auf der *ActiveBPEL Open Source Engine*, <http://tinyurl.com/2mq82e>

<sup>14</sup>Eindeutig zum Abstraktionstyp *Function* zuzuordnen wäre im Vergleich dazu ein Nutzfalldiagramm [Stö05].

*Incident Management und Problem Management auffällig ähnlich* Inhaltlich fallen die Erfahrungen der Modellierung von ITIL-Prozessen im *Owner View* durchwachsen aus. *Incident Management* und *Problem Management* ähneln sich in vielen Aspekten wie auch in der Art ihrer Darstellung in *Service Support*. Für beide sind auch beispielsweise, im Gegensatz zum *Change Management*, die Prozessaktivitäten für ITIL-Verhältnisse klar und übersichtlich dargestellt. Ihre Modellierung fällt daher leichter als die des in vielen Aspekten komplexeren *Change Management*, dessen Darstellung obendrein erheblich schlechter strukturiert erscheint und einige Begriffe völlig anders verwendet. Auf der anderen Seite sind aber *Incident Management* und *Problem Management* nicht in allen Aspekten konzeptuell sauber getrennt und die Integration ihrer Workflows miteinander bleibt weitgehend unklar.

## 7 Zusammenfassung und Ausblick

In den vorangegangenen Kapiteln wurde ein methodischer Ansatz für die Werkzeugunterstützung von ITIL-Dienstmanagementprozessen entwickelt. Die vorgefundene Unübersichtlichkeit des Status Quo im Dienstmanagement allgemein und in der Werkzeugunterstützung der ITIL im besonderen machten hierfür auch einige Vorarbeiten notwendig. In dem weiten und wissenschaftlich bislang vergleichsweise wenig bearbeiteten Feld des prozessorientierten Dienstmanagements stellen diese Klassifikationen und Einordnungen, der vorgestellte Ansatz und die für dessen beispielhafte Umsetzung entwickelten Modelle dabei keine bloßen Zwischenergebnisse einer Werkzeugentwicklung dar, sondern bieten vor allem auch Motivation, Ansatzpunkt und Diskussionsbasis für eine Vielzahl offen bleibender Fragestellungen.

### 7.1 Ergebnisse und Fazit

Das prozessorientierte IT-Dienstmanagement befindet sich gegenwärtig in einem noch frühen Stadium, was sich unter anderem an der häufig inkonsistenten, mehrdeutigen oder sogar widersprüchlich Verwendung von Begriffen festmachen lässt. In der vorliegenden Arbeit wurden daher relevante Begriffe, für die sich noch kein allgemeiner Konsens hinsichtlich ihrer genauen Bedeutung und Verwendung etabliert hat, eingeführt um so den begrifflichen Kontext für die weiteren Kapitel festzulegen.

Die ITIL ist der gegenwärtig populärste, aber bei weitem nicht in der einzige Standard im diesem Umfeld. Herauszuarbeiten, was die essentiellen Unterschiede in Zielsetzung und grundlegender Methodik zwischen den zahlreichen Rahmenwerken sind, ist effizient oftmals nur mit einer regelrecht hermeneutischen Herangehensweise möglich, bei der die eigentlichen Texte auch unter Berücksichtigung des Entstehungskontextes des jeweiligen Standards (*IT Auditing, Software Engineering, Telecommunication Management, ...*) interpretiert werden. Um zumindest schnell abschätzen zu können, wo sich prinzipiell Ergänzungen oder Überschneidungen zwischen den Inhalten der verschiedenen Standards ergeben können, wurde in dieser Arbeit ein auf Lebenszyklus-Aspekten und technischer Orientierung basierendes Einordnungsschema entwickelt, in welches die zur Zeit prominentesten Rahmenwerke platziert wurden.

*Einordnung von  
ITSM-  
Rahmenwerken*

### notwendige Konkretisierung von ITIL

Auch im Bereich des ITIL-orientierten Dienstmanagements steht die Forschung noch weitgehend am Anfang. In der Tat herrscht gerade im ITIL-Umfeld unter vielen Anwendern, Beratern und Buchautoren eine regelrechte Abneigung gegen „theoretische“ Konzepte. In einem noch früheren Stadium, in dem nur wenige Inhalte zu diesem Gebiet überhaupt existierten, mögen die Nachteile der gängigen Praxis – Inhalte kaum strukturiert, unter steter Abwandlung von verwendeten Begriffen und ohne ausreichende Einbindung in den Kontext anderer Arbeiten darzustellen – hinnehmbar im Vergleich zum Aufwand einer formaleren und genaueren Darstellung gewesen sein. Mittlerweile ist aber ein Punkt erreicht, an dem, jenseits eines am Prüfungsstoff der ITIL-Personenzertifizierungen orientierten Konsens über die Kernaspekte der Prozesse (welcher sich auf wenigen Seiten zusammenfassen ließe), eine weitgehende Beliebigkeit in den meisten Details der Prozessgestaltung herrscht. Benötigt man als potentieller Anwender oder als Software-Entwickler, der Dienstmanagement-Werkzeuge herstellen will, eine Konkretisierung oder Detaillierung bestimmter Sachverhalte, so sieht man sich einem Wust von hunderten bis tausenden Seiten Standarddokumenten und Sekundärliteratur gegenüber, denen sich belastbare Aussagen nur sehr mühsam abringen lassen.

### Klassifikation von Werkzeugen und Dienstmanage- mentprozessen

Gerade in Hinblick auf eine Werkzeugunterstützung finden sich kaum konkrete Ansatzpunkte. Auch auf der Seite der Managementwerkzeuge sind bislang keine Konzepte entwickelt worden, die auch die Unterstützung der verschiedenen Arten von Dienstmanagementprozessen in einer allgemeinen und umfassenden Weise berücksichtigen. Es war daher in dieser Arbeit zunächst notwendig, Klassifikations-schemata sowohl für Managementwerkzeuge wie für Dienstmanagementprozesse zu entwickeln, in welche existierende und neu definierte Werkzeugtypen, respektive die ITIL-Prozesse, eingeordnet werden konnten.

### modellbasierter Ansatz

Für ein methodisches Herangehen an die Frage der Werkzeugunterstützung wurde ein modellbasierter Ansatz entwickelt, für den zunächst konkrete Richtlinien bei der Erstellung von Referenzmodellen von ITIL-Prozessen aus den Grundsätzen ordnungsgemäßer Modellierung abgeleitet wurden. Nach einem Vergleich in Frage kommender Rahmenwerke zur modellbasierten Entwicklung von Software-Systemen, wurde auf Basis des *Zachman Framework* eine grundlegende Vorgehensweise und ein Ordnungsrahmen entworfen, die eine strukturierte Modellentwicklung erleichtern.

### Modelle von ITIL-Prozessen und Werkzeugen

Dieser Ansatz wurde auf einen Ausschnitt der ITIL angewendet. Dabei wurde bei der Erstellung von Modellen der ITIL-Prozesse in UML sowohl von bekannten Modellierungskonzepten (z.B. *Thing-Information Pattern*) Gebrauch gemacht, wie auch eigene Lösungen zur prägnanten Darstellung von ITIL-typischen Sachverhalten (z.B. Verwendung von Stereotypen zur Darstellung eskalierbarer Aktivitäten) entwickelt. Beispielhaft für das *Incident Management* wurde auch eine am *Workflow Reference Model* angelehnte Architektur für ein unterstützendes System skizziert und modelliert sowie die Möglichkeit einer Umsetzung dieser

Architektur auf Basis von Web-Services diskutiert. Die interessantesten Ergebnisse ergeben sich aber wohl bei den Prozessmodellen, anhand derer sich verschiedene Konzepte und Beziehungen im Vergleich zur textuellen Beschreibung in der ITIL sehr viel plastischer darstellen lassen, bei deren Erstellung aber auch einige Widersprüche, Inkonsistenzen und Lücken der ITIL besonders augenfällig werden.

## 7.2 Offene Forschungsfragestellungen

Die in der Arbeit getroffene Einordnung von ITSM-Rahmenwerken kann herangezogen werden um zu bestimmen, welche Standards sich potentiell sinnvoll ergänzen, also gleichzeitig in einer Organisation zum Einsatz kommen können – und in Konsequenz, für welche sich die Untersuchung einer Abbildungsbeziehungweise von Integrationskonzepten besonders lohnt. Der Einsatz prozessorientierter ITSM-Standards steht aber in den meisten IT-Organisationen noch am Anfang, weshalb die Fragen der Kompatibilität und Integrierbarkeit beim gleichzeitigen Einsatz mehrerer Dienstmanagement-Frameworks wohl erst in einiger Zeit drängend werden. Für die Praxis unmittelbarer interessant erscheint aber eine Abbildung der bereits heute weiträumig eingesetzten ITIL mit den im Umfeld der Unternehmens-IT ebenfalls weit verbreiteten Vorgehensmodellen zur Systementwicklung wie *Rational Unified Process* (RUP) oder V-Modell XT. Gerade bei der Identifizierung von Integrationspunkten zwischen Rahmenwerken des Dienstmanagements und des *Software Engineering* führt aber an der Verwendung von Modellen kaum ein Weg vorbei (siehe auch [BGN06]).

*Integration von Rahmenwerken des Dienstmanagements und des Software Engineering*

Auch im Bereich der Werkzeugunterstützung für Dienstmanagementprozesse sind zahlreiche wichtige Fragen offen. Eine Einordnung der verschiedenen auf dem Markt befindlichen Managementwerkzeuge in das in dieser Arbeit vorgestellte Klassifikationsschema könnte eine aussagekräftigere Bestandsaufnahme dieses Produktumfeldes liefern als die bislang vorhandenen Marktübersichten. Auch eine Anpassung des vorgestellten Ansatzes für die in Quadrant III einzuordnenden Prozesse der ITIL (*Release Management* und *Service Level Management*) konnte in dieser Arbeit nicht mehr berücksichtigt werden. Während hierfür auf der Ebene des *Owner View* wohl nur kleinere Anpassungen in Bezug auf den Ordnungsrahmen und die verwendeten Modelltypen erforderlich sind, müsste das Systemdesign auf der Ebene des *Designer View* wohl den anderen Typus des zu entwickelnden prozessorientierten Werkzeugs berücksichtigen.

*Weiterentwicklung des vorgestellten Ansatzes*

Auch abgesehen von der Verfolgung einer Werkzeugunterstützung läge ein erheblicher Nutzen in einer Vervollständigung der erstellten Prozessmodelle, sowohl hinsichtlich der verbleibenden Dienstmanagementprozesse der ITIL wie auch weiterer Aspekte, beispielsweise der Erfassung der KPIs in den Modellen. Für die in dieser Arbeit vorgestellte Architektur eines Incident-Management-Systems sollte

die konkrete Umsetzung vorangetrieben werden, um praktische Erfahrungen zu ihren Vor- und Nachteilen zu gewinnen – auch die genauere Untersuchung der Möglichkeiten einer Integration anderer (nicht-prozessorientierter) Management-Werkzeuge in die Architektur steht noch aus.

zu großer  
Interpretations-  
spielraum der  
ITIL

Das größte zu überwindende Hindernis bei der Entwicklung von Werkzeugen für ein ITIL-orientiertes Dienstmanagement bleiben aber die (unter diesem Aspekt feststellbaren) Unzulänglichkeiten der ITIL selbst. Für die Prozessgestaltung und -einführung in einer IT-Organisation mag die sehr große Interpretierbarkeit der ITIL nicht immer ein Nachteil sein – für die Entwicklung eines allgemeinen, nicht auf ein bestimmtes Einsatzszenario spezialisierten Werkzeugs ist sie es schon. Zudem bleiben in der ITIL auch einige wichtige Fragen, wie beispielsweise die der Workflow-Integration zwischen *Incident Management* und *Problem Management*, offen, die in einer allgemeinen Weise und nicht erst bei der Werkzeugentwicklung adressiert werden sollten.

Entwicklung  
eines  
ITIL-basierten,  
aber  
konkretisierten  
Rahmenwerks

„ITIL-Werkzeuge“ im engeren Sinn könnten nur auf Basis einer entsprechenden Spezifikation entstehen, welche wiederum eine Konkretisierung der ITIL-Richtlinien als Basis erfordern würde. Die vorliegende Arbeit und die für sie entwickelten Modelle können hier sowohl auf Prozess- wie auf Systemebene eine Diskussionsbasis schaffen. Die im Moment immer noch ansteigende Popularität der ITIL und die, trotz der für Kunden völlig unklaren und konfusen Situation, augenscheinlich nicht schlecht gehenden Geschäfte mit „ITIL-Tools“ lassen allerdings eine gewisse Skepsis angebracht erscheinen, ob für ein entsprechendes Unterfangen in der ITIL-Community ein großer Bedarf gesehen wird. Es bleibt daher wohl nur die Entwicklung eines für die Werkzeugentwicklung hinreichend konkretisierten Prozessrahmenwerks, welches auf der ITIL aufbaut, aber letztlich nicht mehr mit ihr deckungsgleich sein kann. Dies ist offensichtlich auch der Weg, den IBM für seine Managementwerkzeuge mit ITUP beschreitet – es wäre wünschenswert, wenn sich aus dem wissenschaftlichen Umfeld parallel hierzu auch ein entsprechender herstellerunabhängiger Standard entwickeln würde.



# A Modelle

Die im Folgenden gesammelten Modelle sind nach dem betrachteten Prozess und ihrer Einordnung in den vorgestellten Ordnungsrahmen (vgl. Abschnitt 5.4.1 und insbesondere Abbildung 5.5 auf Seite 138) sortiert. Neben jedem Modell ist am Seitenrand seine Einordnung in Stichworten (Perspektive / Abstraktionstyp / Prozess) und in einem wie Abbildung A.1 aufgebauten Pictogramm vermerkt. Bei Modellen, die keinen der drei betrachteten Prozesse (*Incident Management*, *Problem Management* und *Change Management*) direkt betreffen, aber notwendige Konzepte anderer ITIL-Prozesse ausschnittsmäßig darstellen (z.B. aus dem *Configuration Management*) erfolgt die Nennung des Prozesses nur in der Abbildungsunterschrift.

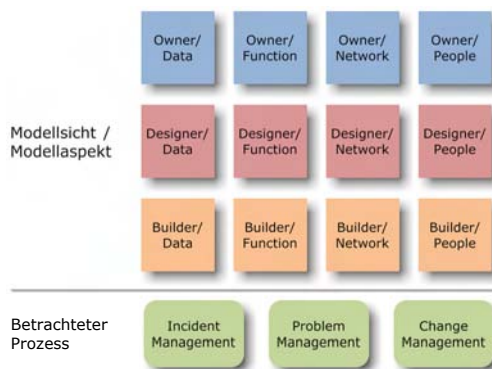


Abbildung A.1: Einordnung der Modelle

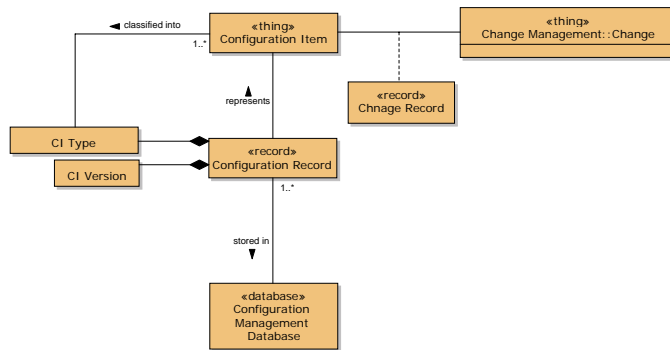
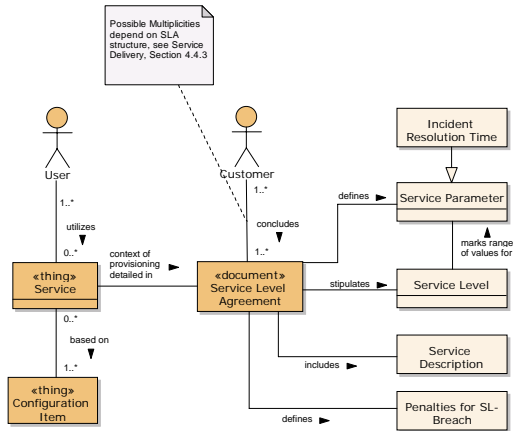


Abbildung A.2: Für die anderen Prozesse relevante zentrale Entitäten und Dokumente im *Configuration Management*



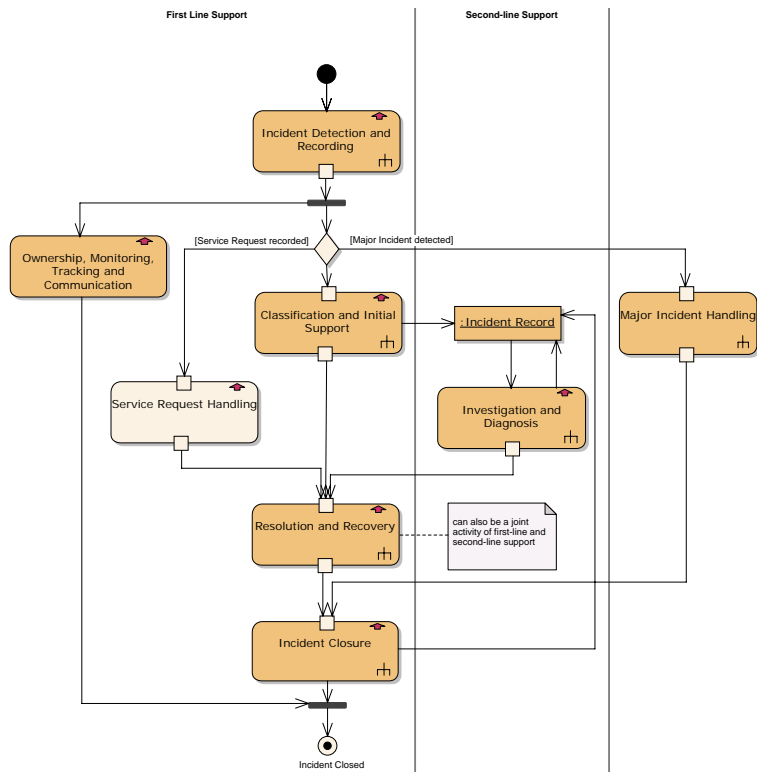
Owner View / Data

Abbildung A.3: Für die anderen Prozesse relevante zentrale Entitäten und Dokumente im *Service Level Management*









Owner View /  
Function /  
Incident Management

Abbildung A.7: Incident Management – Gesamtprozess



Owner View /  
Function /  
Incident  
Management

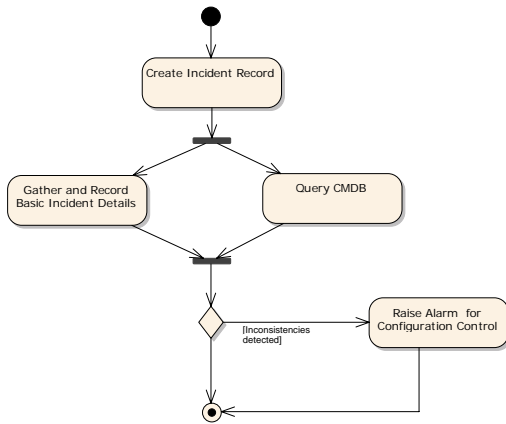


Abbildung A.8: Incident Management – Incident Detection and Recording





Owner View /  
 Function /  
 Incident  
 Management

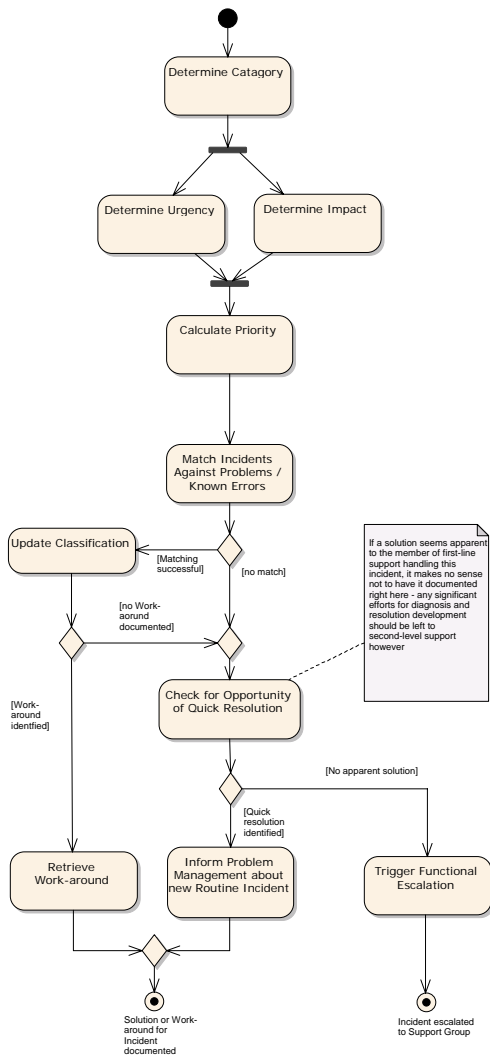


Abbildung A.9: Incident Management – Classification and Initial Support



Owner View /  
 Function /  
 Incident  
 Management

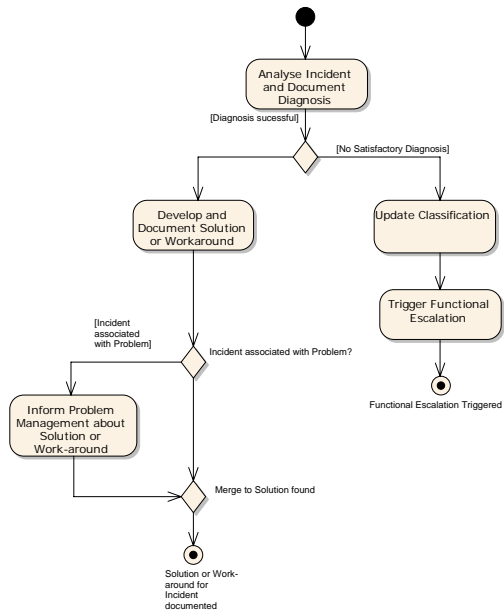


Abbildung A.10: Incident Management – Investigation and Diagnosis



Owner View /  
Function /  
Incident  
Management

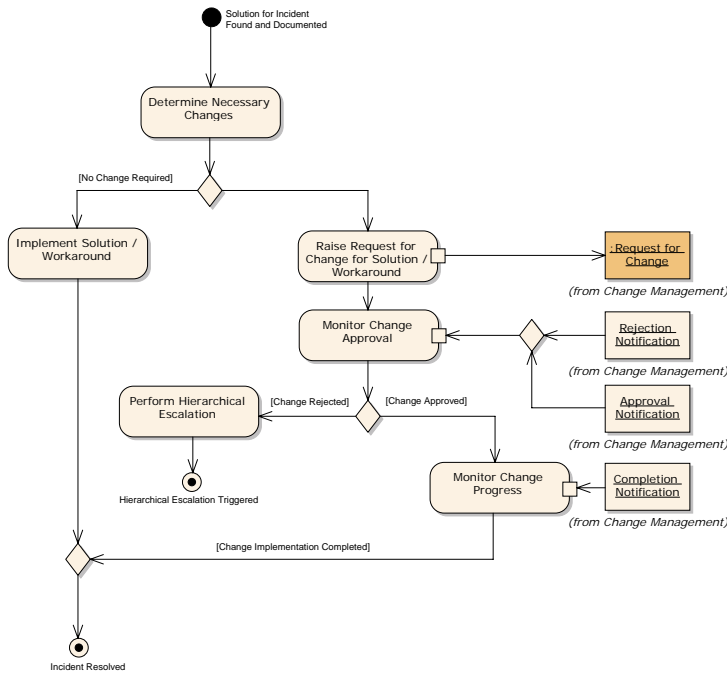


Abbildung A.11: Incident Management – Resolution and Recovery



Owner View /  
Function /  
Incident  
Management

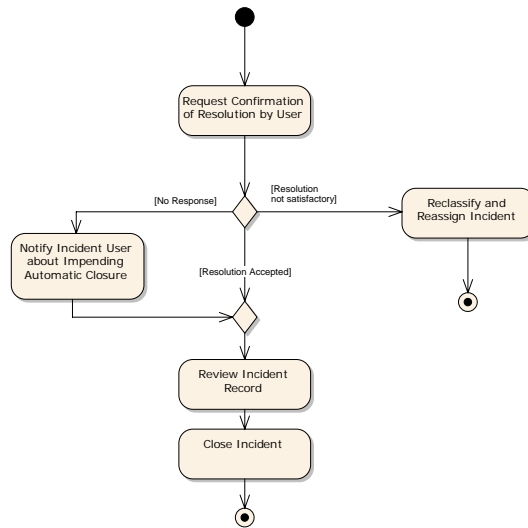
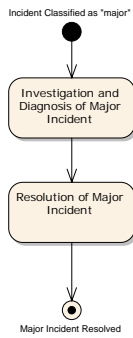


Abbildung A.12: Incident Management – Incident Closure



Owner View /  
 Function /  
 Incident  
 Management

Abbildung A.13: Incident Management – Major Incident Handling



Owner View /  
Function /  
Problem  
Management

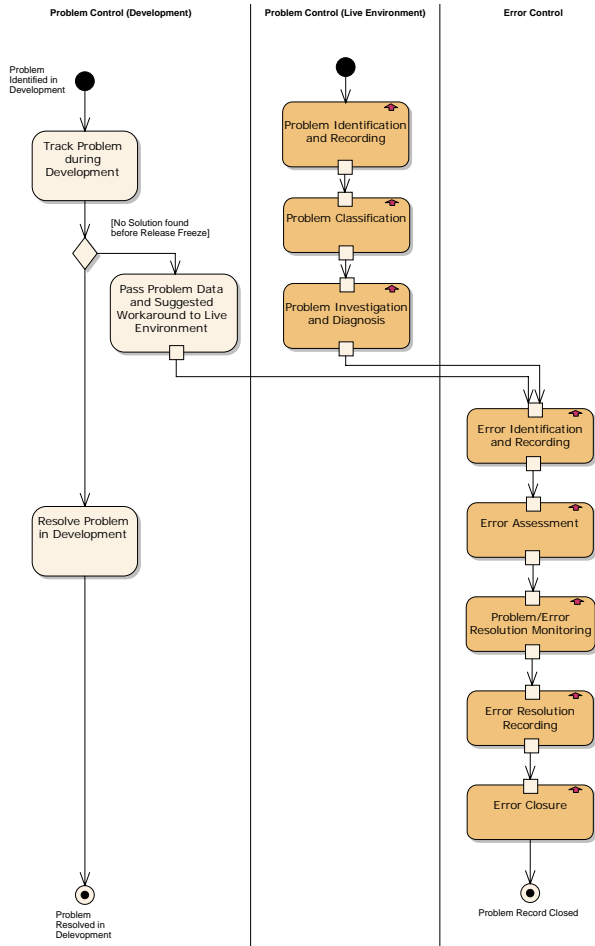


Abbildung A.14: Problem Management – Gesamtprozess (ohne „Proactive Problem Management“)



Owner View /  
Function /  
Change  
Management

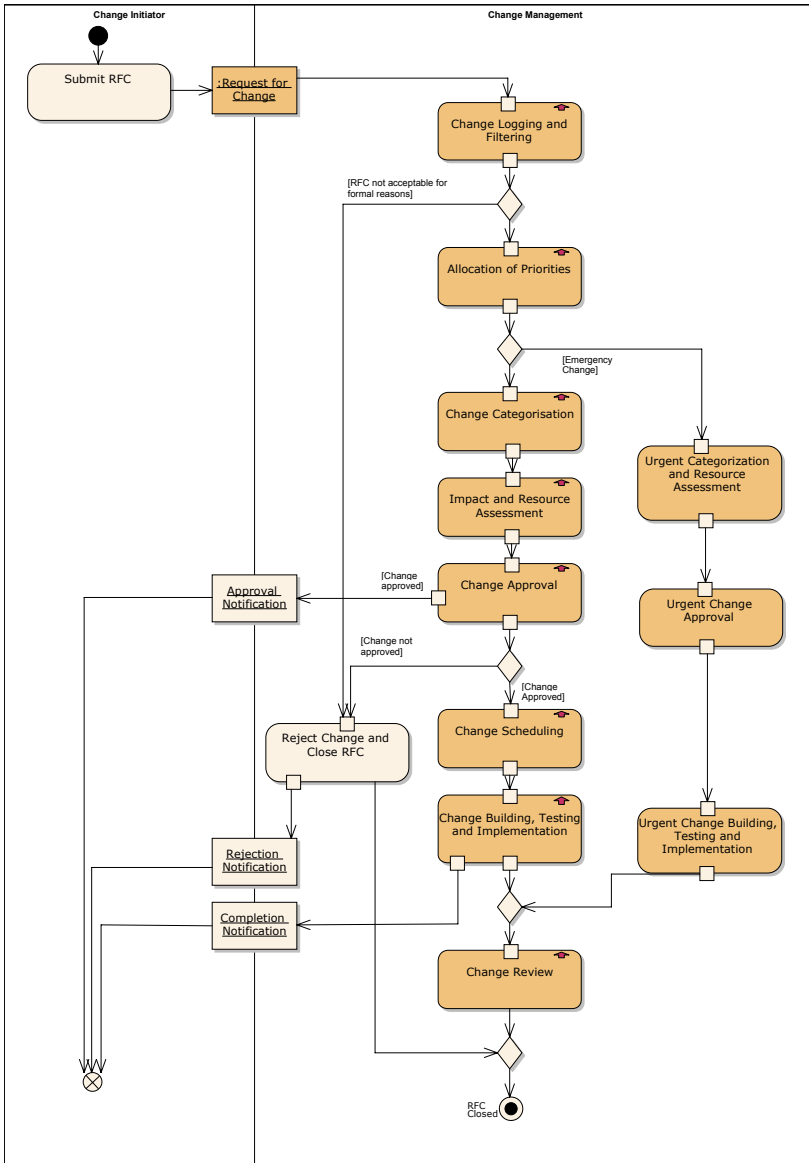


Abbildung A.15: Change Management – Gesamtprozess

## A Modelle

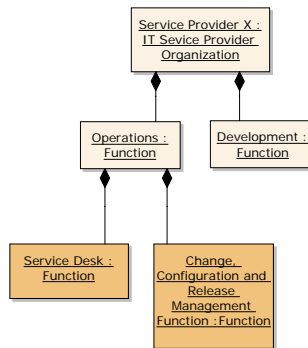
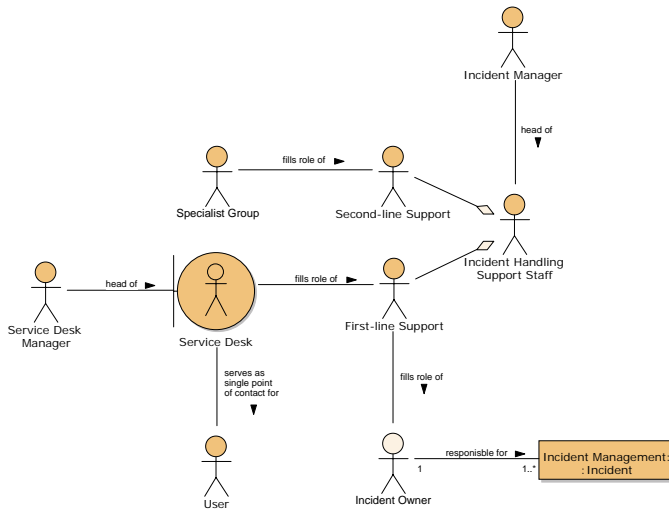


Abbildung A.16: Beispielhafte Einordnung der in ITIL definierten „Functions“





Owner View /  
 People /  
 Incident  
 Management

Abbildung A.17: Incident Management – zentrale Rollen und Funktionsbereiche

# A Modelle



Owner View /  
People /  
Problem  
Management

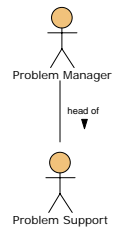
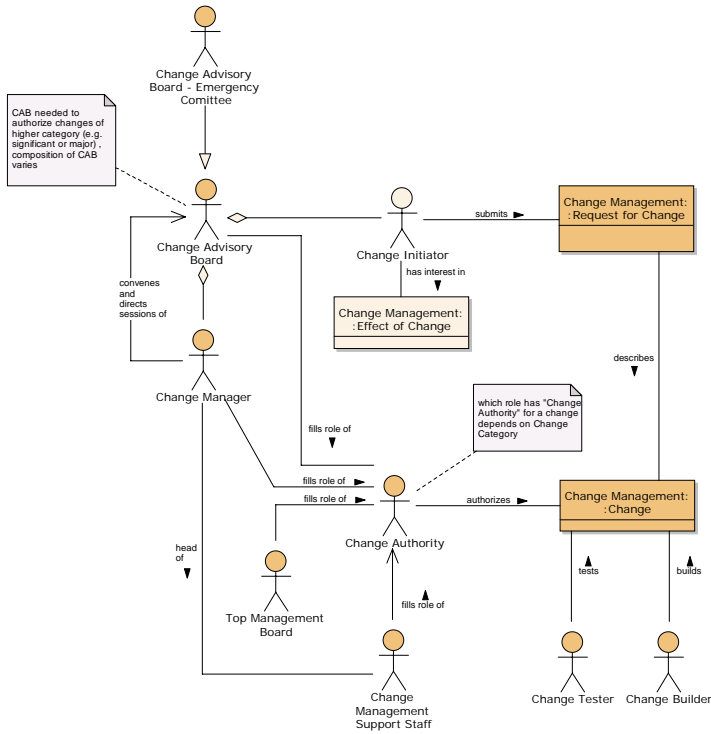


Abbildung A.18: Problem Management – zentrale Rollen



Owner View /  
 People /  
 Change Management

Abbildung A.19: Change Management – zentrale Rollen

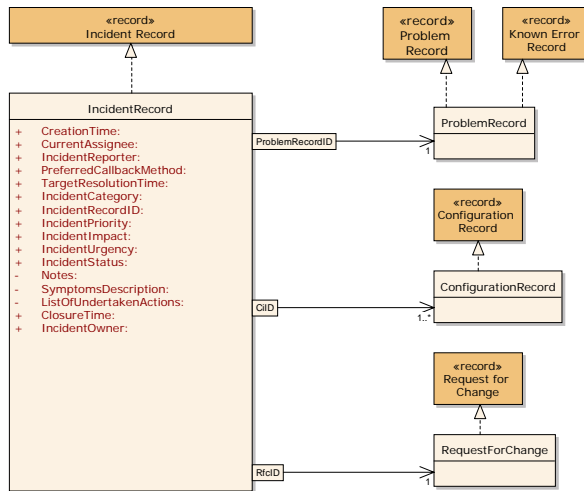
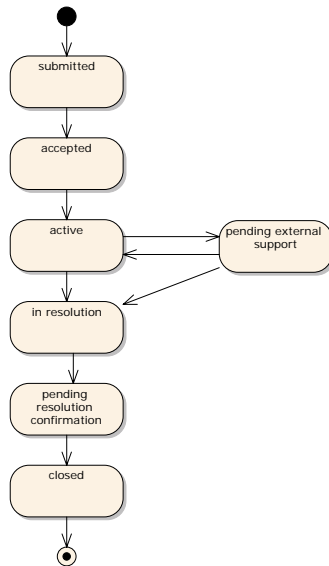


Abbildung A.20: Incident Management – grundlegendes Datenmodell für eine Realisierung eines *Incident Record*



Designer View /  
Data /  
Incident  
Management

Abbildung A.21: Incident Management – Status eines *Incident Record* (mit erlaubten Übergängen)



Designer View /  
Function /  
Incident  
Management

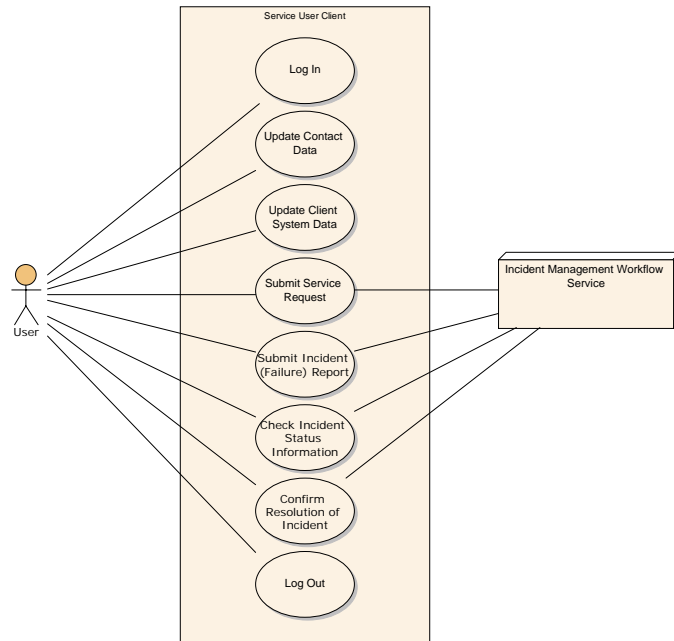


Abbildung A.22: Incident Management Nutzfalldiagramm für einen Client für User (Service User Client)

Identifier	Name
Inc.UC-6	Submit Incident Report (via Service User Client)
Description	
User reports a new Incident via the Service User Client	
Actors	
User, Service User Client (System), Incident Management Workflow Service	
Trigger	
User brings up form for failure report (incident submission)	
Preconditions	
User logged in to Service User Client	
Standard Path	Extensions and Variations
1. User completes data for failure report 2. User submits report 3. Service User Client forwards report to Incident Management Workflow Service 4. Incident Management Workflow Service returns reference data (incident id, target resolution time, ...) 5. Service User Client displays reference data for User	
End Conditions	
Incident Management Workflow Service: new process instance (and Incident Record) created Service User Client: incident added to list of open incidents	



Designer View /  
Function /  
Incident  
Management

Abbildung A.23: Incident Management – Nutzfalldbeschreibung für *Submit Incident (Failure) Report* (vgl. Abbildung A.22)



Designer View /  
Network  
Incident  
Management

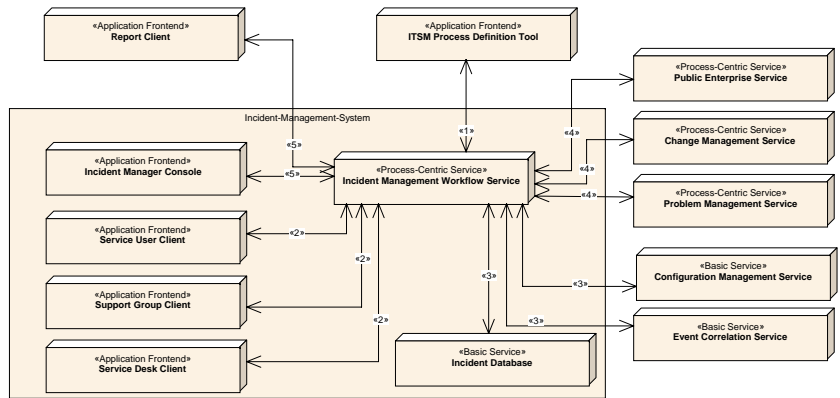
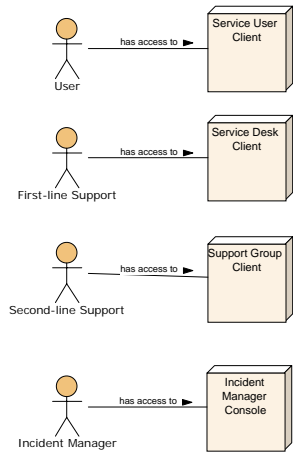


Abbildung A.24: Incident Management – Interfaces des Incident-Management-Systems im Sinne des Workflow Reference Model





Designer View /  
 People /  
 Incident  
 Management

Abbildung A.25: Incident Management – Zugriff auf Workflow Clients



Builder View /  
Data /  
Incident  
Management

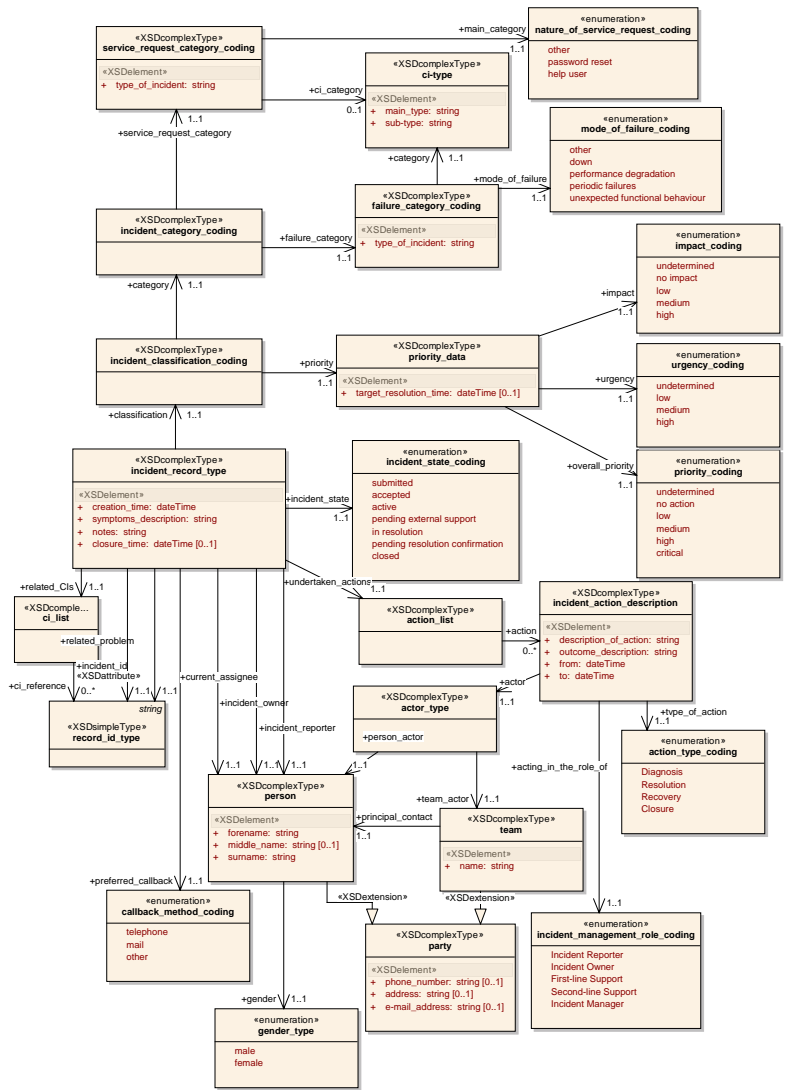


Abbildung A.26: Incident Management – Definition der Datentypen in einem XML Schema

```

<?xml version="1.0"?>
<Incident_Record incident_id="inc-2007-23777"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="http://www.example.com/incident_management_schema.xsd">
  <creation_time>2007-07-27T11:30:00+01:00</creation_time>
  <incident_reporter>
    <phone_number>+49-89-5555-7777</phone_number>
    <e-mail_address>jane.doe@example.com</e-mail_address>
    <forename>jane</forename>
    <surname>doe</surname>
    <gender>female</gender>
  </incident_reporter>
  <preferred_callback>telephone</preferred_callback>
  <incident_owner>
    <phone_number>+49-89-4444-8888</phone_number>
    <e-mail_address>john.doe@servicedesk.example.com</e-mail_address>
    <forename>John</forename>
    <surname>Doe</surname>
    <gender>male</gender>
  </incident_owner>
  <incident_state>accepted</incident_state>
  <classification>
    <category>
      <failure_category>
        <type_of_incident>failure</type_of_incident>
        <category>
          <main_type>Client System</main_type>
          <sub-type>Windows XP-Pro Laptop Type 1</sub-type>
        </category>
        <mode_of_failure>down</mode_of_failure>
      </failure_category>
    </category>
    <priority>
      <overall_priority>medium</overall_priority>
      <target_resolution_time>2007-07-28T11:30:00+01:00</target_resolution_time>
      <impact>low</impact>
      <urgency>medium</urgency>
    </priority>
  </classification>
  <symptoms_description>System hangs after BIOS startup screen, displays "Keyboard
error or no keyboard present"</symptoms_description>
  <related_problem>pro-2007-67</related_problem>
  <related_CIs>
    <ci_reference>con-cl-lpl-2468</ci_reference>
  </related_CIs>
  <undertaken_actions>
  </undertaken_actions>
  <notes></notes>
</Incident_Record>

```



Builder View /  
Data /  
Incident  
Management

Abbildung A.27: Incident Management – Beispiel für *Incident Record*, vgl. Abbildung A.26

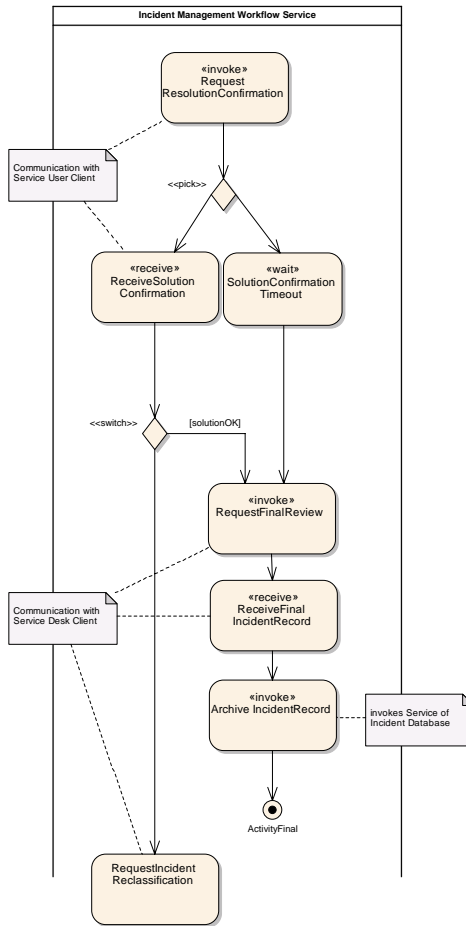
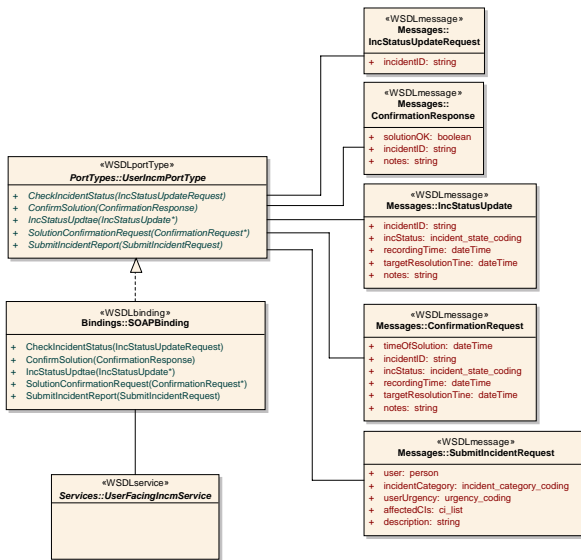


Abbildung A.28: Incident Management – BPEL-Workflow als Aktivitätsdiagramm



Builder View /  
Network /  
Incident  
Management

Abbildung A.29: Incident Management – WSDL-Elemente für die Schnittstelle zum *Service User Client*



## Literaturverzeichnis

- [ACD<sup>+</sup>03] ANDREWS, TONY, FRANCISCO CURBERA, HITESH DHOLAKIA, YARON GOLAND, JOHANNES KLEIN, FRANK LEYMAN, KEVIN LIU, DIETER ROLLER, DOUG SMITH, SATISH THATTE, IVANA TRICKOVIC und SANJIVA WEERAWARANA: *Business Process Execution Language for Web Services*. Specification, BEA and IBM and Microsoft and SAP AG and Siebel Systems, Mai 2003. Version 1.1, <http://tinyurl.com/3xntz3>.
- [AGGI03] AMSDEN, JIM, TRACY GARDNER, CATHERINE GRIFFIN und SRIDHAR IYENGAR: *Draft UML 1.4 Profile for Automated Business Processes with a mapping to BPEL 1.0*. Technischer Bericht Version 1.1, IBM, 2003. <http://tinyurl.com/2pbh4n>.
- [All97] ALLEE, VERNA: *The Knowledge Evolution – Expanding Organizational Intelligence*. Butterworth-Heinemann, 1997.
- [And00] ANDERSEN, BJØRN: *Enterprise Modeling for Business Process Improvement*. In: ROLSTADÅS, ASBJØRN und BJØRN ANDERSEN (Herausgeber): *Enterprise Modeling – Improving Global Industrial Competitiveness*, Band 560 der Reihe *The International Series in Engineering and Computer Science*, Seiten 137–158. Springer, 2000.
- [Apf00] APFEL, AUDREY L.: *ASP Six-Layer Model*. Research Note M-12-1321, Gartner, November 2000.
- [ASL05] ASL FOUNDATION: *ASL in the Forest of Models*, November 2005. <http://tinyurl.com/341qsm>.
- [BADK02] BECKER, JÖRG, LARS ALGERMISSEN, PATRICK DELFMANN und RALF KNACKSTEDT: *Referenzmodellierung*. Das Wirtschaftsstudium (WISU), 31:1392–1395, November 2002.
- [BBB03] BERNHARD, MARTIN G., ROLAND BLOMER und JÜRGEN BONN (Herausgeber): *Strategisches IT-Management*. Symposium Publishing, 2003.
- [BD07] BUTLER, JOHN und JOHN DODD: *In Search of a Common SOA Language*. Technischer Bericht, CBDi, Januar 2007.

- [BDRH<sup>+</sup>07] BRENNER, MICHAEL, GABI DREO-RODOSOK, ANDREAS HANEMANN, HEINZ-GERD HEGERING und RALF KÖNIG: *Service Provisioning - Challenges, Process Alignment and Tool Support*. In: BERGSTRA, J. und M. BURGESS (Herausgeber): *Handbook of Network and System Administration*. Elsevier, 2007. to appear.
- [BE94] BOGAN, CHRISTOPHER E. und MICHAEL J. ENGLISH: *Benchmarking for Best Practices – Winning Through Innovative Adaptation*. McGraw-Hill, 1994.
- [BGN06] BRENNER, MICHAEL, MARKUS GARSCHHAMMER und FRIEDERIKE NICKL: *Requirements Engineering und IT Service Management - Ansatzpunkte einer integrierten Sichtweise*. In: MAYR, H. C. und R. BREU (Herausgeber): *Proceedings der Modellierung 2006*, Band 82 der Reihe *Lecture Notes in Informatics - LNI*, Seiten 51–66. Gesellschaft für Informatik (GI), 2006.
- [BGSS06] BRENNER, MICHAEL, MARKUS GARSCHHAMMER, MARTIN SAILER und THOMAS SCHAAF: *CMDB - Yet Another MIB? On Reusing Management Model Concepts in ITIL Configuration Management*. In: STATE, RADU, SVEN VAN DER MEER, DECLAN O’SULLIVAN und TOM PFEIFER (Herausgeber): *Large Scale Management of Distributed Systems – 17th IFIP/IEEE International Workshop on Distributed Systems, Operations and Management, DSOM 2006*, Band 4269/2006 der Reihe *Lecture Notes in Computer Science*, Seiten 269–280. Springer, 2006.
- [BK03] BECKER, JÖRG und DIETER KAHN: *Der Prozess im Fokus*. In: BECKER, JÖRG et al. [BKR03], Seiten 3–16.
- [BKR03] BECKER, JÖRG, MARTIN KUGELER und MICHAEL ROSEMAN (Herausgeber): *Prozessmanagement – Ein Leitfaden zur prozessorientierten Organisationsgestaltung*. Springer, 2003.
- [Bla92] BLACK, UYLESS: *Network Management Standards - The OSI, SNMP and CMOL Protocols*. McGraw-Hill, 1992.
- [Blu04] BLUM, RICK: *IT Service Management and ITIL*. IT Industry Survey, International Network Services (INS), November 2004. <http://www.ins.com/resources/surveys/>.
- [Blu06] BLUM, RICK: *IT Infrastructure Library*. IT Industry Survey, International Network Services (INS), März 2006. <http://www.ins.com/resources/surveys/>.
- [BM03] BECKER, JÖRG und VOLKER MEISE: *Strategie und Ordnungsrahmen*. In: BECKER, JÖRG et al. [BKR03], Seiten 107–157.



- [Bre06] BRENNER, MICHAEL: *Classifying ITIL Processes - A Taxonomy under Tool Support Aspects*. In: *Proceedings of First IEEE/IFIP International Workshop on Business-Driven IT Management (BDIM 06)*, Seiten 19–28. IEEE, April 2006.
- [BS00] BORGHOFF, UWE M. und JOHANN H. SCHLICHTER: *Computer-Supported Cooperative Work*. Springer, 2000.
- [BS04] BARTOLINI, CLAUDIO und MATHIAS SALLE: *Business Driven Prioritization of Service Incidents*. In: *Proceedings of the 15th IFIP/IEEE International Workshop on Distributed Systems: Operations and Management (DSOM 2004)*, Davis, CA, USA, November 2004. IFIP/IEEE, Springer.
- [Buh05] BUHL, ULRIKE: *ITIL Praxisbuch - Beispiele und Tipps für die erfolgreiche Prozessoptimierung*. mitp, 2005.
- [Cam89] CAMP, ROBERT C.: *Benchmarking - The Search for Industry Best Practices that Lead to Superior Performance*. ASQC Quality Press, 1989.
- [Cap07] CAPGEMINI: *Studie IT-Trends 2007*, 2007. <http://tinyurl.com/2tr3b2>.
- [CFSD90] CASE, JEFFREY D., MARK S. FEDOR, MARTIN L. SCHOFFSTALL und JAMES R. (CHUCK) DAVIN: *Simple Network Management Protocol (SNMP)*. IAB, Mai 1990. RFC 1157.
- [CKS03] CHRISIS, MARY BETH, MIKE KONRAD und SANDY SHRUM: *CM-MI - Guidelines for Process Integration and Product Development*. Addison-Wesley, Boston, 2003.
- [Cla06] CLAUSS, CHRISTIAN: *Entwicklung und Anwendung einer Methodik zur Verfeinerung von ITIL Prozessbeschreibungen am Beispiel des ITIL Change Managements*. Diplomarbeit, Ludwig-Maximilians-Universität München, November 2006. <http://tinyurl.com/39nnyd>.
- [Dep04] DEPARTMENT OF TRADE AND INDUSTRY: *Standards Matters in terms of the Standards Act, 1993 (Act No 29 of 1993)*. In: *Government Gazette No 26877*. Government Printer, Pretoria, South Africa, Oktober 2004.
- [Dij82] DIJKSTRA, EDGER W.: *On the Role of Scientific Thought*. In: *Selected Writings on Computing: A Personal Perspective*, Seiten 60–66. Springer, 1982. <http://tinyurl.com/3afdwwj>.

- [DMMP06] DUTOIT, ALLEN H., RAYMOND McCALL, IVAN MISTRİK und BARBARA PAECH: *Rationale Management in Software Engineering*. Springer, 2006.
- [DR02] DREO RODOSEK, GABI: *A Framework for IT Service Management*. Habilitation, Ludwig-Maximilians-Universität München, Juni 2002.
- [Dre95] DREO, GABI: *A Framework for Supporting Fault Diagnosis in Integrated Network and Systems Management: Methodologies for the Correlation of Trouble Tickets and Access to Problem-Solving Expertise*. Dissertation, Ludwig-Maximilians-Universität München, Juli 1995.
- [Dug04] DUGMORE, JENNY: *BS15000: Past, Present and Future*. Service-talk, 04, April 2004. <http://tinyurl.com/y93xgn>.
- [Dug06] DUGMORE, JENNY: *Benchmarking Provision of IT Services*. ISO Focus, Mai 2006.
- [Eco03] THE ECONOMIST: *At your service*, Mai 2003. Survey: The IT Industry.
- [EDJ99] EVANS, JAMES R. und JAMES W. DEAN JR.: *Total Quality: Management, Organization and Strategy*. South-Western College Publishing, 1999.
- [EP98] ERIKSSON, HANS-ERIK und MAGNUS PENKER: *Business Modeling With UML – Business Patterns at Work*. John Wiley & Sons, Inc., New York, NY, USA, 1998.
- [ER06] ERICKSON, JON und SADAF ROSHAN: *The Financial Impact of Operational Process Improvements within a Microsoft Windows Environment – Examining the Total Economic Impact of Microsoft Operations Framework (MOF)*. Forrester Research, Juni 2006. <http://go.microsoft.com/fwlink/?LinkId=76252>.
- [EW03] EXELER, STEFFEN und SVEN WILMS: *Change Management with ARIS*. In: SCHEER, AUGUST-WILHELM et al. [SAJK03], Seiten 23-48.
- [EXI07] EXIN (NATIONAAL EXAMENINSTITUUT VOOR INFORMATICA): *ITIL Certificaten*, 2007. <http://tinyurl.com/36uw61>.
- [Fac02] FACTOR, ALEXANDER L.: *Analyzing Application Service Providers*. Sun Microsystems Press, 2002.
- [FAS04] FAST GESELLSCHAFT FÜR ANGEWANDTE SOFTWARETECHNOLOGIE MBH: *IT Service Management (ITSM) Optimize your IT Processes with ITIL*, 2004. <http://tinyurl.com/32srnt>.

- [FHM<sup>+</sup>03] FRANKEL, DAVID S., PAUL HARMON, JISHNU MUKERJI, JAMES ODELL, MARTIN OWEN, PETE RIVITT, MIKE ROSEN und RICHARD M. SOLEY: *The Zachman Framework and the OMG's Model Driven Architecture*. Business Process Trends, September 2003. <http://tinyurl.com/v9w4c>.
- [Fow97a] FOWLER, MARTIN: *Analysis Patterns — Reusable Object-Models*. Addison-Wesley Professional Computing Series. Addison-Wesley, 1997.
- [Fow97b] FOWLER, MARTIN: *Dealing with Roles*, Juli 1997. <http://www.martinfowler.com/apSUPP/roles.pdf>.
- [GHJV95] GAMMA, ERICH, RICHARD HELM, RALPH JOHNSON und JOHN VLISSIDES: *Design Patterns — Elements of Reusable Object-Oriented Software*. Addison-Wesley Professional Computing Series. Addison-Wesley, 1995.
- [GHK<sup>+</sup>01] GARSCHHAMMER, MARKUS, RAINER HAUCK, BERNHARD KEMPTER, IGOR RADISIC, HARALD ROELLE und HOLGER SCHMIDT: *The MNM Service Model — Refined Views on Generic Service Management*. Journal of Communications and Networks, 3(4):297–306, Dezember 2001. <http://tinyurl.com/yrn9ue>.
- [Gli99] GLINZ, MARTIN: *Eine geführte Tour durch die Landschaft der Software-Prozesse und -Prozessverbesserung*. Informatik/Informatique, 6:7–15, Dezember 1999.
- [GM01] GOVEKAR, MILIND und SIMON MINGAY: *Microsoft 'Piggybacks' ITIL*. Research Note SPA-11-2135, Gartner, Februar 2001.
- [Gra03] GRAWE, TONIO: *Bei den Großen abgeschaut*. In: BERNHARD, MARTIN G. et al. [BBB03], Seiten 365–377.
- [Ham96] HAMMER, MICHAEL: *Das prozeßorientierte Unternehmen – Die Arbeitswelt nach dem Reengineering*. Wilhelm Heyne Verlag, 1996.
- [HAN99] HEGERING, HEINZ-GERD, SEBASTIAN ABECK und BERNHARD NEUMAIR: *Integrated Management of Networked Systems — Concepts, Architectures and their Operational Application*. Morgan Kaufmann Publishers, Januar 1999.
- [Har91] HARRINGTON, H. JAMES: *Business Process Improvement – The Breakthrough Strategy for Total Quality, Productivity, and Competitiveness*. McGraw-Hill Professional, 1991.
- [Har03] HARMON, PAUL: *Business Process Change – A Manager's Guide to Improving, Redesigning, and Automating Processes*. Morgan Kaufmann Publishers, 2003.

- [Has04] HASSENPFUG, PASCAL: *Analyse von Service-Level-Management-Prozessen in der BMW Group*. Diplomarbeit, Ludwig-Maximilians-Universität München, Februar 2004.
- [HC93] HAMMER, MICHAEL und JAMES CHAMPY: *Business Reengineering – Die Radikalkur für das Unternehmen*. Wilhelm Heyne Verlag, 1993.
- [Hei05] HEISERICH, OTTO-ERNST: *Logistik: Eine praxisorientierte Einführung*. Gabler, 2005.
- [HEN01] HUTH, CARSTEN, INGO ERDMANN und LUDWIG NASTANSKY: *GroupProcess: Using Process Knowledge from the Participative Design and Practical Operation of Ad Hoc Processes for the Design of Structured Workflows*. In: *Proceedings of the 34th Hawaii International Conference on System Sciences*, 2001.
- [HHP06a] HYDER, ELAINE B., KEITH M. HESTON und MARK C. PAULK: *The eSourcing Capability Model for Service Providers (eSCM-SP) v2.01, Part 1: Model Overview*. Technischer Bericht CMU-ITSQC-06-006, ITsqc, ISRI, School of Computer Science, Carnegie Mellon University, Dezember 2006. <http://itsqc.cmu.edu/downloads/>.
- [HHP06b] HYDER, ELAINE B., KEITH M. HESTON und MARK C. PAULK: *The eSourcing Capability Model for Service Providers (eSCM-SP) v2.01, Part 2: Practice Details*. Technischer Bericht CMU-ITSQC-06-007, ITsqc, ISRI, School of Computer Science, Carnegie Mellon University, Dezember 2006. <http://itsqc.cmu.edu/downloads/>.
- [HKK98] HIEBELER, ROBERT, CHARLES KETTEMAN und THOMAS B. KELLY: *Best Practices : Building Your Business With Customer-Focused Solutions*. Simon and Schuster, 1998.
- [HKM<sup>+</sup>02] HYDER, ELAINE B., BENNET KUMAR, VIVEK MAHENDRA, JANE SIEGEL, KEITH M. HESTON, RAJESH GUPTA, HAHEEB MAHABOOB und PALANIVLARJAN SUBRAMANIAN: *eSourcing Capability Model (eSCM) for IT-enabled Service Providers v1.1*. Technischer Bericht CMU-CS-02-115, School of Computer Science, Carnegie Mellon University, Oktober 2002. <http://itsqc.cmu.edu/downloads/>.
- [HL06a] HEFLEY, WILLIAM E. und ETHEL A. LOESCHE: *The eSourcing Capability Model for Client Organizations (eSCM-CL) v1.1, Part 1: Model Overview*. Technischer Bericht CMU-ITSQC-06-002, ITsqc, ISRI, School of Computer Science, Carnegie Mellon University, September 2006. <http://itsqc.cmu.edu/downloads/>.

- [HL06b] HEFLEY, WILLIAM E. und ETHEL A. LOESCHE: *The eSourcing Capability Model for Client Organizations (eSCM-CL) v1.1, Part 2: Practice Details*. Technischer Bericht CMU-ITSQC-06-003, ITSqc, ISRI, School of Computer Science, Carnegie Mellon University, September 2006. <http://itsqc.cmu.edu/downloads/>.
- [Hol95] HOLLINGSWORTH, DAVID: *The Workflow Reference Model*. WPMC (Workflow Management Coalition), Januar 1995. <http://tinyurl.com/y5pjb>.
- [Hup06] HUPPERTZ, PAUL G.: *IT-Service - Der Kern des Ganzen*, Band 6 der Reihe *IT Service Management Advanced Pocket Book*. Servview GmbH, April 2006.
- [HV93] HENDERSON, J. C. und N. VENKATRAMAN.: *Strategic Alignment - Leveraging Information Technology for Transforming Organizations*. IBM Systems Journal, 32(1), 1993.
- [HZB04] HOCHSTEIN, AXEL, RÜDIGER ZARNEKOW und WALTER BRENNER: *ITIL als Common-Practice-Referenzmodell für das IT-Service-Management - Formale Beurteilung und Implikationen für die Praxis*. Wirtschaftsinformatik, 46(5):382-389, 2004.
- [HZB05] HOCHSTEIN, AXEL, RÜDIGER ZARNEKOW und WALTER BRENNER: *ITIL as Common Practice Reference Model for IT Service Management: - Formal Assessment and Implications for Practice*. In: *Proceedings of the 2005 IEEE International Conference on E-Technology, E-Commerce and E-Service*, Hong Kong, März 2005.
- [IBM07] IBM: *IBM Tivoli Unified Process*, 2007. v2.1.3., <http://tinyurl.com/3daq59>.
- [IDG<sup>+</sup>04] IQBAL, MAJID, JENNY DUGMORE, SUBRATA GUHA, WILLIAM E. HEFLEY, ELAINE B. HYDER und MARK C. PAULK: *Comparing the eSCM-SP v2 and BS 15000 - A comparison between the eSourcing Capability Model for Service Providers v2 and BS 15000-1:2002 (IT Service Management)*. Technischer Bericht CMU-ISRI-04-129b, Institute for Software Research International, Carnegie Mellon University, Oktober 2004. <http://itsqc.cmu.edu/downloads/>.
- [Inf06] INFORM-IT: *Tool Selector*, 2006. <http://tools.itsmportal.net>.
- [ISO05a] ISO - INTERNATIONAL ORGANIZATION FOR STANDARDIZATION: *ISO/IEC 20000 benchmarks provision of IT service management*, Dezember 2005. Press Release, Ref.: 985.

- [ISO05b] ISO/IEC: *Information Technology - Service Management - Part 1: Specification*, Dezember 2005. International Standard ISO/IEC 20000-1:2005.
- [ISO05c] ISO/IEC: *Information Technology - Service Management - Part 2: Code of Practice*, Dezember 2005. International Standard ISO/IEC 20000-1:2005.
- [IT96] ITU-T: *Principles for a Telecommunications management network - M.3010*. <http://www.itu.int/>, Mai 1996.
- [ITG00] ITGI (IT GOVERNANCE INSTITUTE): *CobiT 3rd Edition – Control Objectives*, Juli 2000.
- [ITG05a] ITGI AND OGC: *Aligning COBIT, ITIL and ISO 17799 for Business Benefit – A Management Briefing from ITGI and OGC*, 2005. <http://tinyurl.com/ypty5r>.
- [ITG05b] ITGI (IT GOVERNANCE INSTITUTE): *CobiT 4.0*, 2005. <http://www.isaca.org/Bookstore/>.
- [ITS04] ITSMF DEUTSCHLAND (Herausgeber): *IT Service Management – Eine Einführung basierend auf ITIL*. Van Haren Publishing, 2004. (Redaktion: Jan van Bon, Annelies van der Veen, Mike Pieper).
- [ITU92] ITU-T: *Data Communication Networks – Management Framework for Open Systems Interconnection (OSI) for CCITT Applications*, September 1992. Recommendation X.700.
- [ITU95] ITU-T: *Information Technology – Open Distributed Processing – Reference Model: Architecture*, November 1995. Recommendation X.903.
- [ITU97] ITU-T: *OSI Networking and System Aspects – Quality of Service*, Dezember 1997. Recommendation X.641.
- [Kay03] KAYE, DOUG: *Loosely Coupled – The Missing Pieces of Web Services*. RDS Press, April 2003.
- [KBS05] KRAFZIG, DIRK, KARL BANKE und DIRK SLAMA: *Enterprise SOA – Service-Oriented Architecture Best Practices*. Pearson, 2005.
- [Kel05] KELLER, ALEXANDER: *Automating the Change Management Process with Electronic Contracts*. In: *Proceedings of the Seventh IEEE International Conference on E-Commerce Technology Workshops (CECW 05)*, Seiten 99–107, Juli 2005.

- [KKS01] KELLER, ALEXANDER, HEATHER KREGER und KARL SCHOPMEYER: *Towards a CIM Schema for Run Time Application management*. In: *Proceedings of the 12th IFIP/IEEE International Workshop on Distributed Systems: Operatios & Management (DSOM 2001)*. IEEE, Oktober 2001.
- [Kni06] KNITTL, SILVIA: *Entwicklung eines Konzepts zur plattformübergreifenden Integration des Release-Managements bei der HVB-Info*. Diplomarbeit, Ludwig-Maximilians-Universität München, März 2006.
- [Koy07] KOYCHEVA, GALINA: *Evaluierung von BPEL-Implementierungen für IT-Service-Management-Prozesse*. Fortgeschrittenenpraktikum, Ludwig-Maximilians-Universität München, Februar 2007.
- [Köh05] KÖHLER, PETER T.: *ITIL*. Springer, Berlin Heidelberg, Januar 2005.
- [LBN02] LEE, JOHN J. und RON BEN-NATAN: *Integrating Service Level Agreements – Optimizing Your OSS for SLA Delivery*. Wiley, 2002.
- [Lew99] LEWIS, LUNDY: *Service Level Management for Enterprise Networks*. Artech House, 1999.
- [Lin03] LINTHICUM, DAVID S.: *Next Generation Application Integration: From Simple Information to Web Services*. Addison-Wesley Professional, 2003.
- [Mar04] MARKUS, M. LYNNE: *Technochange Management: Using IT to Drive Organizational Change*. *Journal of Information Technology*, 19(1):4–20, 2004.
- [MAS<sup>+</sup>03] MCGOVERN, JAMES, SCOTT W. AMBLER, MIKE STEVENS, JAMES LINN, VIKAS SHARAN und ELIAS K. JO: *A Practical Guide to Enterprise Architecture*. Prentice Hall, 2003.
- [McC01] MCCORMACK, KEVIN P.M JOHNSON, WILLIAM C.: *Business Process Orientation*. CRC Press, 2001.
- [MD96] MELNYK, STEVEN A. und DAVID R. DENZLER: *Operations Management – A Value-Driven Approach*. McGraw Hill, 1996.
- [MG02] MINGAY, SIMON und MILIND GOVEKAR: *Effects of the British Standard for IT Service Management*. Research Note SPA-13-3434, Gartner, März 2002.
- [Mic04] MICROSOFT COOPERATION: *MOF Executive Overview*, April 2004. <http://tinyurl.com/ygttds>.

- [Mic05] MICROSOFT COOPERATION: *Microsoft Operations Framework – Service Management Functions – Incident Management*, 2005. <http://tinyurl.com/yoocd8>.
- [Mis04] MISRA, KUNDAN: *OSS for Telecom Networks – An Introduction to Network Management*. Springer, Juni 2004.
- [Nat96] NATIONAL COMMUNICATIONS SYSTEM TECHNOLOGY AND STANDARDS DIVISION: *Federal Standard 1037C: Telecommunications – Glossary of Telecommunication Terms*. General Services Administration Information Technology Service, USA, August 1996. <http://www.its.bldrdoc.gov/fs-1037/fs-1037c.htm>.
- [NCTvV05] NIESSINK, FRANK, VIKTOR CLERC, TON TIJDINK und HANS VAN VLIET: *The IT Service Capability Maturity Model*. Version 1.0, Release Candidate 1, Department of Computer Science, Faculty of Sciences, Vrije Universiteit, Januar 2005. <http://www.itservicecmm.org/download.html>.
- [Nex00] NEXTSLM.ORG: *Service Level Management - North American Survey 2000*, 2000. <http://www.nextslm.org/SLMSurvey2000.pdf>.
- [NPW03] NEUMANN, STEFAN, CHRISTIAN PROBST und CLEMENS WERNSMANN: *Kontinuierliches Prozessmanagement*. In: BECKER, JÖRG et al. [BKR03], Seiten 309–335.
- [OGC99] OGC (OFFICE OF GOVERNMENT COMMERCE) (Herausgeber): *Security Management*. IT Infrastructure Library. The Stationary Office, Norwich, UK, 1999.
- [OGC00] OGC (OFFICE OF GOVERNMENT COMMERCE) (Herausgeber): *Service Support*. IT Infrastructure Library. The Stationary Office, Norwich, UK, 2000.
- [OGC01] OGC (OFFICE OF GOVERNMENT COMMERCE) (Herausgeber): *Service Delivery*. IT Infrastructure Library. The Stationary Office, Norwich, UK, 2001.
- [OGC02a] OGC (OFFICE OF GOVERNMENT COMMERCE) (Herausgeber): *Application Management*. IT Infrastructure Library. The Stationary Office, Norwich, UK, 2002.
- [OGC02b] OGC (OFFICE OF GOVERNMENT COMMERCE) (Herausgeber): *ICT Infrastructure Management*. IT Infrastructure Library. The Stationary Office, Norwich, UK, 2002.
- [OGC02c] OGC (OFFICE OF GOVERNMENT COMMERCE) (Herausgeber): *Planning to Implement Service Management*. IT Infrastructure Library. The Stationary Office, Norwich, UK, 2002.



- [OGC03] OGC (OFFICE OF GOVERNMENT COMMERCE) (Herausgeber): *Software Asset Management*. IT Infrastructure Library. The Stationary Office, Norwich, UK, 2003.
- [OGC04] OGC (OFFICE OF GOVERNMENT COMMERCE) (Herausgeber): *Business Perspective: The IS View on Delivering Services to the Business*. IT Infrastructure Library. The Stationary Office, Norwich, UK, 2004.
- [OGC05a] OGC (OFFICE OF GOVERNMENT COMMERCE) (Herausgeber): *Introduction to ITIL*. IT Infrastructure Library. The Stationary Office, 2005.
- [OGC05b] OGC (OFFICE OF GOVERNMENT COMMERCE): *ITIL Refresh – Results of public consultations*, April 2005.
- [OGC06a] OGC (OFFICE OF GOVERNMENT COMMERCE) (Herausgeber): *ITIL Small-scale Implementation*. IT Infrastructure Library. The Stationary Office, Norwich, UK, 2006.
- [OGC06b] OGC (OFFICE OF GOVERNMENT COMMERCE) (Herausgeber): *Managing Successful Projects with PRINCE2*. The Stationary Office, 3 Auflage, 2006.
- [OGC07] OGC (OFFICE OF GOVERNMENT COMMERCE): *IT Infrastructure Library (ITIL) – Frequently Asked Questions*, April 2007. <http://www.itil.co.uk/faqs.htm>.
- [OMG03] OMG (OBJECT MANAGEMENT GROUP): *MDA Guide*, Juni 2003. Version 1.0.1, omg/2003-06-01, <http://tinyurl.com/3am9uz>.
- [OMG06] OMG (OBJECT MANAGEMENT GROUP): *Business Process Modeling Notation Specification – BPMN 1.0*, Februar 2006. Final Adopted Specification, dtc/06-02-01, <http://tinyurl.com/2vh3at>.
- [OWS<sup>+</sup>03] OESTEREICH, BERND, CHRISTIAN WEISS, CLAUDIA SCHRÖDER, TIM WEILKIENS und ALEXANDER LENHARD: *Objektorientierte Geschäftsprozessmodellierung mit der UML*. dpunkt Verlag, 2003. <http://www.oogpm.de>.
- [Pin06] PINK ELEPHANT: *IT Service Management Tools – Information Technology Infrastructure Library (ITIL) Compatibility Considerations*, Version 4 Auflage, Januar 2006. <http://tinyurl.com/y8ejcu>.
- [Pin07] PINK ELEPHANT: *PinkVerify Certification Criteria*, April 2007. Webseite, <http://tinyurl.com/3du7da>.

- [PNC00] PANDE, PETER S., ROBERT P. NEUMAN und ROLAND R. CAVANAGH: *The Six Sigma Way – How GE, Motorola, and Other Top Companies are Honing Their Performance*. McGraw-Hill, 2000.
- [PQA03] PULTURAK, DAVE, PETE QUAGLIARIELLO und ROLF AKKER: *Das MOF-Taschenbuch*. Van Haren Publishing, Juli 2003.
- [Pro03] PROBST, CHRISTIAN: *Referenzmodell für IT-Service-Informationssysteme*, Band 3 der Reihe *Advances in Information Systems and Management Science*. Logos Verlag Berlin, 2003.
- [PS06] PRÄNGER, THOMAS und RAINER SCHMIDT: *Die Zukunft der IT Infrastructure Library (ITIL) auf solidem Fundament*. Technischer Bericht, BPM-Guide.de, 2006. <http://www.bpm-guide.de/articles/35>.
- [Pul02] PULTORAK, DAVID: *An Introduction to IT Service Management – ITIL and MOF*. DCM Magazine, September 2002. AFCOM.
- [Put00] PUTMAN, JANIS R.: *Architecting with RM-ODP*. Prentice Hall PTR, 2000.
- [RC05] REILLY, JOHN P. und MARTIN J. CREAMER: *NGOSS Distilled – The Essential Guide to Next Generation Telecoms Management*. The Lean Corporation, August 2005.
- [Red06] REDMONDMAG.COM: *The Best of the Best – Redmond’s 2006 Readers’ Choice Awards*, Mai 2006. <http://tinyurl.com/2etsq5>.
- [Roe05] ROELLE, HARALD: *Eine dienstorientierte Methodik zur Kopplung von Netz-QoS-Architekturen*. Dissertation, Ludwig-Maximilians-Universität München, Juli 2005.
- [RSD03] ROSEMANN, MICHAEL, ANSGAR SCHWEGMANN und PATRICK DELFMANN: *Vorbereitung der Prozessmodellierung*. In: BECKER, JÖRG et al. [BKR03], Seiten 47–105.
- [RtHEvdA04] RUSSELL, NICK, ARTHUR H. M. TER HOFSTEDÉ, DAVID EDMOND und WIL M. P. VAN DER AALST: *Workflow Data Patterns*. Technischer Bericht FIT-TR-2004-01, Queensland University of Technology, April 2004. <http://tinyurl.com/37v7xq>.
- [RtHvdAM06] RUSSELL, NICK, ARTHUR H.M. TER HOFSTEDÉ, WIL M. P. VAN DER AALST und NATALIYA MULAR: *Workflow Control-Flow Patterns: A Revised View*. BPM Center Report BPM-06-22, BPM Center, 2006. <http://tinyurl.com/3xz6wn>.
- [SAJK03] SCHEER, AUGUST-WILHELM, FERRI ABOLHASSAN, WOLFRAM JOST und MATHIAS KIRCHMER (Herausgeber): *Business Process Change Management – Aris in Practice*. Springer, 2003.

- [Sal04] SALLÉ, MATHIAS: *IT Service Management and IT Governance – Review, Comparative Analysis and their Impact on Utility Computing*. Technischer Bericht HPL-2004-98, HP Laboratories Palo Alto, Juni 2004. <http://tinyurl.com/3dajwl>.
- [San06] SANTIFALLER, MICHAEL: *Tivoli auf Kurs zu Service Management*. Computerwoche, 2006(27), 2006.
- [Sch97] SCHEER, AUGUST-WILHELM: *Wirtschaftsinformatik: Referenzmodelle für industrielle Geschäftsprozesse*. Springer, 7 Auflage, 1997. ISBN 3-540-62967-X.
- [Sch01] SCHMIDT, HOLGER: *Entwurf von Service Level Agreements auf der Basis von Dienstprozessen*. Dissertation, Ludwig-Maximilians-Universität München, Juli 2001.
- [Sch04] SCHMIDT, RAINER: *IT-Service-Management - Aktueller Stand und Perspektiven für die Zukunft*. FH Aalen, ITSMF, Dezember 2004. Präsentation der Umfrageergebnisse auf dem 4. ITSMF-Kongress in Hamburg, <http://tinyurl.com/3a4aey>.
- [Sco99] SCOTT, DONNA: *Making Smart Investments to Reduce Unplanned Downtime*. Research Note TG-07-4033, Gartner, März 1999.
- [Sco03] SCOTTISH EXECUTIVE, St. Andrews House, Edinburgh: *Benchmarking and Definitions within Contact Centres*, 2003. <http://tinyurl.com/36j4yy>.
- [SH05] SCHRÖN, BIRGIT und MARTIN HAFNER: *Integration des Prozessmanagements in eine zukunftsorientierte CIO-Organisation am Beispiel von Siemens ICN*. In: SCHELP, JOACHIM und ROBERT WINTER [SW05], Seiten 23–39.
- [She97] SHEARD, SARAH. A.: *The Frameworks Quagmire – A Brief Look*. In: *Proceedings of INCOSE (International Council on Systems Engineering) Conference 1997*, 1997.
- [She01] SHEARD, SARAH. A.: *Evolution of the Frameworks Quagmire*. IEEE Computer, 34(7):96–98, 2001.
- [SL03] SCHWEGMANN, ANSGAR und MICHAEL LASKE: *Istmodellierung und Istanalyse*. In: BECKER, JÖRG et al. [BKR03], Seiten 159–190.
- [SS03] SPECK, MARIO und NORBERT SCHNETGÖKE: *Sollmodellierung und Prozessoptimierung*. In: BECKER, JÖRG et al. [BKR03], Seiten 191–226.
- [SS04] SCHMELZER, HERMANN J. und WOLFGANG SESSELMANN: *Geschäftsprozessmanagement in der Praxis*. Hanser, 2004.

- [SSE03] SSE-CMM PROJECT: *Systems Security Engineering Capability Maturity Model SSE-CMM – Model Description Document (Version 3.0)*, Juni 2003. <http://tinyurl.com/ydh22h>.
- [Sta04] STANDARDS AUSTRALIA: *ICT Service Management – Part 1: Specification for Service Management*, 2004. Australian Standard AS 8018.1–2004.
- [Sti99] STIENEN, HANS: *Nach CMM und BOOTSTRAP: SPiCE – Die neue Norm für Prozessbewertungen*. Informatik/Informatique, 6:16–22, Dezember 1999.
- [Stö05] STÖRRLE, HARALD: *UML 2 erfolgreich einsetzen. Einführung und Referenz (Programmer’s Choice)*. Addison-Wesley, 2005.
- [Sun01] SUN MICROSYSTEMS: *SunTone Architecture Methodology – A 3-Dimensional Approach to Architectural Design*, 2001.
- [SW05] SCHELP, JOACHIM und ROBERT WINTER (Herausgeber): *Auf dem Weg zur Integration Factory – Proceedings der DW2004 – Data Warehousing und EAI*. Springer, 2005.
- [Teu01] TEUFEL, STEPHANIE: *ICT-Management Framework*. In: *Proceedings of the Conference on Advances in Infrastructure for Electronic Business, Science and Education on the Internet (SSGRR 2001)*, L’Aquila, Italy, 2001.
- [TMF00] TMF (TELEMANAGEMENT FORUM): *Telecom Operations Map*, März 2000. GB910, Version 2.1.
- [TMF04a] TMF (TELEMANAGEMENT FORUM): *enhanced Telecom Operations Map (eTOM)*, November 2004. GB921, Version 4.5.
- [TMF04b] TMF (TELEMANAGEMENT FORUM): *eTOM – Public B2B Business Operations Map (BOM) Application Note C – An initial proposal for the scope and structure of ICT Business Transactions*, 2004. GB921C, Version 4.0.
- [TMF04c] TMF (TELEMANAGEMENT FORUM): *NGOSS Compliance Testing Strategy – Technical Specification*, 2004. TMF 050, Version 4.1.
- [TMF04d] TMF (TELEMANAGEMENT FORUM): *The NGOSS Technology-Neutral Architecture*, 2004. TMF 053, Version 3.1.
- [TMF04e] TMF (TELEMANAGEMENT FORUM): *Shared Information/Data (SID) Model – Concepts, Principles, and Domains*, Release 4.5, Version 5.1.1 Auflage, November 2004. GB922.

- [TMF04f] TMF (TELEMANAGEMENT FORUM): *Telecom Operations Map (eTOM) – The Business Process Framework – Addendum D: Process Decompositions and Descriptions*, 2004. GB921D, Version 4.0.
- [TMF04g] TMF (TELEMANAGEMENT FORUM): *Telecom Operations Map (eTOM) – The Business Process Framework – Addendum F: Process Flow Examples*, 2004. GB921F, Version 4.0.
- [TMF04h] TMF (TELEMANAGEMENT FORUM): *Telecom Operations Map (eTOM) – The, Business Process Framework – eTOM-ITIL Application Note – Using eTOM to model the ITIL Processes*, 2004. GB921L, Version 4.0.
- [vB05] BON, JAN VAN (Herausgeber): *Foundations of IT Service Management based on ITIL*. Van Haren Publishing, 2005.
- [vdP04] POLS, REMKO VAN DER: *ASL – A Framework for Application Management*. Van Haren Publishing, 2004.
- [vdPDvO05] POLS, REMKO VAN DER, RALPH DONATZ und FRANK VAN OUTHORST: *BISL – Een Introductie*. Van Haren Publishing, 2005.
- [Vog00] VOGT, WALTER (Herausgeber): *Nutzen ohne Frust: IT Services kundenorientiert planen und steuern*. Perseo Consult AG, 2000.
- [Wal98] WALTON, MARY: *The Deming Management Method*. Berkley Publishing Group, 1998.
- [WFM05] WFMC (WORKFLOW MANAGEMENT COALITION): *Process Definition Interface – XML Process Definition Language*, Oktober 2005. Version 2.0, WFMC-TC-1025, <http://tinyurl.com/2juuj8>.
- [WK06] WALTER, SVEN MARKUS und HELMUT KRCDMAR: *Reorganisation der IT-Prozesse auf Basis von Referenzmodellen – eine kritische Analyse*. itService Management, 2:3–10, November 2006.
- [Zac87] ZACHMAN, JOHN A.: *A Framework for Information Systems Architecture*. IBM Systems Journal, 26(3), 1987. <http://tinyurl.com/kj2xz>.
- [Zac03] ZACHMAN INSTITUTE FOR FRAMEWORK ADVANCEMENT: *The Framework for Enterprise Architecture – Cell Definitions*, 2003. <http://www.zifa.com/>.
- [zM03] MÜHLEN, MICHAEL ZUR: *Workflow- und Prozessmodellierung bei einem Energieversorgungsunternehmen*. In: BECKER, JÖRG et al. [BKR03], Seiten 511–532.

- [ZS92] ZACHMAN, JOHN A. und JOHN F. SOWA: *Extending and Formalizing the Framework for Information Systems Architecture*. IBM Systems Journal, 31(3), 1992. <http://tinyurl.com/2jttvk>.

# Abkürzungsverzeichnis

- ARIS** Architektur integrierter Informationssysteme
- AS** Australian Standard (*wie in „AS 8018“*)
- ASL** Application Services Library
- ASP** Application Service Provider
- ASZ** Anwender Service Zentrum (BMW)
- BCS** British Computer Society
- BiSL** Business Information Services Library
- BPDM** Business Process Definition Metamodel
- BPEL** Business Process Execution Language
- BPMI** Business Process Management Initiative
- BPML** Business Process Modelling Language
- BPMN** Business Process Modelling Notation
- BPR** Business Process Reengineering
- BS** British Standard (*wie in „BS 15000“*)
- BSI** British Standards Institution
- CAB** Change Advisory Board
- CCTA** Central Computer and Telecommunications Agency
- CFIA** Component Failure Impact Analysis
- CI** Configuration Item
- CMDB** Configuration Management Database
- CMM** Capability Maturity Model
- CMMI** CMM Integration
- CMU** Carnegie Mellon University
- CobiT** Control objectives for information and related Technology

<b>CORBA</b>	Common Object Request Broker Architecture
<b>CRM</b>	Customer Relationship Management
<b>CSCW</b>	Computer Supported Cooperative Work
<b>CSF</b>	Critical Success Factor
<b>DB</b>	Database
<b>DMAIC</b>	Define, Measure, Analyze, Improve, Control
<b>EAI</b>	Enterprise Application Integration
<b>ERP</b>	Enterprise Resource Planning
<b>EPK</b>	Ereignisgesteuerte Prozesskette
<b>EXIN</b>	Examination Institutein
<b>eTOM</b>	enhanced Telecom Operations Map
<b>FAB</b>	Fulfillment, Assurance and Billing ( <i>eTOM</i> )
<b>FTA</b>	Fault Tree Analysis
<b>GoM</b>	Grundsätze ordnungsgemäßer Modellierung
<b>IEC</b>	International Electrotechnical Commission
<b>ICT</b>	Information Communication Technology
<b>ICSP</b>	Internet and Communications Service Provider
<b>IPD-CMM</b>	Integrated Product Development CMM
<b>ISACA</b>	Information Systems Audit and Control Association
<b>ISEB</b>	Information Systems Examination Board
<b>ISP</b>	Internet Service Provider
<b>ISO</b>	International Organization for Standardization
<b>ISRI</b>	Institute for Software Research International
<b>IT</b>	Information Technology (deutsch: Informationstechnologie)
<b>ITGI</b>	IT Governance Institute
<b>ITIL</b>	IT Infrastructure Library
<b>ITSM</b>	IT Service Management
<b>itSMF</b>	IT Service Management Forum
<b>ITSP</b>	IT Service Provider



**ITsqc** IT Services Qualification Center

**ITU-T** International Telecommunication Union – Telecommunication Standardization Sector

**ITUP** IBM Tivoli Unified Process

**JTC** Joint Technical Committee

**KPI** Key Performance Indicators

**MDA** Model Driven Architecture

**MO** Managed Object

**MOF** Microsoft Operations Framework

**MSF** Microsoft Solutions Framework

**MTBF** Mean Time Between Failures

**MTTR** Mean Time To Repair

**NGOSS** New Generation Operation Support Systems

**NMF** Network Management Forum

**OASIS** Organization for the Advancement of Structured Information Standards

**OGC** Office of Government Commerce

**OLA** Operating Level Agreement

**MOF** Operations Management Review

**OMG** Object Management Group

**OSI** Open Systems Interconnection

**OSS** Operation Support Systems

**P-CMM** People CMM

**PDCA** Plan, Do, Check, Act (*Deming Cycle*)

**PDSA** Plan, Do, Study, Act (*Deming Cycle*)

**PRINCE** Projects In Controlled Environments

**QoS** Quality of Service

**RCB** Registered Certification Body

**RFC** Request for Change

**RM-ODP** Reference Model for Open Distributed Processing

**RM&O** Resource Management & Operations (*eTOM*)

<b>RUP</b>	Rational Unified Process
<b>SANS</b>	South African National Standard ( <i>wie in „SANS 15000“</i> )
<b>S/P</b>	Supplier/Partner ( <i>eTOM</i> )
<b>S/PRM</b>	Supplier/Partner Relationship Management ( <i>eTOM</i> )
<b>SA-CMM</b>	Software Acquisition CMM
<b>SAP</b>	Service Access Point
<b>SE-CMM</b>	Systems Engineering CMM
<b>SEI</b>	Software Engineering Institute
<b>SID</b>	Shared Information/Data (Model)
<b>SIP</b>	Strategy, Infrastructure and Product ( <i>eTOM</i> )
<b>SLA</b>	Service Level Agreement
<b>SLM</b>	Service Level Management
<b>SMF</b>	Service Management Function
<b>SM&amp;O</b>	Service Management & Operations ( <i>eTOM</i> )
<b>SNMP</b>	Simple Network Management Protocol
<b>SOA</b>	Service Oriented Architecture
<b>SOAP</b>	Simple Object Access Protocol
<b>SPOC</b>	Single Point Of Contact
<b>SPICE</b>	Software Process Improvement and Capability dEtermination
<b>SSE-CMM</b>	Systems Security Engineering Capability Maturity Model
<b>SW-CMM</b>	Software CMM
<b>TMF</b>	Telemanagement Forum
<b>TMN</b>	Telecommunications Management Network
<b>TOM</b>	Telecom Operations Map
<b>TQM</b>	Total Quality Management
<b>TTS</b>	Trouble Ticket System
<b>TÜV</b>	Technischer Überwachungsverein
<b>UC</b>	Underpinning Contract
<b>UML</b>	Unified Modelling Language

**VKD** Vorgangskettendiagramm  
**UDDI** Universal Description, Discovery and Integration  
**WFM** Workflow Management  
**WFMS** Workflow Management System  
**WfMC** Workflow Management Coalition  
**WSDL** Web Services Description Language  
**W3C** World Wide Web Consortium  
**XML** Extensible Markup Language  
**XPDL** XML Process Definition Language



# Abbildungsverzeichnis

1.1	Problematik des komponentenorientierten IT-Managements . . .	2
1.2	Service Level Management im komponentenorientierten Netz- und Systemmanagement . . . . .	4
1.3	Service Management Plattform [DR02] . . . . .	5
1.4	Availability (in Anlehnung an [vB05])) . . . . .	7
1.5	Vervollständigte Sicht auf das Service Management . . . . .	9
1.6	Einordnung der zentralen Fragestellung . . . . .	11
1.7	Aufbau der Arbeit . . . . .	13
2.1	Klassische organisatorische Strukturierung unternehmensinterner IT . . . . .	18
2.2	Qualitätskreis von Deming (Darstellung nach [OGC00]) . . . . .	20
2.3	Phasen des kontinuierlichen Prozessmanagements nach [NPW03] . . . . .	21
2.4	Vereinfachtes Lebenszyklusmodell des prozessorientierten ITSM . . . . .	22
2.5	Allgemeines Modell des Prozessmanagements (nach [OGC00]) . . . . .	27
2.6	MNM-Dienstmodell . . . . .	29
2.7	Rollen und Parteien . . . . .	29
3.1	ITIL Kernprozesse . . . . .	35
3.2	Zentrale Begriffe im ITIL Service Level Management . . . . .	45
3.3	TMN-Pyramide [IT96] . . . . .	51
3.4	Telecom Operations Map [TMF00] . . . . .	53
3.5	Strukturierung der eTOM-Prozesse [TMF04a] . . . . .	55
3.6	eTOM Level 0 Sicht [TMF04a] . . . . .	56
3.7	eTOM Level 1 in der Übersicht [TMF04a] . . . . .	58
3.8	eTOM – Customer Relationship Management – Zerlegung in Level 2 Prozesselemente [TMF04f] . . . . .	59
3.9	eTOM – Problem Handling – Zerlegung in Level 3 Prozesselemente . . . . .	59
3.10	Service Management & Operations – Zerlegung in Level 2 Prozesselemente . . . . .	60
3.11	Resource Management & Operations – Zerlegung in Level 2 Prozesselemente . . . . .	60
3.12	Supplier/Partner Relationship Management: Zerlegung in Level 2 Prozesselemente . . . . .	61
3.13	Zuordnung Operations Level 2 Prozesse zu den vertikalen end-to-end Prozessgruppierungen . . . . .	63

3.14	Beispiel für einen Prozessablauf in der Assurance-Prozessgruppe	65
3.15	Reifegrade und Fähigkeitsgrade im CMMI	71
3.16	CobiT-Rahmenstruktur	74
3.17	Fähigkeitsgrade im eSCM	78
3.18	MOF Prozessmodell	80
3.19	ISO/IEC 20000 – Service Management Prozesse [ISO05b]	86
3.20	Das „ITSM Quagmire“	93
3.21	Grundlegende Ausrichtung gängiger ITSM-Rahmenwerke	96
3.22	Einordnung der Schwerpunkte ITSM-relevanter Rahmenwerke	97
4.1	Klassifikation von Management-Werkzeugen nach [HAN99]	103
4.2	Einordnung der Schwerpunkte gängiger Management-Werkzeuge	105
4.3	Grundlegende Arten von prozessorientierten ITSM-Werkzeugen	108
4.4	Einfluss von Kriterien zur Prozessklassifikation auf Aspekte der Werkzeugunterstützung	113
4.5	Relative Einordnung der Strukturiertheit der ITIL-Prozesse	118
4.6	Charakteristiken der ITIL-Prozesse	119
4.7	Einordnung von ITIL-Prozessen in vier Quadranten	122
5.1	Anwendungsdomäne des Ansatzes	130
5.2	Richtlinien für ITIL-Referenzmodelle und ihr Einfluss auf die Grundsätze ordnungsgemäßer Modellierung	131
5.3	ARIS-Konzept: Das „ARIS-Haus“ [Sch97]	135
5.4	Zachman Framework nach [ZS92]	136
5.5	Ordnungsrahmen für modellbasierten Ansatz (beispielhafte Modellinhalte)	138
5.6	Vorgehen bei der Modellerstellung	139
6.1	Farbliche Kennzeichnung direkt übernommener ITIL-Konzepte (vgl. Abbildung A.3)	146
6.2	Ausschnitt aus dem Datenmodell für Incident Management (vgl. Abbildung A.4)	148
6.3	Haupteinflussgrößen für Einteilung und Zuweisung von Kategorien (vgl. Abbildungen A.4, A.5 und A.6)	149
6.4	Darstellung des Workflows von ITIL-Prozessen mittels Aktivitätsdiagrammen (vgl. Abbildungen A.7, A.14 und A.15)	151
6.5	Integration zwischen Incident Management und Change Management (vgl. Abbildungen A.11 und A.15)	153
6.6	Architekturelemente im Workflow Reference Model [Hol95]	154
A.1	Einordnung der Modelle	163
A.2	Für die anderen Prozesse relevante zentrale Entitäten und Dokumente im <i>Configuration Management</i>	164

A.3	Für die anderen Prozesse relevante zentrale Entitäten und Dokumente im <i>Service Level Management</i> . . . . .	165
A.4	Incident Management – zentrale Entitäten und Dokumente . . . . .	166
A.5	Problem Management – zentrale Entitäten und Dokumente . . . . .	167
A.6	Change Management – zentrale Entitäten und Dokumente . . . . .	168
A.7	Incident Management – Gesamtprozess . . . . .	169
A.8	Incident Management – Incident Detection and Recording . . . . .	170
A.9	Incident Management – Classification and Initial Support . . . . .	171
A.10	Incident Management – Investigation and Diagnosis . . . . .	172
A.11	Incident Management – Resolution and Recovery . . . . .	173
A.12	Incident Management – Incident Closure . . . . .	174
A.13	Incident Management – Major Incident Handling . . . . .	175
A.14	Problem Management – Gesamtprozess (ohne „Proactive Problem Management“) . . . . .	176
A.15	Change Management – Gesamtprozess . . . . .	177
A.16	Beispielhafte Einordnung der in ITIL definierten „Functions“ . . . . .	178
A.17	Incident Management – zentrale Rollen und Funktionsbereiche . . . . .	179
A.18	Problem Management – zentrale Rollen . . . . .	180
A.19	Change Management – zentrale Rollen . . . . .	181
A.20	Incident Management – grundlegendes Datenmodell für eine Realisierung eines <i>Incident Record</i> . . . . .	182
A.21	Incident Management – Status eines <i>Incident Record</i> (mit erlaubten Übergängen) . . . . .	183
A.22	Incident Management Nutzfalldiagramm für einen Client für User (Service User Client) . . . . .	184
A.23	Incident Management – Nutzfallbeschreibung für <i>Submit Incident (Failure) Report</i> (vgl. Abbildung A.22) . . . . .	185
A.24	Incident Management – Interfaces des Incident-Management-Systems im Sinne des Workflow Reference Model . . . . .	186
A.25	Incident Management – Zugriff auf Workflow Clients . . . . .	187
A.26	Incident Management – Definition der Datentypen in einem XML Schema . . . . .	188
A.27	Incident Management – Beispiel für <i>Incident Record</i> , vgl. Abbildung A.26 . . . . .	189
A.28	Incident Management – BPEL-Workflow als Aktivitätsdiagramm . . . . .	190
A.29	Incident Management – WSDL-Elemente für die Schnittstelle zum <i>Service User Client</i> . . . . .	191
A.30	Incident Management – grundlegende Benutzeroberflächen-Navigation aus Anwendersicht . . . . .	192





# Index

- Prozessmanagement, 21, 62
- 3K-Modell, 106
  
- Application Integration, 107
- Application Management, 87
- ARIS, 23, 88
- AS 8018, 84
- ASL, 87
- ASL Foundation, 87
- ASP, 17
- ASZ, 116
- Availability, 6
  - Detection Time, 8
  - Downtime, 6
  - Fix Time, 8
  - MTBF, 6, 7, 117
  - MTTR, 6, 7, 117
  - Recovery Time, 8
  - Response Time, 8
  - Uptime, 6
- Availability Management, 47, 85
  - Availability Manager, 47
  - CFIA, 47
  - FTA, 47
  
- BCS, 33
- Benchmarking, 24
- Best Practice, 25, 26, 33, 34, 68, 70, 77, 92
  - Best Practices Database, 25
  - Common Practice, 25, 34
  - Global Best Practices Database, 25
  - Good Practice, 26
  - Industry Best Practices, 25, 26
- BiSL, 87
  
- BMW Group, 116
- Bootstrap, 70
- BPDM, 88
- BPEL, 88, 157
- BPMI, 88
- BPML, 88
- BPMN, 88, 157
- BS 15000, 78, 83, 84, 90, 91
  - BIP 0005, 83
  - BS15000-1:2002, 84
  - BS15000-2:2003, 84
  - BS15000:2000, 84
  - Code of Practice, 83, 84
  - IT Service Management Self-assessment Workbook, 84
  - Manager's Guide to Service Management, 83
  - PD 0015, 84
  - PD0005, 83, 84
  - PD0005:1998, 83
  - PD0015:2002, 84
  - Self-assessment Workbook, 84
  - Self-assessment-workbook, 84
  - Supplier Processes, 85
- BSI, 83, 85
  
- Capability Level, 75
- Capacity Management, 35, 46, 47, 111, 123
  - Business Capacity Management, 46
  - Capacity Manager, 46
  - Capacity Plan, 46
  - CDB, 46
  - Resource Capacity Management, 46

- Service Capacity Management, 46
- Carnegie Mellon University, 70, 76
  - ISRI, 77
  - ITsqc, 77
  - School of Computer Science, 76
  - SEI, 70, 72, 76
- Case Data, 152
- CCTA, 32, 33, 87
- Change Management, 35, 41–44, 47, 69, 120, 121, 123, 124, 129, 141–143, 149–152, 158, 163
  - Allocation of Priorities, 42
  - Backout Plan, 42
  - Backouts, 43
  - CAB, 42
  - CAB Meetings, 42
  - CAB meetings, 150
  - Change, 41–43, 149, 151, 152
  - Change Approval, 42
  - Change Authority, 43
  - Change Builder, 43
  - Change Building, Testing and Implementation, 43
  - Change Categorisation, 42
  - Change Log, 42
  - Change Logging and Filtering, 42
  - Change Manager, 42, 43
  - Change Model, 42
  - Change Record, 42
  - Change Review, 43, 152
  - Change Scheduling, 43
  - Change Tester, 43
  - Establishing a Change Advisory Board, 43, 150
  - FSC, 42
  - Impact and Resource Assessment, 42
  - Implementation Plan, 42
  - PIR, 42
  - Planning the Implementation of Operational Processes, 42, 142, 150
  - PSA, 42
  - Reviewing the Change Management Process for Efficiency and Effectiveness, 43, 150
  - RFC, 37, 41–43, 147
  - Roles and Responsibilities, 43, 150
  - Staff, 43
  - Standard Changes, 42
  - Urgent Change, 150, 151
  - Urgent Change Building, Testing and Implementation, 43
    - Urgent Changes, 43
- CMM, 69, 70, 76, 112
  - IPD-CMM, 70
  - P-CMM, 70
  - SA-CMM, 70
  - SE-CMM, 70
  - SSE-CMM, 70
  - SW-CMM, 70
- CMMI, 70, 71, 91
  - Capability Level, 71, 72
  - Continuous Representation, 71
  - Generic Goals and Practices, 71, 72
  - Goals, 71
  - Integrated Product and Process Development (EPD), 70
  - Maturity Level, 71
  - Maturity-Level, 72
  - Practices, 71
  - Process Area, 71
  - Software Engineering (SE), 70
  - Staged Representation, 71
  - Subpractices, 71, 72
  - Supplier Sourcing, 70
  - Systems Engineering, 70
- CobiT, 73, 90, 91, 112
  - Acquire and Implement, 74
  - Deliver and Support, 74
  - Monitor and Evaluate, 74
  - Plan and Organize, 74

- Practices, 75
- Version 3, 75
- Configuration Management, 10, 35, 39–42, 44, 47, 118–120, 122, 124, 143, 147, 163, 164
  - Change Record, 42
  - CI Record, 144
  - CMDB, 40, 124, 147
  - CMDB Back-ups, Archives and Housekeeping, 40
  - Configuration Audits, 40
  - Configuration Identification, 40, 120
  - Configuration Item, 40–42, 144, 149
  - Configuration Librarian, 41
  - Configuration Management Planning, 40, 119
  - Configuration Management Team, 41
  - Configuration Manager, 41
  - Configuration Record, 147
  - Configuration Status Accounting, 40
  - Configuration Verification and Audit, 40, 119
  - Control of CIs, 40, 120
  - Key Performance Indicators, 41
  - Providing a Configuration Management Service, 40
  - Setting up a Change, Configuration and Release Management Function, 44
  - Status Accounting, 120
- Continuous Improvement, 20
- Cost Center, 19
- CSCW, 11, 106
- Data Model, 145
- Deming Cycle, 20–22, 80
- Descriptive Guidance, 48
- Design Patterns, 92
- DMAIC, 21
- Domain Model, 134
- DQS GmbH, 84
- Durchlaufzeit, 110
- Enterprise Service, 3, 4, 6, 16, 17, 28
- eSCM, 76–78, 90, 91, 95, 112
  - Capability Area, 78
  - Capability Areas, 77
  - Capability Level, 77
  - eSCM-CL, 77
  - eSCM-SP, 77
  - Practices, 77, 78
  - Sourcing Life-Cycle, 77
- eTOM, 28, 48, 50
  - Application Notes, 53, 54
  - Assurance, 62, 64, 66
  - Billing, 62
  - Close Problem, 59
  - Customer, 55, 56
  - Customer Interface Management, 59, 64
  - Customer QoS/SLA Management, 64
  - Customer Relationship Management, 58, 66
  - Employees, 55
  - End User, 56
  - end-to-end Process Grouping, 57
  - end-to-end Process Groupings, 66
  - Enterprise Management, 57, 66, 67
  - Enterprise Risk Management, 67
  - eTOM-ITIL Application Note, 54, 67
  - FAB Processes, 62
  - Fulfillment, 62, 69
  - Functional Areas, 57
  - ICSP, 50, 67
  - Infrastructure Life Cycle Management, 66
  - Isolate Problem & Initiate Resolution, 59

- Level 0, 54, 56, 66
- Level 1, 54, 61, 62, 66
- Level 2, 57, 58, 60, 66
- Level 3, 54, 57, 58, 66, 67
- Level 4, 54
- Levels, 54
- Market, Product & Customer, 57
- Marketing & Offer Management, 66
- Operations, 54, 56, 57, 61, 62, 66
- Operations Support & Readiness, 62
- Other Stakeholders, 55
- Problem Handling, 58, 64
- Process Decompositions and Descriptions, 54
- Process Flow, 57, 66, 68
- Process Flow Examples, 54
- Process Flows, 64
- Product Life Cycle Management, 66
- Public B2B Business Operations Map, 54, 61
- Report Problem, 59
- Resource Development & Management, 66
- Resource Management & Operations, 66
- Resource Trouble Management, 64
- Retention & Loyalty, 64
- Service Development & Management, 66
- Service Management & Operations, 59, 60, 66
- Service Problem Management, 64
- Service Quality Management, 64
- Shareholders, 55
- SLA, 58, 62, 64
- Solution Suites, 53
- Strategy & Commit, 66
- Strategy Infrastructure & Product, 66
- Strategy, Infrastructure & Product, 56, 57, 60, 62
- Strategy, Infrastructure & Support, 57
- Subscriber, 56
- Supplier Chain Development & Management, 66
- Supplier/Partner Relationship Management, 60
- Suppliers/Partners, 55, 57
- Track & Manage Problem, 59
- EXIN, 33, 87
- Fachbegriffsmodell, 146
- Fallfrequenz, 110
- Financial Management for IT Services, 45, 46, 85, 101
  - Auditing, 46
  - Budgeting, 46
  - Charging, 46, 85
  - Financial Manager, 46
  - IT Accounting, 46
- Framework Quagmire, 92
- Geschäftsprozessmanagement, *siehe* Prozessmanagement1, 18
- Grundsätze ordnungsgemäßer Modellierung, 48, 130
- IEC, 84
- IITUP, 97
- Incident Management, 9, 10, 26, 27, 34–39, 41, 42, 47, 107, 110, 115–117, 121, 123, 124, 129, 141, 142, 148–158, 160, 162, 163
  - Classification and Initial Support, 37, 49, 150
  - Functional Escalation, 37
  - Handling of Major Incidents, 37, 150
  - Hierarchical Escalation, 37

- Impact, 37, 148
- Incident, 26, 27, 36–39, 42, 107, 110, 111, 115–117, 143, 144, 147–150, 152
- Incident Closure, 37, 149, 150, 152
- Incident Detection and Recording, 37, 49, 150
- Incident Manager, 37, 110, 151, 153
- Incident Matching, 37
- Incident Record, 37, 41, 110, 144, 147, 148, 155, 182, 183, 189
- Incident-Handling Support Staff, 153
- Incident-handling support staff, 38, 39
- Investigation and Diagnosis, 37, 150
- Major Incident, 151
- Major Incidents, 150
- Ownership, Monitoring, Tracking and Communication, 37, 150
- Priority, 37
- Resolution and Recovery, 37, 150
- Service Request, 36
- Service Request Handling, 150
- Service Request Procedures, 49
- Support Group, 37
- Urgency, 37, 148
- Incident Management System, 153
- Incident Management Workflow Service, 155
- Incident-Management-System, 154–156, 161
- Internet-Management, 89, 94
- ISACA, 73
- ISEB, 33
- ISO, 83
  - JTC 1, 84
  - JTC 1 / SC 7, 84
- ISO 15504, 70, 79
- ISO/IEC 20000, 9, 31, 42, 78, 83–87, 90, 91, 95–97, 147
  - Budgeting and Accounting for IT Services, 85
  - Business Relationship Management, 85
  - Change Record, 42
  - Code of Practice, 84, 86
  - minimum set of requirements, 86
  - Relationship Processes, 85
  - Service Level Reports, 85
  - Service Reporting, 85
  - Specification, 84, 86
  - Supplier Management, 85
- ISRI, *siehe* Carnegie Mellon University
- IT Auditing, 74, 159
- IT Governance Institute, 90
- IT Service Continuity Management, 46, 85, 101
  - Business Continuity Managements, 47
  - ITSC Manager, 47
  - ITSC Team Leader, 47
  - ITSC Team Member, 47
- ITGI, 74
- ITIL, 8, 32, 91
  - Activity, 49
  - Application Management, 32, 79, 87
  - Code of Practices, 32
  - Complementary Guidance, 33
  - Computer Operations Set, 32
  - Core Guidance, 33
  - Core Titles, 32
  - FAQ, 100
  - Function, 34
  - ICT Infrastructure Management, 32, 81
  - Incident, 49
  - Introduction to ITIL, 33
  - ITIL Back Catalogue, 33
  - ITIL Core Guidance, 132, 142

- ITIL Foundation, 33
- ITIL Practitioner, 33
- ITIL Small-scale Implementation, 33
- ITIL Version 1, 33
- Management Reporting, 49
- Manager's Certificate in IT Service Management, 33
- Manager's Set, 32
- Metrics, 49
- Networks Set, 32
- Planning to Implement Service Management, 32, 79
- Public Domain Framework, 34
- Security Management, 32
- Service Delivery, 32–36, 39, 41, 44, 47, 48, 75, 80, 83, 85, 120
- Service Management Software Tools, 47
- Service Request, 36
- Service Support, 32–39, 46–49, 75, 78, 81, 83, 85, 141, 143, 147, 149, 150, 152, 153, 158
- Service-Request, 36
- Software Asset Management, 33
- Software Support Set, 32
- Support Group, 37
- The Business Perspective, 32
- Version 2.0, 33, 83
- Workaround, 36
- ITS-CMM, 72, 91
- ITSM, 1
- ITSMF, 33
- ITSP, 17
- ITsqc, *siehe* Carnegie Mellon University
- ITU-T, 51
- ITUP, 82, 90
  - Availability Management, 83
  - Description, 82, 83
  - IBM Tivoli Process Manager, 83
  - IBM Tivoli Unified Process Tool, 82
  - Incident Management, 83
  - ITAPM, 83
  - Process Mappings, 90
  - Team Allocation, 82
  - Work Breakdown Structure, 82
  - Work Product Usage, 82
- Kernprozess, 109
- KPI, 21, 26–28, 38, 39, 44, 47, 49, 95, 104, 142
- long running processes, 110
- Managed Object, 93
- Managed Objects, 94
- Maturity Level, 75
- MDA, 134
  - CIM, 134
  - Computation Independent Viewpoint, 134, 136
  - PIM, 134
  - Platform Independent Viewpoint, 134, 136
  - Platform Specific Viewpoint, 134, 136
  - PSM, 134
  - viewpoints, 134
- MNM-Dienstmodell, 28
- Modellierungssprache, 88, 91, 132, 137, 140
- MOF, 9, 41, 49, 79, 81, 91, 96, 97
  - Change Management, 41
  - Changing Quadrant, 80, 81
  - Infrastructure Engineering, 81
  - IT Life Cycle, 80
  - MOF Pocket Guide, 79
  - Operating Quadrant, 80, 81
  - Operations Management Reviews, 80
  - Operations Review, 80
  - Optimizing Quadrant, 80
  - Prescriptive Guidance, 81

- Process Model, 79
- Release Approved Review, 80
- Release Readiness Review, 80
- Risk Model, 79
- SLA Review , 80
- SMF, 79, 81
- Supporting Quadrant, 80, 81
- Team Model, 79, 81
- White Papers, 81
- Workforce Management, 81
- MSF, 79, 80
- Multi-Tier Architecture, 15
  
- NGOSS, 52, 67, 69
- Nichtverfügbarkeit, 7
- NMF, 51
  
- OGC, 19, 33, 48, 50, 88, 100
- OMG, 88, 134
- Organisatorische Komplexität, 111
- OSI Management, 51, 89, 94, 95
  - OSI Management Framework, 94, 134
  - Systems Management Functional Areas, 94
- OSS, 51
  
- PDCA, *siehe* Deming Cycle
- People Model, 146
- Pink Elephant, 101
- PinkRocade, 87
- PinkVerify, 101
- Prince2, 87
- PRM-IT, 90, 91
- Problem Management, 35, 38, 39, 41, 115–117, 121, 123, 124, 129, 141–143, 147, 149, 152, 158, 162, 163
  - Error Closure, 152
  - Error Control, 38, 39, 143, 144
  - Known Error, 37–39, 143, 144
  - Known Error Record, 144
  - Post Implementation Review, 39
  - Proactive Problem Management, 38, 39, 115
  - Problem, 37–39, 115, 143, 144, 146, 147, 149, 152
  - Problem / Error Database, 38
  - Problem Category, 147
  - Problem Control, 38, 39, 143, 144
  - Problem Manager, 39
  - Problem Record, 39, 41, 144, 146, 147
  - Problem Support, 39
  - Root Cause, 38, 39, 143, 144, 149, 152
  - Root Cause Analysis, 38
  - Work-around, 38, 152
- Profit Center, 19
- Prozessmanagement, 10, 18, 20, 22, 29, 88, 93, 94, 100, 110
  - BPR, 18
  - Fall, 26, 110, 112, 113
  - Geschäftsprozess, 18
  - Istmodellierung, 22
  - IT-Prozess, 19
  - Kernprozess, 19
  - kontinuierliches Prozessmanagement, 21
  - Process Executive, 29
  - Process Manager, 23, 29, 30, 37, 39, 41
  - Process Owner, 26, 29
  - Process Responsible, 29
  - Sollmodellierung, 22
  - Supportprozess, 19
  - TQM, 18
- QoS, 3, 89
- QoS-Parameter, 3, 6, 62, 64
  
- RCB, 84
- Reifegrad, 91
- Release Management, 35, 43, 44, 47, 75, 115, 117, 120, 121, 123, 140, 152, 161

- Communication, Preparation and Training, 44
- Designing, Building and Configuring a Release, 44
- DHS, 44
- Distribution and Installation, 44
- DSL, 44
- Release, 43, 44
- Release Acceptance, 44
- Release Manager, 44
- Release Planning, 44
- Release Unit, 44
- Rollout Planning, 44
- RM-ODP, 134
  - Viewpoints, 134
- RUP, 161
  
- SANS 15000, 84
- SAP, 15
- Security Management, 85
- SEI, *siehe* Carnegie Mellon University
- Service Desk, 34, 62, 81, 116, 117, 153
- Service Level, 3, 6, 9, 44, 46, 49, 107, 111, 113, 117, 121, 124, 145, 148
- Service Level Management, 3, 6, 35, 44, 45, 59, 118, 120, 121, 123, 140, 147, 148, 161, 165
  - Implementing the process, 45
  - IT Service Catalogue, 45
  - OLA, 44
  - On-going Process, 45, 120
  - Planning the process, 45
  - Service Level Manager, 45
  - Service Review Meeting, 120
  - UC, 44
- Service-Parameter, *siehe* QoS-Parameter
- Shewhart Cycle, *siehe* Deming Cycle
- SID, 28, 52, 69
- Six Sigma, 21
  
- SLA, 3, 8, 9, 17, 27, 28, 34, 44, 46, 59, 116, 120, 124, 145, 147
- SNMP, 89
- SNMP-Management, *siehe* Internet-Management
- SOA, 15, 88, 155
- Software Engineering, 31, 43, 69, 70, 72, 88, 89, 91, 92, 159, 161
- Sourcing Management, 77
- SPICE, 70, 79
- SPOC, 34
- Strategic Alignment Model, 87
  
- TÜV Süd, 84
- Thing-Information Pattern, 146, 147, 160
- TM-Forum, *siehe* TMF
- TMF, 48, 50, 51, 61
- TMN, 51, 57, 61
  - TMN Enterprise Management, 57
- TNA, 52
- Toolselector, 101
- Trouble-Ticket-System, 104
  
- UML, 23, 88
  - Activity Diagrams, 104
  - Deployment Diagrams, 104
- V-Modell XT, 161
- Van Haren, 87
  
- Web-Services, 88, 156, 157, 161
- Workflow, 23
- Workflow Management Coalition, 88, 152, 154
- Workflow Patterns, 156
- Workflow Reference Model, 152, 154, 155, 160
  - Electronic Mail Model, 156
  - Interface 1, 154
  - Interface 6, 154
  - Workflow Application Data, 155



- Workflow Client Application, 155, 156
- Workflow Enactment Service, 154–156
- Workflow Engine, 154, 156
- Workflow Interoperability, 152
- Worklist, 156
- Workflow-Kontinuum, 109
- Workflow-Management-System, 107, 121, 154
  
- XPDL, 88
  
- Zachman Framework, 14, 89, 127, 135–139, 160
  - Builder View, 139, 141, 145, 155–157
  - Designer View, 139, 141, 144, 145, 154, 156, 157, 161
  - Owner View, 137–139, 141, 142, 145, 147, 148, 150, 157, 158, 161

