

# UNIVERSITÄT LEIPZIG

## **Methoden überbetrieblicher Service- und Prozessmodellierung am Beispiel von RosettaNet**

Jan Oberländer

**Heft 5**



**FORSCHUNGSBERICHTE DES INSTITUTS FÜR WIRTSCHAFTSINFORMATIK**



# UNIVERSITÄT LEIPZIG





**Forschungsberichte des Instituts für Wirtschaftsinformatik  
der Universität Leipzig**  
Heft 5

Methoden überbetrieblicher Service- und Prozessmodellierung  
am Beispiel von RosettaNet

Jan Oberländer

Herausgeber	Prof. Dr. Rainer Alt, Prof. Dr. Ulrich Eisenecker, Prof. Dr. Bogdan Franczyk
ISSN	1865-3189
Redaktion	Institut für Wirtschaftsinformatik der Universität Leipzig
Telefon	(0341) 97 33 720 (0341) 97 33 600
E-Mail	<a href="mailto:iwi@wifa.uni-leipzig.de">iwi@wifa.uni-leipzig.de</a>
Internet	<a href="http://www.iwi.uni-leipzig.de/">http://www.iwi.uni-leipzig.de/</a>
Redaktionsschluss	15.02.2010

## **Vorwort**

"RosettaNet" bezieht sich auf den sogenannten "Rosetta Stone", einen in der ägyptischen Stadt Rosette um 1799 gefundenen Stein. Dieser enthielt Hieroglyphen, welche in Ägyptisch und Griechisch wiedergegeben waren, sodass ihm eine wichtige Übersetzungsfunktion zukam. Angewandt auf die Geschäftsabwicklung eines Unternehmens mit Kunden und Lieferanten findet sich häufig eine ähnliche Heterogenität bezüglich der Produktbezeichnungen, Prozessabläufe und der ausgetauschten Nachrichtentypen wieder. Eine einheitliche Sprache zwischen den in einer Wertschöpfungskette, einer Branche, einer Volkswirtschaft oder gar international agierenden Unternehmen würde auf erhebliche Weise die Transaktionskosten zwischen Unternehmen reduzieren.

Hier setzt RosettaNet an. Der vorliegende Forschungsbericht beruht auf der Diplomarbeit von Jan Oberländer und beschreibt neben Struktur und Zielen des RosettaNet-Konsortiums die von RosettaNet vorgesehenen Methoden zur überbetrieblichen Vernetzung, insbesondere jenes der privaten und öffentlichen Prozesse. Wie kaum ein zweiter gegenwärtig existierender Standardisierungsansatz reichen die Konventionen von RosettaNet von einzelnen, eher technisch ausgeprägten Nachrichtentypen, hin zur Spezifikation betriebswirtschaftlicher Prozesse. Diese reichen von der Entwicklung über die Auftragsabwicklung und das Lagermanagement hin zu Vertrieb und Service. Die Bedeutung für die überbetriebliche Vernetzung ist daher nicht hoch genug einzuschätzen.

Die Arbeit von Jan Oberländer vermittelt einen sehr guten Einblick in RosettaNet und präsentiert eine Einordnung in die überbetriebliche Vernetzung sowie die Anwendung mit einem konkreten Beispiel. Anhand eines Vertriebsprozesses aus dem Finanzbereich zeigt sie, dass sich RosettaNet nicht nur für Abläufe in der Elektronikbranche eignet, sondern eine Übertragbarkeit auf andere Branchen gegeben ist. Zusätzlich findet sich eine Umsetzung mit gängigen Entwicklungswerkzeugen. Diese deuten darauf hin, dass zwar weiterhin die Notwendigkeit zur Modellierung besteht und eine vollständige ad-hoc-Automatisierung zwischen Unternehmen aussteht, die heutigen Grenzen von RosettaNet aber auch zukünftige Entwicklungsmöglichkeiten darstellen.

Leipzig, im September 2010

Rainer Alt





# Inhaltsverzeichnis

<b>Abbildungsverzeichnis .....</b>	<b>VII</b>
<b>Tabellenverzeichnis .....</b>	<b>VIII</b>
<b>1 Einleitung .....</b>	<b>1</b>
<b>2 Grundlagen der Service- und Prozessmodellierung.....</b>	<b>4</b>
2.1 Grundlagen der Servicemodellierung.....	4
2.1.1 Begriffsverständnis Serviceorientierung .....	4
2.1.2 Aufbau und Bestandteile serviceorientierter Architekturen.....	6
2.1.3 Servicemodellierung im Rahmen der Serviceentwicklung .....	8
2.1.4 Unterscheidung von Servicearten .....	9
2.2 Grundlagen der Geschäftsprozessmodellierung .....	11
2.2.1 Einordnung in den Rahmen des Geschäftsprozessmanagements .....	11
2.2.2 Motivation und Ziel der Geschäftsprozessmodellierung.....	14
2.2.3 Begriffsbestimmungen im Rahmen der Prozessmodellierung.....	16
2.2.4 Modellierungsmöglichkeiten auf Basis vorhandener Methoden und Sprachen .....	18
2.3 Zusammenhänge zwischen Prozess- und Servicemodellierung .....	20
2.3.1 Services als Bausteine von Prozessketten.....	20
2.3.2 Vor- und Nachteile bei der Verwendung von Services in Prozessketten .....	22
2.3.3 Probleme auf dem Weg vom Geschäftsprozess- zum Servicemodell .....	22
2.4 Zusammenfassung .....	24
<b>3 Entwicklung organisationsübergreifender Prozesse und Services..</b>	<b>24</b>
3.1 Motivation und Zweck unternehmensübergreifender Prozesse .....	24
3.2 Grundlagen öffentlicher Prozesse .....	26
3.3 Grundlagen öffentlicher Services.....	29
3.4 Organisation und Architektur unternehmensübergreifender Prozesse und Services.....	31
3.4.1 Unternehmensexterne Lösungsansätze für die Prozessgestaltung.....	31
3.4.2 Organisationsübergreifendes Prozessmanagement.....	32
3.4.3 Modellierung organisationsübergreifender Prozesse.....	33
3.4.4 Implementierung organisationsübergreifender Prozesse .....	36
3.4.5 Ausführung organisationsübergreifender Prozesse.....	39
3.4.6 Sprachen und Ansätze zur Prozessmodellierung .....	42
3.5 Organisationsübergreifende Prozesse am Beispiel von RosettaNet.....	45
3.5.1 Entstehung und Ziele des RosettaNet Konsortiums .....	45
3.5.2 Standardisierungsbereiche.....	46
3.5.3 Sichten und Cluster des RosettaNet Frameworks .....	46
3.5.4 Partner-Interface-Process Spezifikationen .....	48

3.5.5	Das RosettaNet Implementation Framework.....	50
3.5.6	RosettaNet im Rahmen des Geschäftsprozessmanagements.....	51
3.5.7	RosettaNet-basierte WebServices mit WSDL und BPEL4WS.....	52
<b>4</b>	<b>Unterstützung des Vertriebs im Finanzdienstleistungsbereich durch RosettaNet .....</b>	<b>56</b>
4.1	Referenzprozess Vertrieb als Beispiel eines unternehmensübergreifenden Prozesses .....	56
4.2	Fallbeispiel Bestellung im Rahmen des Vertriebsprozesses.....	59
4.3	Implementierung des Fallbeispiels mit dem ORACLE WebLogic Server .....	62
4.4	Modellierung des Vertriebsprozesses mithilfe der Business Process Modeling Notation .....	67
<b>5</b>	<b>Schluss .....</b>	<b>69</b>
	<b>Literaturverzeichnis .....</b>	<b>72</b>
	<b>Anhänge.....</b>	<b>78</b>
	Anhang A: RosettaNet Message Transfer (Framework Overview) .....	78
	Anhang B: BPMN 1.1 Notationselemente und Beispieldiagramm Vertrieb .....	79
	Anhang C: RosettaNet Clusters and Segments .....	80
	Anhang D: Bestellprozess bei der Universal Business Language .....	81
	Anhang E: Beschreibung des Referenzprozess Vertrieb .....	82
	Anhang F: PIP-Spezifikationen für den Dienstleistungsbereich.....	85
	Anhang G: Kombination der PIP-Spezifikationen mit dem Vertriebsprozess.....	89
	Anhang I: Bildschirmfotografie der ORACLE WebLogic Integration Console .....	97
	Anhang J: Geschäftsprozessmodell zur Implementierung der PIP 3A4 .....	98
	Anhang K: Bildschirmfotografien der Ausführung der RosettaNet-PIP 3A4.....	100

## Abbildungsverzeichnis

Abbildung 1: Schematischer Aufbau und Gliederung der Arbeit.....	2
Abbildung 2: Reifegrade der SOA in Bezug auf unternehmensübergreifende Prozesse .....	8
Abbildung 3: Bestandteile des Modellierungsszenarios .....	12
Abbildung 4: Phasenkonzept der Prozessmodellierung.....	14
Abbildung 5: Terminologieübersicht Geschäftsprozess- und Servicemodellierung .....	18
Abbildung 6: Schemata organisationsübergreifender Prozesse.....	28
Abbildung 7: Sicherheitsarchitektur für öffentliche Services .....	31
Abbildung 8: Integrationsebenen organisationsübergreifender Prozesse .....	33
Abbildung 9: Transformationskonzept privater zu öffentlichen Prozessen .....	34
Abbildung 10: Organisationsübergreifende Prozessintegration .....	35
Abbildung 11: Implementierung überbetrieblicher Prozesse am Beispiel der Bestellung .....	38
Abbildung 12: ebXML Design-Time und Run-Time Referenzmodell.....	44
Abbildung 13: RosettaNet Clusters, Segments, PIPs, Activities und Actions.....	48
Abbildung 14: RosettaNet Application Protocol Stack View .....	50
Abbildung 15: RosettaNet B2C-Server-Architektur.....	51
Abbildung 16: Vergleich von RosettaNet, WebServices und EbXML.....	53
Abbildung 17: Referenzprozess Vertrieb .....	56
Abbildung 18: RosettaNet PIP 3A4: Request Purchase Order .....	59
Abbildung 19: Referenzprozess zum Wertpapiergeschäft .....	60
Abbildung 20: UML Sequenzdiagramm der RosettaNet-PIP 3A4.....	61
Abbildung 21: PIP 3A4 mit WebServices und BPEL4WS .....	61
Abbildung 22: Modellausschnitt des Vertriebsprozesses auf Basis der BPMN .....	68
Abbildung 23: Teilbereiche serviceorientierter Geschäftsprozessmodellierung .....	69
Abbildung 24: Abstraktions- und Integrationsebenen überbetrieblicher Prozessmodellierung.....	70
Abbildung 25: RosettaNet Message Transfer .....	78
Abbildung 26: Notationselemente der BPMN 1.1 im Überblick.....	79
Abbildung 27: BPMN-Beispiel für den Vertrieb.....	79
Abbildung 28: RosettaNet Clusters und Segmente (englische Version) .....	80
Abbildung 29: Ordering Process als Teil der Universal Business Language .....	81

## **Tabellenverzeichnis**

Tabelle 1: Morphologischer Kasten für Merkmalsausprägungen von Services .....	9
Tabelle 2: Unterscheidung von Servicearten.....	10
Tabelle 3: Ebenenkonzept der Prozessmodellierung .....	11
Tabelle 4: Methoden und Sprachen der Prozessmodellierung.....	19
Tabelle 5: Morphologischer Kasten zur Differenzierung von Services .....	21
Tabelle 6: Morphologischer Kasten zur Klassifizierung von Prozessarten .....	26
Tabelle 7: Ausführungsvarianten organisationsübergreifender Prozesse .....	41
Tabelle 8: RosettaNet Clusters und Segmente.....	48
Tabelle 9: Kombination des Referenzprozess Vertrieb mit den PIP-Spezifikationen von RosettaNet .....	58

# 1 Einleitung

Der zentrale Bestandteil unternehmerischer Tätigkeit besteht darin, mit den gegebenen Ressourcen an Menschen, Material, Produktionsmitteln und Kapital den größtmöglichen Ertrag bei minimalem Ressourceneinsatz zu erzielen. Im Unternehmen laufen zu diesem Zweck Transformationsprozesse ab, um Güter und Dienstleistungen den Kunden gegenüber in der gewünschten Qualität, Menge und dem richtigen Preis anbieten zu können. Das Management dieser Prozesse ist eine komplexe Aufgabenstellung und erfordert sowohl gute Kenntnisse bei den zuständigen Mitarbeitern, die an der Planung der Prozesse beteiligt sind, als auch den Einsatz der richtigen Modellierungstechniken, Sprachen-, Standards- und Werkzeugen zur Erstellung, Ausführung und Überwachung der Prozesse.

Im Hinblick auf die zunehmende Spezialisierung der Unternehmen aber gleichzeitigen Zunahme der arbeitsteiligen Leistungserbringung, sind dabei nicht mehr nur unternehmensinterne Prozesse Gegenstand der Gestaltung, sondern zunehmend unternehmensübergreifende Prozesse, z. B. beim Outsourcing, strategischen Partnerschaften, Kooperationen und Fusionen. Die Fähigkeit, auch die Ressourcen von Partnern, Zulieferern und eventuell auch Kunden in die eigene Prozessplanung einbeziehen zu können, entscheidet zunehmend über den Markterfolg eines Unternehmens. Um den Marktbedingungen der heutigen Zeit gerecht zu werden, sind effektive und effiziente Geschäftsprozesse erforderlich, deren Unterstützung seitens der IT durch den Einsatz von Services geschehen kann, die innerhalb des eigenen Unternehmens, oder auch öffentlich zur Verfügung stehen können, um Partnern, Kunden und Lieferanten den Zugriff auf unternehmensinterne Informationen zu ermöglichen, bzw. diese von ihnen zu beziehen.

Den Kern dieser Arbeit bildet die Untersuchung des Spannungsfeldes zwischen fachlicher Geschäftsprozess- und Servicemodellierung, sowie der Ausführung von Geschäftsprozessen auf technischer Ebene, die durch Services unterstützt werden. Der Schwerpunkt liegt dabei auf dem unternehmens- bzw. organisationsübergreifenden Bereich, insbesondere dem Finanzdienstleistungssektor. Im ersten Teil der Arbeit wird zu den Grundlagen der Service- und Prozessmodellierung ein allgemeines Verständnis geschaffen. Zunächst wird der Frage nachgegangen, wie Services definiert und innerhalb von serviceorientierten Architekturen verwendet werden können. Dabei wird auf die verschiedenen Servicearten- bzw. Typen eingegangen und diese genauer charakterisiert.

Anschließend wird der Bereich der Geschäftsprozessmodellierung ausführlich betrachtet und auf seine Stellung im Rahmen des Geschäftsprozessmanagements eingegangen. Aufgrund der Vielzahl unterschiedlicher Definitionen erfolgt eine Begriffsbestimmung der wichtigsten Bestandteile von Geschäftsprozessmodellen und deren Charakterisierung. Als letztes werden verschiedene Modellierungsmöglichkeiten und die zugehörigen Sprachen und Standards dargestellt. Das Ende des ersten Teils bildet eine Untersuchung der Zusammenhänge zwischen der Geschäftsprozess- und Servicemodellierung. Dabei wird unter anderem den Fragen nachgegangen, inwiefern Geschäftsprozesse durch Services unterstützt werden können, welche Voraussetzungen in den Unternehmen dafür geschaffen werden müssen, welche Vor- und Nachteile sich bei der Verwendung von Services zur automatisierten Unterstützung von Prozessschritten ergeben und wie Geschäftsprozessmodelle als Ausgangspunkt zur Entwicklung von Servicemodellen verwendet werden können.

Der Hauptteil der Arbeit zeigt die Möglichkeiten zur Entwicklung organisationsübergreifender Prozesse und Services auf. In einer kurzen Motivation wird auf die wesentlichen Treiber eingegangen, die Unternehmen dazu bewegen, organisationsübergreifende Prozesse zu entwickeln, um Daten mit anderen Unternehmen auszutauschen. Im nächsten Schritt erfolgt eine Definition öffentlicher Prozesse und Services. Hierbei wird den Fragen nachgegangen, wann ein Prozess oder Service öffentlich ist, welche Eigenschaften öffentliche Prozesse und Services aufweisen und welche Rolle öffentliche Services innerhalb öffentlicher Prozesse einnehmen können. Zu diesem Zweck werden sowohl verschiedene öffentliche Prozessarten, als auch Servicearten charakterisiert. Einen kritischen Punkt bei öffentlichen Prozessen und Services bildet stets die Sicherheit und Vertraulichkeit beim Nachrichtenaustausch. Aus diesem Grund wird das Konzept einer serviceorientierten Sicherheitsarchitektur vorgestellt.

## Einleitung

Die folgende Abbildung stellt den schematischen Aufbau der Arbeit dar und enthält die inhaltlichen Zusammenhänge der wichtigsten Kapitel.

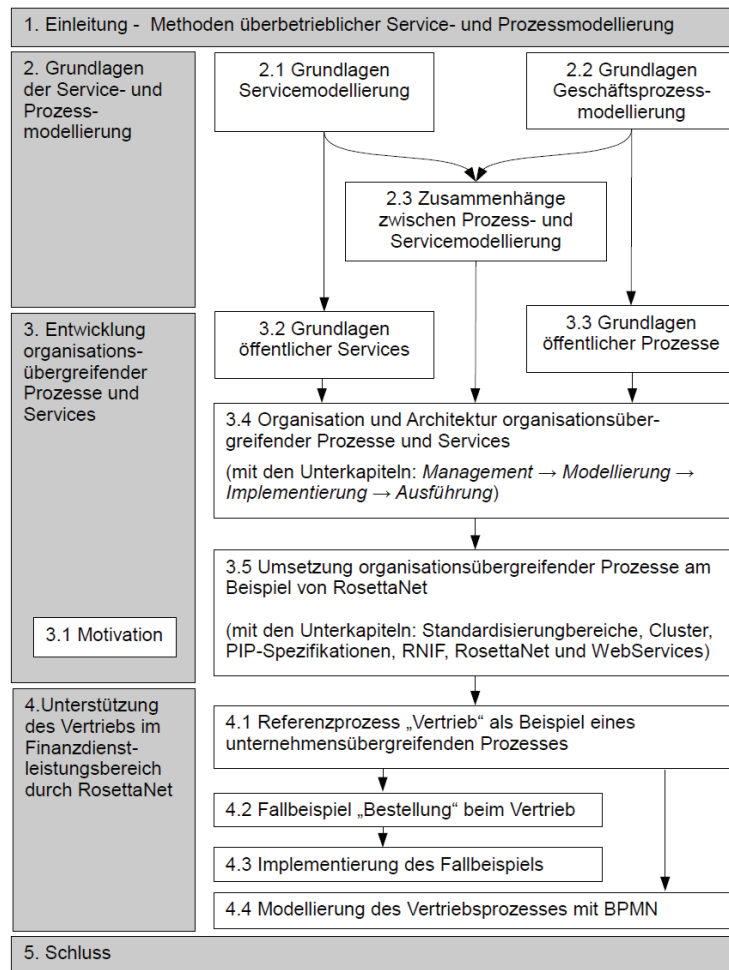


Abbildung 1: Schematischer Aufbau und Gliederung der Arbeit

Nachdem die Grundlagen öffentlicher Prozesse und Services betrachtet wurden, erfolgt eine strukturelle Beschreibung, wie organisationsübergreifende Geschäftsprozesse organisiert und eine entsprechende Architektur entwickelt werden kann. Zuerst wird im betriebswirtschaftlichen Kontext geklärt, welche unternehmensexternen Lösungsansätze die Entwicklung organisationsübergreifender Prozesse nötig machen, beispielsweise beim Outsourcing von Unternehmensleistungen oder Kooperationen. Aus der Art und Weise, wie die beteiligten Unternehmen sich zueinander verhalten, beispielsweise ob sie gleichberechtigt sind und ob eine zentrale Steuerungsinstanz vorhanden ist, ergeben sich entsprechende Implikationen für das organisationsübergreifende Prozessmanagement. Wer plant und verwaltet die Prozesse? Wer führt die Prozesse mit welchen Systemen aus? Wer hat Zugriff auf welche Daten und Informationen? Welche Werkzeuge, Sprachen, Standards und Notationen werden verwendet?

All diese Fragen werden im Folgenden geklärt, wenn nacheinander die *Modellierung*, *Implementierung* und *Ausführung* organisationsübergreifender Geschäftsprozesse genauer betrachtet werden. Bei der Modellierung wird zunächst auf die Unterschiede zwischen unternehmensinterner- und -übergreifender Modellierung eingegangen und deren Auswirkungen auf die Erstellung abstrakter und konkreter Prozessdefinitionen und Serviceschnittstellen. Einen weiteren Punkt bildet die Übertragung von fachlichen zu ausführbaren Prozessmodellen.

Bei der Implementierung wird der zentralen Frage nachgegangen, wie die Interaktion zwischen privaten und öffentlichen Prozessen geregelt werden kann, insbesondere das Nachrichtenrouting, und welche Implementierungsansätze unterschieden werden. Für die Ausführung von Geschäftsprozessmodellen stehen verschiedene Ansätze zur Verfügung, die sich im Wesentlichen in zentrale

oder dezentrale, sowie fachlich-aktivitätsorientierte und technisch-serviceorientierte Ausführungsvarianten unterscheiden und auf die genauer eingegangen wird. Anschließend erfolgt die kurze Besprechung einiger bekannter Ansätze und Sprachen, wie BPMN, BPML, BPEL, XPDL, ebXML und UBL.

Nachdem die Möglichkeiten der Organisation und Architektur organisationsübergreifender Geschäftsprozesse von einem theoretischen Standpunkt aus betrachtet wurden, erfolgt eine Untersuchung der Möglichkeiten und Methoden, die das RosettaNet-Framework zur Umsetzung solcher Geschäftsprozesse konkret bietet. Dabei wird auf die Entstehung und Ziele des RosettaNet Konsortiums, die vorhandenen Standardisierungsbereiche, Sichten und Ebenen, die vordefinierten Geschäftsprozesse (Partner-Interface-Processes), sowie deren Implementierung eingegangen. Damit wird die Grundlage für den letzten Teil der Arbeit geschaffen, in dem die Unterstützung eines Prozessschrittes aus dem Vertriebsprozess für den Finanzdienstleistungssektor durch einen konkreten Partner-Interface-Process erfolgt und programmtechnisch bis zur Implementierungs- und Ausführungsebene umgesetzt wird.

Für diesen Zweck wird zunächst der Vertriebsprozess und dessen Prozessbestandteile kurz erläutert und anschließend ein Vergleich (Matching) der vorhandenen RosettaNet-Partner-Interface-Processes mit den einzelnen Prozessschritten des Vertriebs durchgeführt. Dabei soll die Frage geklärt werden, welche Teile des Vertriebsprozesses durch RosettaNet unterstützt werden können. An einem konkreten Prozessschritt wird diese Unterstützung programmtechnisch bis zur Implementierungs- und Ausführungsebene umgesetzt. Dafür wird der ORACLE WebLogic 10gR3 Application Server verwendet.

## 2 Grundlagen der Service- und Prozessmodellierung

### 2.1 Grundlagen der Servicemodellierung

#### 2.1.1 Begriffsverständnis Serviceorientierung

Durch die große Anzahl unterschiedlicher Standpunkte und Sichtweisen sind zahlreiche Definitionen des Begriffes „Service“ vorhanden. Die in der Literatur gegebenen Begriffsbestimmungen nähern sich der Definition aus verschiedenen Richtungen, abhängig vom Verwendungskontext der beschriebenen Services auf fachlicher oder technischer Ebene.

Der Verwendung von Services liegt das Prinzip der Serviceorientierung zugrunde, bei dem es sich um eine Denkweise des Softwaredesigns beziehungsweise der Softwareentwicklung handelt, die jedoch auch für Vorgänge der wirklichen Welt Anwendung finden kann. Beispielsweise kapselt eine Sekretärin die Funktionen „Anrufe entgegennehmen“, „Termine organisieren“ und „Posteingang bearbeiten“ gegenüber einem oder mehreren Abnehmern, z. B. ihrem Chef, der diese Leistungen mit ihr vereinbart hat und von ihr in Anspruch nehmen kann. Masak formuliert das Paradigma der Serviceorientierung mit folgenden Worten:

*„Alle Funktionen in einem realen System, seien es Abläufe in Organisationen, Prozessen, Aktivitäten, Funktionen in Softwaresystemen, Applikationen, Teile von Applikationen oder Softwarefunktionen lassen sich als Service darstellen und aus Services aufbauen.“ [Masak, 2007,16]*

[Krafzig u. a., 2007, 36] definieren Services als „entfernt zur Verfügung gestellte, eigenständige Anwendungsmodule, die entweder Geschäftsfunktionalitäten oder technische Funktionalitäten unterstützen“. Ihre Aufgabe besteht darin, eine Funktionalität für einen oder mehrere andere Abnehmer verwendbar zu machen, indem diese gekapselt und über Schnittstellen zur Verfügung gestellt wird. Das W3C definiert Services folgendermaßen:

*“A service is an abstract resource that represents a capability of performing tasks that represents a coherent functionality from the point of view of provider entities and requests entities. To be used, a service must be realized by a concrete provider agent.” [W3C, 2004]*

Jeder Service hat, abhängig von der Abstraktionsebene, auf der er sich befindet und verwendet wird, einen Vertrag, bestehend aus einer nichtformalen Spezifikation des Zwecks, der Funktionalität und der Beschränkungen seiner Nutzung, sowie gegebenenfalls eine formale Spezifikation, die seine konkrete Verwendung ermöglicht. Der Vertrag enthält dabei eine oder mehrere Schnittstellen, mit denen die Funktionalität des Service seinen Verwendern, den Servicenutzern, über ein Netzwerk zur Verfügung gestellt wird. Dieses Netzwerk kann sowohl auf technischer Ebene, z. B. in Form der Daten- und Nachrichtenübertragung im Internet oder Intranet, als auch auf fachlicher Ebene verstanden werden, bei dem ein Service als Teilstück beziehungsweise Baustein innerhalb eines Prozesses verwendet wird. Alle Arten von Services befinden sich zwischen dieser abstrakten, semantischen Ebene und der technischen Ebene, das heißt der ausführbaren Serviceimplementierung. Diese stellt die angebotenen Funktionen und zugehörigen Daten physisch zur Verfügung und bildet damit die technische Realisierung des Servicevertrages. [Krafzig u. a., 2007, 78]

Services stellen Mechanismen für den Zugriff auf Ressourcen zur Verfügung und werden dadurch selbst zu einer verfügbaren Ressource, die mit anderen Ressourcen gekoppelt werden kann. Um die Entstehung und Verwendung eines Service in zeitlicher Hinsicht beschreiben zu können, ist ein Servicelebenszykluskonzept verfügbar. Dieses reicht vom gedanklichen Entwurf des Service, über seine nichtformale und formale Beschreibung, seine Veröffentlichung in einem Verzeichnis, dem Auffinden und Aufruf durch die Verwender, über seine Verwaltung während der Verwendung bis hin zu seiner Löschung oder Deaktivierung. [Casati u. a., 2001, 7]

[Masak, 2007, 13] beschreibt die Serviceorientierung als die logische Weiterentwicklung der Prinzipien der Objektorientierung, sowie der Komponentenorientierung und nähert sich dem Servicebegriff aus dem Blickwinkel der IT. Services schlagen aus seiner Sicht die Brücke zwischen *konkreter, forma-*



ler Schnittstellenbeschreibung, die zur Verwendung zwingend notwendig ist und *semantischer* Schnittstellenbeschreibung. Diese ergänzt die technische Beschreibung um Konzepte auf Geschäftsebene, mit denen die technischen Ausführungsbedingungen eines Service aus fachlicher Sicht beschrieben werden. Mit Bezug auf die vorangegangenen Definitionen lassen sich verschiedene Sichtweisen auf Services unterscheiden, auf die im Folgenden näher eingegangen wird.

*„Ein **Geschäftsservice** (business service) ist ein Element geschäftlichen Verhaltens. Er stellt eine geschäftliche Leistung dar, die ein Servicegeber gegenüber Servicenehmern erbringt. Der Servicegeber ist eine Einheit des Unternehmens, z. B. eine Abteilung oder einzelne Stelle. Servicenehmer sind Kunden oder andere externe Partner des Unternehmens oder andere Einheiten im gleichen Unternehmen. Jedem Geschäftsservice liegt ein Vertrag zugrunde. Dieser legt die ein- und ausgehenden Informationen und Güter fest. Er beschreibt die im Rahmen des Service durchzuführenden Schritte und ihre Reihenfolge, sofern sie für den Servicenehmer relevant sind. Diese Schritte heißen **Geschäftsserviceaktionen** (business service actions), kurz Aktionen.“*  
[Engels u. a., 2008, 96]

Mithilfe dieser Definition lassen sich Services auf fachlicher Ebene beschreiben. Die Definition enthält neben der „statischen“ Beschreibung eines Service auch dessen „dynamische“ Eigenschaft, zur Bearbeitung von Aktionen verwendet zu werden, beziehungsweise selbst Aktionen auszulösen. Auf die Grundlagen von Geschäftsprozessen wird unter Abschnitt 2.2 genauer eingegangen. Um die Abläufe im Unternehmen möglichst effizient zu gestalten, können die fachlichen Services durch den Einsatz von Anwendungen unterstützt werden. Daraus ergibt sich folgende Definition:

*„Ein **Anwendungsservice** (application service) ist ein Geschäftsservice oder ein Teil davon, der mittels IT von der Anwendungslandschaft erbracht wird. Die durchzuführenden Schritte, sofern für den Servicenehmer relevant, heißen **Anwendungsserviceaktionen** (application service actions).“* [Engels u. a., 2008, 155]

Zur Unterstützung von Anwendungsservices können im nächsten Schritt technische Services verwendet werden:

*„**Technische Services** (technical services) sind Dienstleistungen im Bereich der technischen Infrastruktur, die im Zusammenhang mit der Realisierung der physischen Anwendungslandschaft genutzt werden können. Sie werden von einer Integrationsplattform im Sinne einer Komponente der technischen Infrastruktur bereitgestellt.“* [Engels u. a., 2008, 198]

Die Unterscheidung von Services auf unterschiedlichen Abstraktionsebenen wird im Abschnitt 2.1.3 weiter fortgeführt und genauer detailliert. [Störrle und Glock, 2007, 79] folgen dieser „Dreiteilung“ der grundlegenden Servicearten und ergänzen die Beschreibung. Auf oberster Ebene betrachten sie Services als Gegenstand und Möglichkeit fachlicher Betrachtungen, mit deren Hilfe die Fachabteilungen ihre Anforderungen an die betrieblichen Abläufe sinnvoll strukturieren können. Auf mittlerer Ebene sind Services ein Mittel zur Anwendungsentwicklung, indem diese zur Darstellung und Implementierung komplexer Anwendungen aus aggregierten und komponierten Services verwendet werden. Die unterste Ebene bilden Services als Bausteine zum Betrieb der IT, indem diese zur Betriebszeit zur Verwaltung und Unterstützung der IT-Infrastruktur und komplexer Services genutzt werden können. Zusammenfassend lassen sich folgende Servicecharakteristiken feststellen:

Services ...

- ... sind für Außenstehende nutzbar, z. B. für andere Services, Personen, Anwendungen
- ... haben ein wohl definiertes Verhalten, welches für Anbieter und Verwender bekannt ist
- ... erfüllen gemäß Servicevertrag funktionale und nichtfunktionale Anforderungen
- ... sind durch ihre Schnittstellenbeschreibung eindeutig von der Umwelt abgegrenzt
- ... sind modellierbar und kombinierbar, um komplexe Anwendungsszenarien zu unterstützen
- ... werden von einem Anbieter zur Verfügung gestellt, von einem Verwender genutzt und gegebenenfalls in einem Verzeichnis erfasst

- ... haben definierte Ein- und Ausgaben, bei denen es sich um Informationen, Daten oder Nachrichten handeln kann

Zu den wesentlichen nichtfunktionalen Eigenschaften jedes Service gehört ein Serviceanbieter, der eindeutig identifizierbar ist, zum Beispiel eine rechtlich zuständige Person, Rolle, Abteilung oder Organisation, sowie einen Zeitpunkt oder Zeitabschnitt, in dem der Service verwendet werden kann. Weitere nichtfunktionale Eigenschaften sind seine Verfügbarkeit, Vertraulichkeit, Sicherheit, Performance, Preis, Zahlungsmodalitäten, Sprache und andere Qualitätsanforderungen im Rahmen von Service-Level-Agreements [Masak, 2007, 19]. Im nächsten Abschnitt wird die Rolle von Services im Rahmen serviceorientierter Architekturen genauer betrachtet.

### 2.1.2 Aufbau und Bestandteile serviceorientierter Architekturen

Die Verwendung von Services im Unternehmen geht im Idealfall mit der Entwicklung einer serviceorientierten Architektur einher. Eine serviceorientierte Architektur ist eine spezielle Ausprägung einer Softwarearchitektur für Unternehmen. Diese wird von Krafzig u. a. folgendermaßen definiert:

*„Eine **Softwarearchitektur** ist die Menge aller Aussagen, die Softwarekomponenten beschreiben und diesen Komponenten die Funktionalität des Systems zuordnen. Sie beschreibt die technische Struktur, Einschränkungen und Eigenschaften der Komponenten sowie der Schnittstellen zwischen diesen Komponenten. Die Architektur ist der Bauplan für das System und damit implizit der auf höchster Ebene angeordnete Plan für seine Realisierung.“* [Krafzig u. a., 2007, 75]

Werden Services als Komponenten betrachtet, die Funktionalitäten realisieren, entsteht die Notwendigkeit, besondere Architekturkonzepte einzuführen, die zur Verwaltung und dem Betrieb einer serviceorientierten Architektur nötig sind und diese von anderen Architekturvarianten abgrenzen:

*„Eine **serviceorientierte Architektur** (SOA) ist eine Softwarearchitektur, die auf den folgenden Schlüsselkonzepten basiert: Anwendungsfrontend, Service, Service-Repository und Service-Bus. Ein Service besteht aus einem Vertrag, einer oder mehreren Schnittstellen und einer Implementierung.“* [Krafzig u. a., 2007, 76]

Diese Definition einer SOA enthält bereits wichtige Punkte, ist aber noch nicht umfassend genug und wird noch durch mehrere andere Definitionen ergänzt. Die OASIS definiert eine SOA in folgender Weise:

*“A **Service Oriented Architecture** (SOA) is paradigm for organizing and utilizing distributed capabilities that may be under control of different ownership domains. From a holistic perspective, a SOA-based system is a network of independent services, machines, the people who operate, affect, use, and govern those services as well as the suppliers of equipment and personnel to these people and services.”* [OASIS, 2008, 10]

Anhand der in der Literatur verfügbaren Definitionen sind techniknahe und technikferne Begriffsbestimmungen zu finden, abhängig vom Standpunkt der Autoren. Sowohl eine rein technische Definition, als auch eine technikferne Definition sind jedoch nicht zielführend. Eine SOA kann als Mittel zum Zweck verstanden werden, Unternehmensressourcen auf eine sinnvollere Art und Weise verfügbar zu machen, als dies mit anderen Architekturvarianten möglich wäre, um dem Unternehmen Wettbewerbsvorteile zu ermöglichen, die es ohne dem Einsatz einer SOA nicht hätte. Die Entwicklung und der Einsatz von Services zur Erfüllung dieses Zwecks erscheinen sinnvoll, jedoch kann abhängig von den Bedürfnissen eines Unternehmens auch eine andere Architekturvariante zweckmäßiger erscheinen. Der Einsatz einer SOA bewirkt damit nicht pauschal Vorteile für jedes Unternehmen.

Nach [Störrle und Glock, 2007, 81] besteht der wesentliche Vorteil einer SOA in ihrer zugrunde liegenden Idee beziehungsweise Strategie, die Beschreibung des Geschäfts vor ihrer Umsetzung in der Technik vorzunehmen. Das Geschäft wird dabei durch eine Reihe von fachlichen Modellen repräsentiert, die durch eine Folge von Umformungs- und Anreicherungs-schritten in eine technische Repräsentation gebracht, anschließend implementiert und in Betrieb genommen wird. Demzufolge bildet eine SOA sowohl das Bindeglied, als auch die Grundlage der Ausrichtung der Anwendungslandschaft an die Anforderungen des Unternehmens. Es bleibt festzuhalten, dass eine SOA durch bestimmte Eigen-

schaften charakterisiert ist, die sie von anderen Architekturvarianten abgrenzt und sich diese Eigenschaften auf die Art und Weise beziehen, wie Services zur Unterstützung der betrieblichen Abläufe eingesetzt werden können.

Zu diesem Zweck enthält eine SOA mehrere grundlegende Bausteine. Diese sind die eigentlichen *Services*, der *Service-Bus*, sowie eine *Serviceverzeichnis*, beispielsweise in Form einer Service-Registry oder eines Service-Repository. Das Serviceverzeichnis besteht in der Regel aus einer Datenbank, in der die Beschreibungen der Services gespeichert werden. Dadurch wird der Austausch von Services zwischen verschiedenen Anwendungsbereichen ermöglicht und somit deren Wiederverwendbarkeit gefördert. Je nachdem, in welchem Umfeld die modellierten Services verwendet werden sollen, kann das Serviceverzeichnis öffentlich, das heißt unternehmensübergreifend, oder nur unternehmensintern verfügbar sein [vom Brocke, 2008, 15ff.]. Im Detail wird dies im Kapitel 3 besprochen.

Der Service-Bus stellt das Verbindungsglied der SOA dar und dient der Kommunikation der Services untereinander, als auch zwischen den Services und dem Serviceverzeichnis. Er bildet damit die Grundlage zur Realisierung der Ablaufsteuerung innerhalb der SOA und damit den Kern der Architektur. Bei Huvar u. a. unterstützt der Service-Bus weiterhin verschiedene Kommunikationsprotokolle und Interaktionsparameter für den Nachrichtenaustausch zwischen den Services. Dadurch wird die Implementierung der Services abstrahiert und verborgen, wodurch die Heterogenität der IT-Landschaft teilweise behoben wird. Weiterhin übernimmt der Service-Bus das Mapping von Strukturen und Datenformaten. Der Service-Bus verbindet dadurch die Servicelaufzeit (Kommunikation, Zustandsverwaltung) auf der einen Seite, mit der Integrationslaufzeit (Routing, Geschäftsprozessausführung) auf der anderen Seite [Huvar u. a., 2008, 55]. Weitere Bestandteile einer SOA sind beispielsweise Modellierungs- und Monitoringwerkzeuge [vom Brocke, 2008, 15ff.]. Durch die hohe Komplexität einer SOA und einer potenziell hohe Anzahl von Services erscheint die Unterteilung einer SOA in mehrere Schichten oder Ebenen sinnvoll. Krafzig u. a. unterscheiden dabei vier grundlegende Schichten, auf denen sich Services eines bestimmten Typs befinden.

Die *Unternehmensschicht* enthält Anwendungsfrontends und öffentliche Services, die als Zugriffspunkte auf die SOA dienen. Darunter befindet sich die *Prozessschicht* mit prozesszentrierten Services, die zur Entwicklung komplexer Geschäftsprozesse verwendet werden können und sich zur Erledigung ihrer Aufgaben den Services der *Zwischenschicht* bedienen. Auf dieser Schicht sind Fassaden, Technologie-Gateways und Adapter anzutreffen. Die niedrigste Ebene bildet die *Basisschicht* mit atomaren Services, die den Zugriff auf die Unternehmensdaten- und Anwendungen realisieren [Krafzig u. a., 2007, 99f.]. Eine genauere Unterscheidung der verschiedenen Servicearten erfolgt im Abschnitt 2.3.

Ein weiteres Unterscheidungs- und Charakterisierungsmerkmal in Bezug auf die SOA bildet ihr Reife- bzw. Entwicklungsgrad. Die Einführung einer SOA geschieht nicht über Nacht, denn sie stellt kein fertiges Produkt dar, welches gekauft und installiert werden kann, sondern eine langfristige IT-Managementstrategie mit einer Reihe von Entwicklungsschritten. Krafzig u. a. unterscheiden in diesem Zusammenhang zwischen der *grundlegenden SOA*, der *vernetzten SOA* und, in ihrer höchsten Entwicklungsstufe, der *prozessfähigen SOA* [Krafzig u. a. 2007, 105ff.].

Jeder der Entwicklungsstufen erhöht die Komplexität der SOA, erweitert im Gegenzug jedoch ihre Einsatzmöglichkeiten. Erst in der letzten Phase der prozessfähigen SOA, die am schwierigsten zu erreichen ist und deren Entwicklung mehrere Jahre dauern kann, unterliegt die Steuerung der Unternehmensprozesse der SOA selbst und nicht mehr den Anwendungen. In dieser höchsten Phase bilden nicht mehr die Anwendungen den Ausgangspunkt für die Abwicklung der Geschäftsabläufe, sondern die SOA, welche die Anwendungen zu diesem Zweck verwendet. Hat das Unternehmen diese Phase erreicht, dann besteht die Möglichkeit unternehmensübergreifende Prozesse zu entwickeln. In Abschnitt 3.4 wird dies genauer beschrieben. Die folgende Abbildung verdeutlicht diesen Zusammenhang.

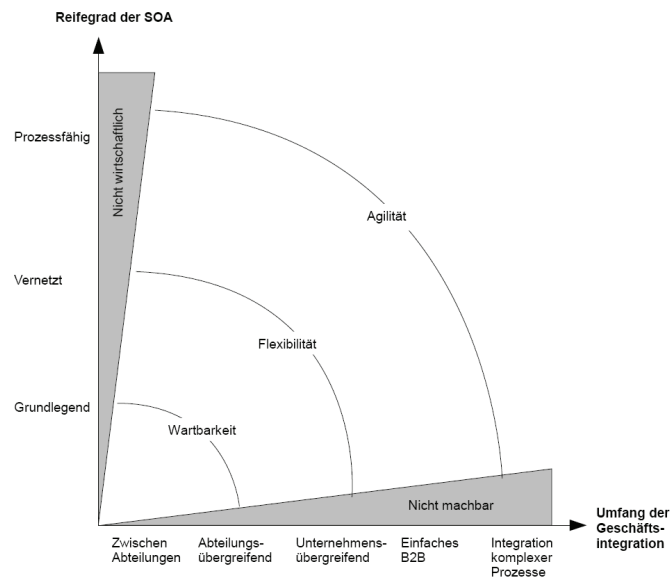


Abbildung 2: Reifegrade der SOA in Bezug auf unternehmensübergreifende Prozesse in Anlehnung an [Krafzig u. a., 2007, 105]

### 2.1.3 Servicemodellierung im Rahmen der Serviceentwicklung

Im vorherigen Abschnitt wurde bereits darauf eingegangen, dass die Entwicklung und Verwendung von Services in mehrere Phasen unterteilt werden kann, die als Servicelebenszyklus bezeichnet werden. Der Schwerpunkt dieser Arbeit liegt auf der Phase der Modellierung.

Nach [Masak, 2007, 269] kann die Serviceentwicklung in mehrere Phasen untergliedert werden, die entsprechend der klassischen Softwareentwicklung als Analyse-, Design-, und Implementierungsphase bezeichnet werden. In welcher Reihenfolge, Häufigkeit und Ausprägung diese Phasen durchlaufen werden sollen wirft die Frage nach einem passenden Vorgehensmodell für die Serviceentwicklung auf. Entsprechend der bisherigen Erfahrungen im Bereich der Serviceentwicklung erscheint die Anwendung des Wasserfallmodells nicht zweckmäßig, bei dem die drei eben genannten Phasen nur einmal hintereinander durchlaufen werden. Stattdessen betont Masak die Notwendigkeit, diese Phasen wie ein Zyklus zu wiederholen.

Dies ergibt sich bereits aus der Tatsache, dass Services meist in dynamischen und flexiblen Umgebungen eingesetzt werden sollen, bei denen die Anforderungen und Schnittstellen im Vorherein fast nie vollständig spezifiziert werden können, was für die Anwendung des Wasserfallmodells jedoch zwingend notwendig wäre. Weiterhin ist die Zeit, um den Zyklus aus Analyse, Design und Implementierung zu durchlaufen, deutlich kürzer als bei der klassischen Anwendungsentwicklung, wodurch ihre Wiederholung erleichtert wird. Dies liegt insbesondere am Granularitätsniveau und dem Umfang, den die einzelnen Services annehmen, der in der Regel deutlich geringer ist, als für Anwendungsprogramme. Um diesen Entwicklungsbedingungen gerecht zu werden, empfiehlt Masak agile Softwareentwicklungsverfahren für die Serviceentwicklung, wie beispielsweise das eXtreme Programming oder Scrum [Masak, 2007, 270].

Die erste Phase der Serviceentwicklung ist die Analyse. Das Ziel der Analyse ist es herauszufinden, welche Services benötigt werden, um eine Menge konkreter, fachlicher Anforderungen einer Domäne möglichst effizient zu befriedigen. Diese Frage lässt sich in der Regel erst beantworten, wenn die fachlichen Anforderungen möglichst umfangreich, jedoch nicht zwangsläufig vollständig, erfasst und dokumentiert wurden. Eine Möglichkeit, dieses Ziel zu erreichen, besteht in der Verwendung von Geschäftsprozessmodellen, worauf im nächsten Abschnitt genauer eingegangen wird. Das Ziel der Analyse ist es demnach nicht, einzelne stehende, geschlossene Anwendungen zu entwickeln, sondern den Service so zu konzipieren, dass er eine konkrete Anforderung befriedigt und darüber hinaus auch in weiteren serviceorientierten Systemen und Szenarios Verwendung finden kann, die zum Entwicklungszeitpunkt noch nicht bekannt sind.

Im Gegensatz zu monolithischen Anwendungen, die für sich allein stehend eine spezielle Aufgabe erfüllen, ohne auf Daten anderer Anwendungen angewiesen zu sein, oder eigene Daten an fremde Anwendungen weitergeben zu müssen, ergibt sich der Nutzen von Services, und damit einer SOA, zu einem großen Teil aus der nahezu universellen Einsetzbarkeit der vorhandenen Services in verschiedenen Zusammenhängen, je nach Bedarf auf fachlicher Ebene, das heißt den Anforderungen des Geschäfts. Eine gute Möglichkeit diesem Bedürfnis gerecht zu werden liegt in der Unterscheidung verschiedener Servicearten zur Befriedigung der unterschiedlichen Anforderungen. Damit geht auch die Festlegung des Granularitätsniveaus für die Services einher. Siehe dazu auch [Masak, 2007, 271].

Für die Phase des Designs gelten ebenfalls verschiedene Anforderungen, die während der Analyse aufgestellt und erfasst wurden. Die gute Wiederverwendbarkeit der Services kann eine dieser Anforderungen sein, muss es jedoch nicht zwangsläufig. Eine gute Charakteristik im Servicedesign ist beispielsweise die klare Zuordnung der in der Serviceschnittstelle aufgeführten Methode zu ihrem fachlichen Verwendungskontext, anstatt reine Datenzugriffe darüber abzuwickeln, d. h. die Serviceschnittstelle sollte durch die Elemente des Domänenmodells sinnvoll beschrieben werden. Dazu kann es zweckmäßig sein, die Services zustandslos zu halten, um den Zugriff aus verschiedenen Zusammenhängen zu ermöglichen.

Weitere, wichtige Designregeln sind die klare Trennung von funktionalen und organisatorischen Aspekten, das Design zur Erweiterbarkeit des Services, indem eine lose Kopplung und späte Bindung angestrebt wird, sowie die Einhaltung der vorhandenen Systemgrenzen. Damit ist gemeint, dass ein Service keine Annahmen über etwas außerhalb seiner Grenzen machen darf, um auf eine festgelegte Weise zu funktionieren, wodurch beispielsweise die Verwendung globaler Variablen untersagt ist. Beim Entwurf der Services muss davon ausgegangen werden, dass sich seine Umwelt jederzeit ändern kann, wie es aufgrund der heutigen Anforderungen des Marktes an die Unternehmen häufig gegeben ist [Masak, 2007, 272f.].

Eine wichtige Entscheidung im Rahmen des Designs ist die Festlegung der für die Serviceentwicklung zu verwendenden Standards. In der Literatur wird derzeit die Verwendung von internetbasierten Standards- und Protokollen zur Bildung von WebServices sehr intensiv diskutiert. Dabei sollte jedoch nicht außer Acht gelassen werden, dass die Entwicklung von WebServices nicht mit der Entwicklung von Services im Allgemeinen gleichzusetzen ist. Vielmehr stellt die Verwendung von WebService-Standards eine mögliche Entscheidung des Servicedesigns dar, wie die Implementierung der Services geschehen kann. Eine Reihe weiterer Eigenschaften von Services kann dem morphologischen Kasten der Servicemerkmale und deren Merkmalsausprägungen entnommen werden, der von Overhage und Turowski entwickelt wurde. Auf die Implementierungsphase wird im 3. Kapitel näher eingegangen.

<b>Merkmal</b>	<b>Merkmalsausprägung</b>		
<b>Abrechnung/Kosten</b>	kostenloses Angebot		nutzungsabhängig   Festpreis
<b>Granularität</b>	Einzelleistung		Menge an Einzelleistungen   Leistungsbündelung
<b>Nutzung weiterer Services</b>	Nutzung anderer Services		keine Nutzung anderer Services
	Ad-Hoc-Bündelung	feste Verknüpfung	
<b>Zustand</b>	zustandslos		zustandsbehaftet
<b>Integration</b>	Ad-Hoc-Integration		feste Verknüpfung
<b>Automatisierungsgrad</b>	vollständig		teilweise

Tabelle 1: Morphologischer Kasten für Merkmalsausprägungen von Services

in Anlehnung an [Overhage und Turowski, 2007, 3]

### 2.1.4 Unterscheidung von Servicearten

Wie bereits im Abschnitt zu den serviceorientierten Architekturen angedeutet wurde, ist eine Untergliederung der zu entwickelnden Services in verschiedene Servicetypen bzw. Servicearten sinnvoll. Jede der Servicearten erfüllt dabei bestimmte Eigenschaften und ermöglicht eine genaue Zuordnung der Services zu einem Verwendungskontext und seinen Ausführungsbedingungen.

Krafzig liefert eine Reihe von Gründen, die eine solche Unterteilung notwendig machen. Die Serviceklassifikation dient zur Bildung einer gemeinsamen Sprache der an der Entwicklung einer SOA beteiligten Personen und Institutionen. Durch die Zuordnung von Serviceeigenschaften, beispielsweise der geplanten Nutzungshäufigkeit oder Einbindung in unternehmenskritische Prozesse mit speziellen Performanceanforderungen kann eine Einschätzung des zukünftigen Entwicklungsaufwands erfolgen. Die Implementierungs- und Wartungskosten sind in der Regel nicht für alle Services gleich groß, sondern schwanken je nach Umfang und Art der Anforderungen, die der Service bedienen soll. Darüber hinaus kann je nach Service eine unterschiedliche Implementierungsstrategie sinnvoll sein, die sich aus seinen geforderten Eigenschaften ableitet. Es ergeben sich bei zustandsbehafteten Services möglicherweise andere Hardwareanforderungen, als bei zustandslosen Services, weil gegebenenfalls mehr Speicherbereich zur Verfügung gestellt werden muss. Als entscheidender Vorteil der Serviceklassifizierung ist jedoch die Möglichkeit der vertikalen Partitionierung zu nennen, um die Komplexität einer SOA mit Hunderten oder Tausenden von Services beherrschbar zu machen und sie in überschaubare Teile zerlegen zu können [Krafzig u. a., 2007, 86ff.].

Nach [vom Brocke, 2008, 65] kann auf niedrigster Abstraktionsebene zwischen elementaren Services und komplexen Services unterschieden werden. Elementare Services werden dabei unmittelbar durch die ihnen zugeordneten Ressourcen realisiert und bedürfen keiner anderen Services zur Erfüllung ihrer Funktionen. Demgegenüber erfordert der Aufruf komplexer Services die Ausführung eines Prozesses, bei dem mehrere weitere Ressourcen oder Services beteiligt sein können. Der komplexe Service dient damit als Schnittstelle auf einen oder mehrere andere (elementare) Services, wobei die Konzepte der Choreografie und Orchestrierung zum Einsatz kommen können.

Masak und Krafzig unterscheiden auf höherer Abstraktionsebene vier verschiedene Servicetypen, die im Folgenden kurz erklärt werden. Siehe hierzu [Masak, 2007, 112 ff. und Krafzig u. a., 2007, 87]. Auf der untersten Stufe liegen „Basisservices“, die den elementaren Services entsprechen. Diese sind zustandslos und bilden das obligatorische Fundament der SOA. Sie können weiterhin in datenzentrierte Services zum Zugriff auf Datenbanken oder ERP-Systeme verwendet werden, um persistente Daten zu verwalten und in logikzentrierte Services, um Aspekte der fachlichen Domäne, wie zum Beispiel Geschäftsregeln, komplexe Berechnungen und Algorithmen zu realisieren. Wird der elementare Service durch eine Anwendungskomponente realisiert, stellt dies einen Anwendungsservice dar [Engels u. a., 2008, 102].

Auf der zweiten Stufe kommen „Zwischenservices“ (nach Masak „Utilityservices“) zum Einsatz. Diese dienen der Überbrückung von Unterschieden in der technischen Infrastruktur und fungieren als technologische Gateways, Adapter, Fassaden oder funktionalitätserweiternde Services und sind ebenfalls zustandslos. Zur Unterstützung komplexer Prozessketten können auf der dritten Stufe so genannte „Prozessservices“ (nach Krafzig „Prozesszentrierte Services“) verwendet werden, in denen die Prozesskomplexität- und Logik des Unternehmens gekapselt werden kann. Auf diese wird im Abschnitt 2.3 detailliert eingegangen, nachdem die Grundlagen zum Verständnis von Geschäftsprozessen gelegt wurden. Auf höchster Stufe sind öffentliche Unternehmensservices (nach Masak „Enterpriseservices“) anzutreffen, die ein Unternehmen seinen Partnern oder Kunden zur Verfügung stellen kann und im Abschnitt 3.2.2 genauer verdeutlicht werden.

	<b>Basisservices</b>	<b>Zwischenservices</b>	<b>Prozessservices</b>	<b>Öffentliche Services</b>
<b>Beschreibung</b>	Elementare, datenzentrierte oder logikzentrierte Services	Services als Technologie-Gateways, Adapter oder Fassaden	Services zur Kapselung von Prozesslogik	Anderen Unternehmen zur Verfügung gestellte Services
<b>Implementierungskomplexität</b>	gering bis mittel	mittel bis hoch	hoch	serviceabhängig
<b>Statusverwaltung</b>	statuslos	statuslos	statusbehaftet	statusbehaftet
<b>Wiederverwendbarkeit</b>	hoch	gering	gering	hoch
<b>Änderungshäufigkeit</b>	gering	mittel bis hoch	hoch	gering
<b>Obligatorischer Bestandteil der SOA</b>	ja	nein	nein	nein

Tabelle 2: Unterscheidung von Servicearten

nach [Masak, 2007, 112] und [Krafzig u. a., 2007, 87]

Overhage und Turowski unterscheiden demgegenüber nur drei verschiedene Servicearten. Auf niedrigster Stufe befinden sich Informationsservices, die ihrem Verwender lediglich Informationen zur Verfügung stellen, ohne Zustands- oder Datenänderungen im System oder bei anderen Services vorzunehmen, beispielsweise die Ausgabe eines Aktienkurses. In der mittleren Stufe befinden sich Funktionsservices, bei denen Informationen verarbeitet und Änderungen im System vorgenommen werden. Es ist dabei nicht näher spezifiziert, ob diese Änderungen mit fachlichem oder technischem Hintergrund geschehen. Auf höchster Stufe liegen auch bei ihnen die Prozessservices, die mehrere Einzelleistungen anderer Services integrieren und zu einer komplexen Leistung zusammenfügen [Overhage und Turowski, 2007, 3]. Die Betrachtung der Serviceorientierung endet an dieser Stelle, wird jedoch im Abschnitt 2.3 wieder aufgegriffen, wenn das Zusammenspiel von Geschäftsprozessen und Services untersucht wird.

## 2.2 Grundlagen der Geschäftsprozessmodellierung

### 2.2.1 Einordnung in den Rahmen des Geschäftsprozessmanagements

Zur Einordnung des Geschäftsprozessmanagements in die Unternehmensstruktur ist zunächst die Unternehmensarchitektur zu betrachten. Nach Schreiter und Laures kann ein Schichtenmodell zur Gliederung der Unternehmensarchitektur angewendet werden [Schreiter und Laures, 2006, 149]. Auf der obersten Ebene befindet sich die *Rollenschicht*, in der interne und externe Organisationseinheiten vorhanden sind, die auf den Betrieb des Unternehmens einwirken. Dazu zählen beispielsweise Kunden, Abteilungen, Abteilungsmitarbeiter, Manager, Lieferanten. Diese greifen auf die unternehmensinternen IT-Systeme über die *Präsentationsschicht* zu, die sich auf der zweiten Ebene befindet. Dazu gehören Portale, Webanwendungen, Rich-Clients, E-Mail-Clients und Benutzeroberflächen der Anwendungsprogramme. Auf der mittleren Ebene liegt die *Geschäftsprozessschicht*, die als Verbindungsglied zwischen fachlicher und technischer Welt dient. Sie beinhaltet Geschäftsprozessmodelle, Prozessinstanzen, Tasks, Business Process Engines und Workflow-Managementsysteme. Auf der vierten Ebene erstreckt sich die *Serviceschicht*, die selbst keine Geschäftsfunktionalität implementiert, sondern diese mithilfe von Services auf Grundlage der darunter liegenden Backend-Schicht verfügbar macht.

Die Serviceschicht kann durch die Entwicklung einer SOA realisiert werden. Auf der untersten Ebene befindet sich die *Backend-Schicht*, in der alle Geschäftsfunktionalitäten implementiert sind. Sie kann darüber hinaus auch den Zugriff auf weitere Unterstützungssysteme, wie ERP- und EAI-Systeme realisieren. Die Geschäftsprozessschicht stellt ein Ergebnis und Tätigkeitsfeld des Geschäftsprozessmanagements dar, welches sich in die Bereiche Geschäftsprozessabgrenzung-, Modellierung-, Ausführung-, Controlling- und Optimierung untergliedern lässt.

Nach [Gadatsch, 2005, 2f.] umfasst der Bereich des Geschäftsprozessmanagements auf strategischer Ebene die Unternehmensstrategie, auf fachlich-konzeptioneller Ebene das Prozessmanagement und auf operativ-technischer Ebene das Workflow-Management. Die folgende Tabelle verdeutlicht den Zusammenhang anhand einer Übersicht von Peter und Wyss:

Ebene	Tätigkeit	Ergebnis	Akteur
strategische Ebene	Strategieentwicklung	Geschäftsfeldstrategie	Strategisches Management
fachlich-konzeptionelle Ebene	Prozessmodellierung	Geschäftsprozessmodell	Geschäftsprozessmanagement
operative Ebene	Workflowmodellierung	Workflowmodell	Workflowmanagement
technische Ebene	Servicemodellierung	Service-Architekturmodell	SOA Service Competence Center/ SOA Board

Tabelle 3: Ebenenkonzept der Prozessmodellierung

in Anlehnung an [Peter und Wyss, 2005, 4]

Während der *Prozessabgrenzung* wird versucht, die zu unterstützenden und damit zu modellierenden Prozesse zu identifizieren. Dies beinhaltet die Prozessentstehung auf Grundlage der vorhandenen Geschäftsfelder, aus denen die Prozesskandidaten ausgewählt und später abgebildet werden.

Die *Prozessmodellierung* verfolgt das Ziel, die Geschäftsprozesse eines gewählten Realitätsausschnittes, d. h. Geschäftsfeldes, unter einer fachlich-konzeptionellen Perspektive zu betrachten. Ausgehend von den strategischen Zielen des Geschäftsfelds erfolgt entweder die Anpassung vorhandener Prozesse oder die Prozessneugestaltung bzw. Prozessentwurf. Das Ergebnis der Prozessmodellierung ist damit das Prozessmodell.

Im Zuge der *Prozessoptimierung* erfolgt die Ausrichtung der Prozesse an vorgegebenen Messgrößen, den so genannten Prozessführungsgrößen, die sich aus den kritischen Erfolgsfaktoren der Geschäftsfelder ableiten lassen. Zu diesem Zweck wird das Prozessmonitoring eingesetzt. Je nach den ermittelten Erfolgswerten oder Schwachstellen kann eine erneute Prozessmodellierung notwendig sein [Gadatsch, 2005, 2f.]. Auf die Bereiche der Prozessoptimierung und des Prozesscontrollings wird in dieser Arbeit nicht näher eingegangen, den Schwerpunkt bildet die Geschäftsprozessmodellierung.

Abecker u. a. betonen die Notwendigkeit einer einheitlichen Modellierungsorganisation, um eine hohe Qualität der Geschäftsprozessmodelle zu gewährleisten [Abecker u. a., 2002, 112ff.]. Viele der Probleme im Zusammenhang mit der Geschäftsprozessmodellierung ergeben sich daraus, dynamische, sich häufig verändernde Systeme modellieren zu müssen, die sich den klassischen Methoden der Modellierung teilweise entziehen. Je nach Modellierungsszenario ergibt sich eine Vielzahl verschiedener Modellierungsansätze mit unterschiedlicher Modellierungsreichweite-, Zielen- und Methoden. Darüber hinaus sind die Anwendungsbereiche der Prozessmodellierung ebenfalls verschieden, beispielsweise die Geschäftsprozessoptimierung, das Business Process Reengineering, die Gestaltung von Informationssystemen, die Verbesserung des Qualitätsmanagements oder die Einführung eines Workflow-Managementsystems. Das Zusammenspiel aus Modellierungsorganisation-, Methoden- und Unterstützung wird durch folgende Abbildung verdeutlicht.

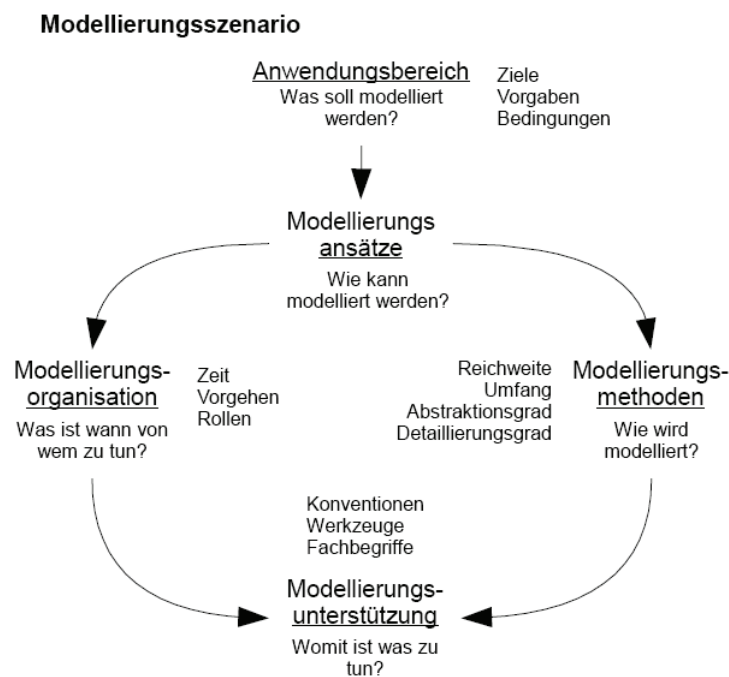


Abbildung 3: Bestandteile des Modellierungsszenarios in Anlehnung an [Abecker u. a., 2002, 112]

Die *Modellierungsorganisation* widmet sich dabei der Erfassung der Prozessabläufe, das heißt des Prozesswissens der Mitarbeiter und deren Kommunikationsstruktur. Ihre Aufgabe ist die systematische Erstellung und kontinuierliche Aktualisierung der Modelle, das heißt deren regelmäßige Anpassung an die Änderungen der Geschäftsabläufe. Die *Modellierungsmethoden* können durch ihre Reich-



weite und den Modellumfang beschrieben werden, hinsichtlich ihrer Sichten, Ebenen, sowie des Detaillierungs- und Abstraktionsgrades. Im Abschnitt 2.2.4 wird auf die vorhandenen Modellierungsmethoden genauer eingegangen. Parallel zu den Modellierungsmethoden dienen die Regelungen im Rahmen der *Modellierungsunterstützung* für einheitliche Modellierungskonventionen. Diese beinhalten die zu verwendenden Modellierungstechniken zur Reduzierung der Modellvariabilität und Erhöhung der Modellqualität, sowie die Festlegung von Fachbegriffen und Ontologien zur Schaffung einer einheitlichen Begriffswelt für die Verbesserung der Kommunikation während des Modellierungsprozesses. Weiterhin werden die zu verwendenden Software-Werkzeuge, das Vorgehensmodell und gegebenenfalls die zu verwendenden Referenzmodelle festgelegt.

Nach [Scheer und Jost, 1996, 40] enthält ein Vorgehensmodell die innerhalb der Projektphasen durchzuführenden Aktivitäten („Was ist zu tun, um die Ziele zu erreichen?“), deren Zielsetzungen („Welche Ergebnisse sollen erzielt werden?“), die einzusetzenden Maßnahmen und die dafür benötigten Beschreibungstechniken. Hangos und Cameron haben sich intensiv mit der Systematisierung der Prozessmodellierung beschäftigt und unterteilen das Vorgehensmodell zur Prozessmodellierung in sieben verschiedene Schritte [Hangos und Cameron, 2001, 24ff.]. Die Ausgangsbasis der Prozessmodellierung bildet das Prozesssystem in der wirklichen Welt, das Modellierungsziel, sowie die Menge aller Validierungskriterien zur Prüfung, ob das Modellierungsziel erreicht wurde.

Im ersten Schritt geschieht die *Problemdefinition*. Dabei werden alle Ein- und Ausgaben des Prozesses erfasst, worunter beispielsweise Daten, Dokumente, Informationen, sowie Waren- und Güterströme zählen. Weiterhin wird die Anzahl der notwendigen Hierarchiestufen und deren Detaillierungsgrad festgelegt. Ziel dabei ist es herauszufinden, welche Objekte während des Prozesses benötigt und durch ihn erzeugt werden. Der zweite Schritt bildet die *Identifizierung der kontrollierenden Faktoren*, das heißt aller Objekte, die auf den Prozess einwirken, wie beispielsweise Mitarbeiter, Abteilungen, Kunden, Lieferanten. Im dritten Schritt werden die zur automatisierten Unterstützung des Prozesses benötigten *Prozessdaten*, sowie deren Werte und Parameter erfasst, um die Transformation des (semi-) formalen Prozessmodells in eine ausführbare Sprache zu unterstützen. Es wird damit die Frage geklärt, welche Daten von welchem am Prozess beteiligten Objekt benötigt werden. Als vierten Schritt nennen Hangos und Cameron die *Entwicklung von Prozessbewertungsmöglichkeiten*, indem eine Menge von Kriterien aufgestellt wird, anhand derer die Messung der Prozesseffizienz geschehen kann. Dazu zählen die für jeden Prozessschritt benötigte Zeit, Kosten und andere Ressourcen.

Im fünften Schritt erfolgt die eigentliche, ggf. grafisch unterstützte *Prozessmodellierung*, das heißt das Auffinden einer Lösungsprozedur zur Erreichung des beschriebenen Ziels. Dabei werden alle während des Prozesses erzeugten und benötigten Objekte in einen zeitlich-logischen Zusammenhang zu den auf den Prozess wirkenden Objekten gebracht. In Abhängigkeit der verwendeten Prozessmodellierungssprache kommen dabei die Konzepte Ereignis, Bedingung, Aktion, Und, Oder, Verzweigung, Vereinigung, Parallelisierung und der gleichen zum Einsatz. Auf die verschiedenen Prozessmodellierungssprachen wird im Abschnitt 2.2.4 genauer eingegangen.

Der sechste und siebente Schritt besteht aus der *Verifizierung*, ob der Prozess korrekt abläuft, und der *Validierung*, ob der Prozess, im Hinblick auf die benötigten Ressourcen, effizient abläuft. Dabei erfolgt der Vergleich des Prozessverhaltens mit dem Modellverhalten, das heißt der Abgleich der Simulationsdaten mit den realen Prozessdaten [Hangos und Cameron, 2001, 24ff.]. Es sollte jedoch beachtet werden, dass die oben beschriebene Vorgehensweise nur eine von vielen Möglichkeiten darstellt, aus der Strategie des betrachteten Geschäftsfeldes die benötigten Prozesse abzuleiten. Die Geschäftsprozessmodellierung bildet die Grundlage zur Unterstützung der Geschäftsabläufe durch die IT. Das Zusammenspiel aus Geschäftsprozessmodellierung und Workflow-Modellierung zeigt folgende Abbildung.

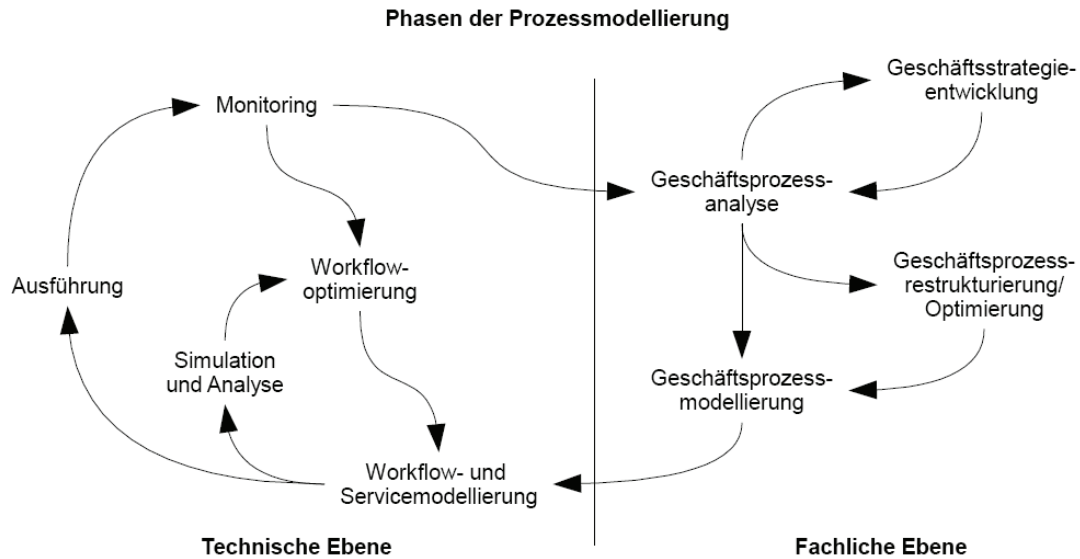


Abbildung 4: Phasenkonzept der Prozessmodellierung  
in Anlehnung an [Gadatsch, 2005, Abb. 36, S. 52]

Im nächsten Abschnitt wird auf die Notwendigkeit und den Zweck der Geschäftsprozessmodellierung genauer eingegangen.

### 2.2.2 Motivation und Ziel der Geschäftsprozessmodellierung

Mit dem Einsatz der Geschäftsprozessmodellierung sind verschiedene Zielstellungen verbunden. Oesterreich u. a. haben sich ausführlich mit der Frage beschäftigt, was Unternehmen dazu bewegen kann, die unternehmensinternen Prozesse mithilfe einer Prozessbeschreibungssprache zu erfassen. Die Struktur und Gestaltung des Arbeitsablaufes beeinflussen entscheidend die Verwirklichung der Unternehmensziele. Um den Arbeitsablauf jedoch bewerten und gegebenenfalls verbessern zu können, muss er zuvor erst einmal verstanden worden sein. Unter Zuhilfenahme der Geschäftsprozessmodellierung kann ein Unternehmen eben dieses Verständnis gewinnen. Daraus ergeben sich nur Vorteile für den betriebswirtschaftlichen Bereich, sondern auch implizit für die Gestaltung und Entwicklung der IT-Systeme.

Bis zur Mitte der 90er Jahre lag der Schwerpunkt zur Verbesserung der Unternehmensleistung auf der Produktivitätssteigerung durch eine bessere Automatisierung der internen Abläufe. Durch den Einsatz moderner Informations- und Kommunikationstechnologien konnte ein Großteil der automatisierbaren Abläufe bereits automatisiert werden, wodurch sich daraus immer weniger Vorteile gegenüber den Konkurrenten erreichen lassen, die ebenfalls bestrebt sind, möglichst weite Teile ihrer internen Wertschöpfungskette durch den Einsatz moderner IT-Lösungen zu automatisieren. Der heutige Fokus liegt auf den Möglichkeiten zur schnelleren Anpassung interner Prozesse an externe Gegebenheiten und der Fähigkeit zur Auslagerung einzelner Geschäftsprozesse an Dritte, hauptsächlich zur Senkung von Kosten oder Durchlaufzeiten, aber auch der Qualitätsverbesserung [Oesterreich u. a., 2004, 4].

Die Voraussetzung zur Auslagerung von Geschäftsprozessen bildet jedoch deren möglichst vollständige Dokumentation und Beschreibung in einer Form, die zur Auslagerung geeignet ist und die mithilfe der Geschäftsprozessmodellierung realisiert werden kann. Hat ein Unternehmen seine Geschäftsprozesse modelliert, gewinnt es der Theorie zufolge an Flexibilität und kann schneller auf Veränderungen auf dem Markt reagieren, in dem Sinne, dass Veränderungen systematisch geplant und überwacht werden können.

Damit einher geht die Problematik schleichender Prozessveränderungen- und Erweiterungen, die oftmals nicht oder nur unzureichend dokumentiert werden und dadurch eine mögliche Auslagerung erschweren. Ein Ergebnis dessen sind IT-Systeme, die nur unzureichend an die Vorgänge auf fachlicher

Ebene angepasst und anpassbar sind. Die Auswirkungen von Veränderungen sind daher nur schwer erkenn- und nachvollziehbar.

Oesterreich u. a. betonen die massiven Probleme vieler Unternehmen, die aus willkürlichen und unkontrollierten Prozessveränderungen resultieren, bei denen keine konkreten Verantwortlichkeiten für die Prozessgestaltung, das heißt keine planende und leitende Instanz festgelegt sind und Prozesse ad-hoc an neue Bedingungen angepasst werden, ohne Seiteneffekte auf andere Prozesse zu bedenken. Einen weiteren wichtigen Aspekt bilden die durch Firmenfusionen und stärkere Vernetzung der Unternehmen angestrebten Synergieeffekte. Diese ergeben sich nicht durch die reine Übernahme oder Kopplung fremder IT-Systeme, sondern erfordern eine systematische Planung der neuen Prozesse, die nur auf Grundlage der Geschäftsprozessmodelle realisiert werden kann. Im Gegenteil kann die unternehmensübergreifende Geschäftsprozessmodellierung die Ausgangsbasis für Kooperationen oder Zusammenschlüsse bilden [Oesterreich u. a., 2004, 6].

Im Bereich der IT können vorhandene Geschäftsprozessmodelle als Grundlage der Anforderungsanalyse für die Softwareentwicklung eingesetzt werden und stellen damit einen wichtigen Beitrag zur „geschäftszielorientierten Anwendungsentwicklung“ dar. In umgekehrter Sicht kann auf Basis der Geschäftsprozessmodelle geprüft werden, inwiefern die vorhandene Software die Umsetzung eines neuen Prozesses unterstützt bzw. die Anpassung vorhandener Modelle möglich ist [Korherr, 2008, 3].

[Oesterreich u. a., 2004, 8] fassen die Vorteile der Geschäftsprozessmodellierung in ihrer Möglichkeit zusammen, Geschäftsprozesse *verstehen*, *dokumentieren*, *analysieren* und dadurch *verbessern* zu können. Dadurch wird der Ist-Soll-Vergleich zwischen momentanen und geplanten Prozessen vereinfacht und unterstützt [Staud, 2006, 17]. Von Hagen und Stucky heben die Möglichkeit der Geschäftsprozessmodellierung hervor, Organisations- und Medienbrüche zu beseitigen. Die Autoren verstehen Organisationsbrüche als Ablaufunterbrechungen- oder Verzögerungen infolge einer mangelhaften Organisationsstruktur und Medienbrüche ebenfalls als Unterbrechungen oder Verzögerungen, jedoch in Folge unterschiedlicher, eingesetzter Medien und deren Verarbeitung. Beide Aspekte, das heißt sowohl die Brüche auf fachlicher Ebene, als auch auf technischer Ebene, können durch die Prozessmodellierung erkannt und behoben werden [von Hagen und Stucky, 2004, 18f.].

Auf die Möglichkeit der Prozessmodellierung, Schwachstellen innerhalb der Prozessorganisation aufzudecken, weisen auch [Scheer und Jost, 1996, 42] hin, die gemäß den Schichten des ARIS-Konzepts die Organisationsprobleme auf funktionaler, organisatorischer, datenzentrierter und DV-zentrierter Ebene sehen. Durch die Reorganisation der Geschäftsprozesse im Anschluss oder während der Prozessmodellierung können Probleme und Ineffizienzen auf diesen Schichten entdeckt und behoben werden. Staud nennt eine ganze Reihe weiterer Schwachstellen auf technischer und fachlicher Ebene, die mithilfe der Geschäftsprozessmodellierung aufgedeckt werden können. Dazu gehört die fehlende oder mangelhafte Daten- und Prozessintegration, zu lange Transport-, Warte-, Liege-, und Bearbeitungszeiten von Geschäftsobjekten, redundante Tätigkeiten während der Prozessausführung, hohe Fehlerraten, hohe Prozesskomplexität, zu lange Kommunikations- und Entscheidungswege, zu hohe Kosten für den Prozessablauf, zu geringe Transparenz, sowie fragmentierte und unklare Verantwortlichkeiten bei der Prozessausführung [Staud, 2006, 18f.]. Diese Aufzählung ergänzen von Hagen und Stucky um weitere Aspekte, wie eine bessere Ressourcenzuteilung auf Basis der Prozesse, sowie die Eignung von Prozessmodellen als Grundlage zur Steuerung, Überwachung, Validierung, Verifizierung und Bewertung der Prozessleistung.

Rosenkranz betont die Möglichkeit, Computer zur Erfassung, Anpassung, Visualisierung, Berechnung, Simulation, Optimierung und Speicherung von Prozessmodellen einzusetzen. Dies ermöglicht die bessere Archivierung und Verteilung von Organisationswissen, wodurch das Controlling der Aufbau- und Ablauforganisation verbessert wird. Unter anderem ergibt sich die Möglichkeit der verursachungsgemäßen Ermittlung von Werten, Nutzen und Kosten, sofern diese Aspekte im Geschäftsprozessmodell berücksichtigt werden und erfassbar sind [Rosenkranz, 2006, 16f.].

Wie anhand der oben genannten Verwendungsmöglichkeiten deutlich wurde, kann eine Vielzahl verschiedener Ziele durch die Geschäftsprozessmodellierung verfolgt werden. Besonderes Augenmerk dieser Arbeit liegt jedoch auf der Verwendung von Geschäftsprozessmodellen als Grundlage zur Ges-

taltung von Services beziehungsweise serviceorientierten Architekturen. Darauf wird im Abschnitt 2.3 genauer eingegangen.

Knolmayer u. a. nennen zahlreiche weitere Anwendungsfelder auf Grundlage der Prozessmodellierung. Dazu gehört die geschäftsprozessorientierte Software-Entwicklung, Software-Auswahl, Anpassung von Standardsoftware an die Gegebenheiten der zu unterstützenden Geschäftsprozesse, Workflow-Spezifikation- und Modellierung, Simulation, Animation, Personaleinsatzplanung, Prozesskostenberechnung, Benchmarking, Zertifizierung, sowie die kontinuierliche Prozessverbesserung [Knolmayer u. a., 2000, 16]. Im folgenden Abschnitt werden die Bestandteile von Geschäftsprozessen genauer erläutert.

### 2.2.3 Begriffsbestimmungen im Rahmen der Prozessmodellierung

Aufgrund der verschiedenen Zielstellungen, die mit der Geschäftsprozessmodellierung verfolgt werden, ergeben sich unterschiedliche Beschreibungen und Definitionsversuche, was unter einem „Geschäftsprozess“ zu verstehen ist. Es erscheint sinnvoll, die Definition der Begriffe rund um den eigentlichen „Geschäftsprozess“ zunächst von oben herab zu betrachten und die Definitionen anschließend immer weiter zu verfeinern. Auf oberster Hierarchieebene befindet sich das fachliche Unternehmensmodell, das in Form einer Geschäftsarchitektur beschrieben werden kann.

*„Die **Geschäftsarchitektur** (business architecture) als System umfasst Geschäftsservices, Geschäftsprozesse, Geschäftsobjekte und die Organisation eines Unternehmens. Die Geschäftsarchitektur als Disziplin wird durch die Rolle des Geschäftsarchitekten abgedeckt. Er gestaltet die Geschäftsarchitektur anhand der Geschäftsstrategie und der Wertschöpfungskette. Er definiert unmittelbar die Geschäftsservices und Geschäftsprozesse des Unternehmens.“ [Engels u. a., 2008, 96]*

[Grässle u. a. 2000, S. 38] verwenden in diesem Zusammenhang den Begriff des „Geschäftssystems“ als Gesamtheit aller statischen und dynamischen Aspekte eines Unternehmens. Das Geschäftssystem umfasst im betriebswirtschaftlichen Sprachgebrauch die Wertschöpfungskette, welche die betriebliche Leistungserstellung beschreibt. Jeder Geschäftsprozess beschreibt damit die dynamischen Eigenschaften einer statischen Organisationsstruktur, in Form eines Geschäftssystems, beschrieben durch eine Geschäftsarchitektur. Auf Grundlage dieser Geschäftsarchitektur können Geschäftsprozesse entwickelt werden.

*„Ein **Geschäftsprozess** ist eine zielgerichtete, zeitlich-logische Abfolge von Aufgaben, die arbeitsteilig von mehreren Organisationen oder Organisationseinheiten unter Nutzung von Informations- und Kommunikationstechnologien ausgeführt werden. Er dient der Erstellung von Leistungen entsprechend den vorgegebenen, aus den Unternehmenszielen abgeleiteten, Prozesszielen. Ein Geschäftsprozess kann formal auf unterschiedlichen Detaillierungsebenen und aus mehreren Sichten beschrieben werden.“ [Gehring, 1998]*

Jeder Geschäftsprozess kann damit aus verschiedenen Sichten betrachtet werden, die in Form von Modellen beschrieben werden können.

*„Ein **Modell** ist ein Abbild oder eine Vereinfachung von existierenden oder gedachten Dingen, Personen, Abläufen oder Beziehungen der Realität.“ [von Hagen und Stucky, 2004, 59]*

Dem Geschäftsprozessmodell zugrunde liegt nach Bertram das Unternehmensmodell.

*„Ein **Unternehmensmodell** beschreibt die betriebliche Welt eines Unternehmens aus konzeptioneller Sicht und befasst sich mit den fachlichen Inhalten des Geschäftslebens, jedoch nicht mit der unterstützenden Technik. Es dient der gesamthaften, fachlichen Darstellung der Abläufe des Unternehmens, sowie deren Zuordnung zu den Organisationseinheiten.“ [Bertram, 1996, 83].*

Das Unternehmensmodell und die daraus ableitbaren Geschäftsprozessmodelle bilden dabei eine Hierarchie [Bertram, 1996, 81]. Das Ergebnis der Modellierung betrieblicher Abläufe und damit Formalisierung eines abstrakten, „gedanklichen“ Geschäftsprozesses ist ein Geschäftsprozessmodell.

„Ein **Geschäftsprozessmodell** (business process model) ist die Beschreibung und Darstellung aller relevanten Aspekte von Geschäftsprozessen in einer definierten Beschreibungssprache. Ziel und Ergebnis ist die modellhafte Nachbildung der Realität.“ [Engels u. a., 2008, 121]

„Ein **Prozessmodell** beschreibt die Struktur eines realen Prozesses. Es bestimmt alle möglichen Pfade entlang des Prozesses und bestimmt die Regeln für die Wahl der Pfade. Weiterhin bestimmt das Prozessmodell alle Aktivitäten, die ausgeführt werden müssen. Synonyme: Prozessdefinition, Prozessschema.“ [von Hagen und Stucky, 2004, 30]

Geschäftsprozessmodelle müssen der Definition nach nicht mithilfe formaler Beschreibungsmöglichkeiten erfasst werden, sondern können auch semiformal oder nichtformal beschrieben werden. Daher sind sie oftmals nicht direkt implementierbar [Galler, 1997, 8]. Die Implementierung erfordert eine Übersetzung oder Transformation der Geschäftsprozessmodelle in ein Workflow-Modell, um den gesamten Geschäftsprozess oder auch Teile daraus automatisieren zu können.

„Ein **Workflow** ist ein formal beschriebener, ganz oder teilweise automatisierter Geschäftsprozess. Er beinhaltet die zeitlichen, fachlichen und ressourcenbezogenen Spezifikationen, die für eine automatische Steuerung des Arbeitsablaufes auf der operativen Ebene erforderlich sind. Die hierbei anzustößenden Arbeitsschritte sind zur Ausführung durch Mitarbeiter oder Anwendungsprogramme vorgesehen. Vom Workflow als Typ oder Schema eines teil- oder automatisierten Arbeitsablaufes zu unterscheiden ist eine **Workflow-Instanz**, die eine konkrete Ausführung eines Workflow bezeichnet.“ [Gehring, 1998]

Nach [von Hagen und Stucky, 2004, 28] bildet ein Workflow damit einen zusammenhängenden, rechnergestützten Teil eines Geschäftsprozesses. Nach [Galler, 1997, 8] stellt ein Geschäftsprozess damit die organisatorisch-strategische und ein Workflow die informationstechnisch-operative Betrachtung der Ablauforganisation dar. Die konkrete technische Ausführung eines Workflows wird durch ein Workflow-Managementsystem realisiert:

“A **workflow management system** is a software system that supports the design, development, execution and analysis of workflow processes.” [Alonso u. a., 2004, 84]

“A **Workflow Management System** consists of software components to store and interpret process definitions, create and manage workflow instances as they are executed, and control their interaction with workflow participants and applications.” [WFMC, 1996]

Jeder Geschäftsprozess und damit jedes Geschäftsprozessmodell lässt sich in weitere Prozessbeschreibungselemente untergliedern. Folgende Elemente lassen sich in Anlehnung an [Bussler, 2003, 70ff.] unterscheiden.

Ein **Prozessschritt** repräsentiert die zur Erstellung der Prozessleistung erforderlichen Tätigkeiten. Synonyme: Aufgabe, Funktion, Vorgang, Arbeitsschritt, Aktivität, Task.

**Prozessobjekte** werden innerhalb der Prozessschritte bearbeitet oder zwischen Prozessschritten ausgetauscht. Sie bilden die Grundlage und den Bestandteil des Daten-, Informations-, Dokumenten-, oder Güterflusses und können Geschäftsobjekte darstellen.

**Prozessabhängigkeiten** sind zeitliche, logische oder technologische Verbindungen zwischen Prozessschritten. Ihre Gesamtheit bildet die Ablaufstruktur- und Logik des Prozesses. Synonyme: Steuerfluss, Kontrollfluss.

**Aufgabenträger** führen innerhalb eines Prozessschrittes eine konkrete Tätigkeit aus und wirken dabei auf Prozessobjekte ein. Beispiele: Abteilung, Rolle, Organisationseinheit, Anwendung.

Staud fasst die Gemeinsamkeiten aller Geschäftsprozessdefinitionen zusammen: Jeder Geschäftsprozess hat ein konkretes Ziel, das sich aus den Unternehmenszielen ableitet. Dieses Ziel bildet die Gesamtaufgabe des Geschäftsprozesses, die sich in Teilaufgaben zerlegen lässt. Jede Aufgabe wird durch einen Aufgabenträger (Mensch oder Maschine) übernommen. Jede Aufgabe kann dadurch manuell, teilautomatisiert oder vollautomatisiert ausgeführt werden. Geschäftsprozesse liegen in der Regel quer zur klassischen Aufbauorganisation und tangieren oft mehrere Abteilungen. Für die Erfüllung der Aufgaben werden Unternehmensressourcen benötigt, deren Verbrauch mithilfe des Prozessmodells

erfasst und der Prozess dadurch bewertet und optimiert werden kann [Staud, 2006, 7f.]. Masak ergänzt diese Beschreibung um folgende Eigenschaften: Jeder Geschäftsprozess erfordert einen eindeutigen Input und erzeugt einen konkreten Output und hat einen festen zeitlichen Anfang und Ende [Masak, 2007, 176]. Das Zusammenspiel der oben genannten Begriffe wird durch folgende Abbildung verdeutlicht.

Terminologieübersicht Geschäftsprozess- und Servicemodellierung

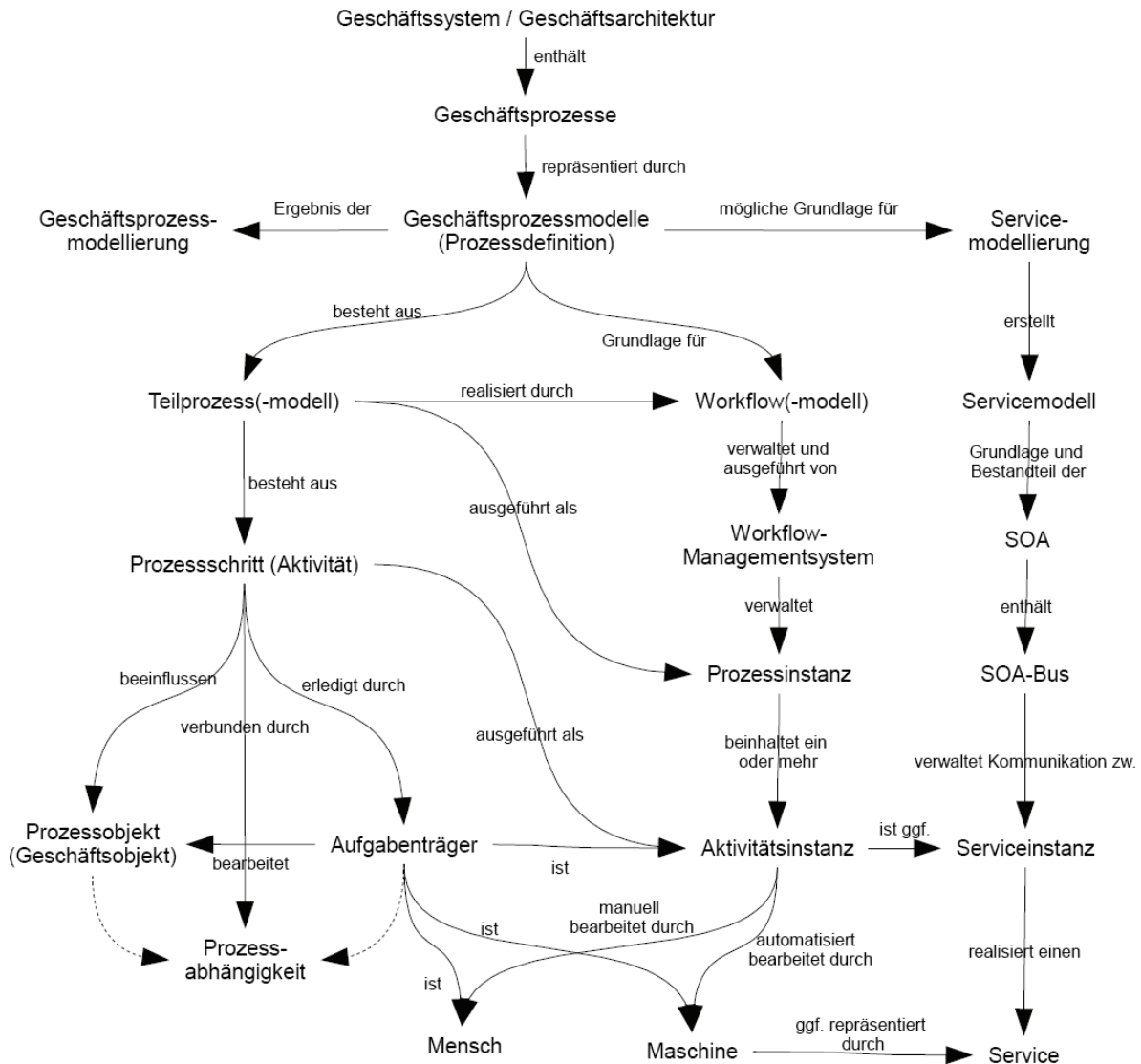


Abbildung 5: Terminologieübersicht Geschäftsprozess- und Servicemodellierung

## 2.2.4 Modellierungsmöglichkeiten auf Basis vorhandener Methoden und Sprachen

Innerhalb der letzten Jahrzehnte entstand eine Vielzahl verschiedener Ansätze und Methoden zur Beschreibung und Modellierung von Prozessketten. Nach [Grover und Markus, 2008, 45] gab es im Jahr 1997 bereits 25 verschiedene Methodologien, 72 Techniken und 102 verschiedene Werkzeuge zur Geschäftsprozessmodellierung. Mit der verstärkten Bearbeitung des Themas in den Unternehmen und der Forschung wuchs die Zahl der verschiedenen Geschäftsprozessmanagement-Werkzeuge bis zum Jahr 2005 auf über 350 an, die jeweils wieder unterschiedliche Sprachen, Notationen und Methoden unterstützen beziehungsweise der Modellierung zugrunde legen [Grover und Markus, 2008, 45].

Ein weiteres Problem bei der Auseinandersetzung mit diesem Thema bilden die ungenauen Begriffsbestimmungen. Zahlreiche Autoren unterscheiden nicht zwischen einer Methode und einer Sprache. Dadurch verschwimmt die Trennlinie zwischen einer Sprache zur Beschreibung von Prozessen und der Methode/Verfahrensweise zu ihrer Anwendung. Die „Sprache“ wird sozusagen zur „Methode“. Dieser Zusammenhang wird noch deutlicher, wenn betrachtet wird, dass eine Methode zur Erfassung eines Prozesses „in der realen Welt“ auf bestimmte Sprachkonstrukte angewiesen ist, um das Modellierungsziel zu erreichen.

Nach [Hoyer u. a., 2007, 185ff.] wird eine Geschäftsprozessmodellierungssprache verwendet, um existierende Prozesse abzubilden, oder neue Prozesse zu entwickeln. Dabei unterscheiden sich die Modellierungssprachen hinsichtlich ihrer Regeln in Bezug auf Syntax (Notation) und Semantik. Dabei kann grob zwischen textuellen und grafischen Sprachen unterschieden werden. Je nach Modellzweck- und Inhalt eignen sich die Sprachen unterschiedlich gut zur Abbildung der Komplexität der jeweiligen Situation. Für diese Arbeit erscheinen insbesondere die Sprachen interessant, die Konstrukte zur Verfügung stellen, um unternehmensübergreifende Prozesse abzubilden. Im 3. Kapitel wird darauf genauer eingegangen. Eine Auswahl der vorhandenen Methoden und Sprachen kann der folgenden Tabelle entnommen werden.

Datenflussorientiert	Objektorientiert	Kontrollflussorientiert	Semantisch
IDEF-Diagramme (Integration Definition for Function Modeling)	UML Aktivitätsdiagramm	Petri-Netze	Semantische EPK (sEPK)
Datenflussdiagramme gemäß der SSA	UML Use Case Diagramm	(Erweiterte) Ereignisgesteuerte Prozessketten (eEPK, EPK)	Web Service Modeling Ontology (WSMO)
Flussdiagramme gemäß SADT	Interaktionsdiagramme und Vorgangsereignisschemata gemäß SOM	Business Process Diagram (BPD) gemäß BPMN	Web Service Modeling Language (WSML)
	Objektorientierte Ereignisgesteuerte Prozessketten (oEPK)	Aufgabenkettendiagramm gemäß PROMET BPR	Business Process Management Ontology (BPMO)
	Zustandsübergangsdigramme (Statechart)	Struktogramme	Process Representation Language (PSL)
	Aktivitätsdiagramme (Activitychart)	Swimlane-Diagramme	Semantic Business Process Management (SBPM)
	Abstract Status Machine (ASM)	GPM-Diagramme (Ganzheitliche Prozessmodellierung)	OWL-S
		Herstellerspezifische Ansätze von GPO-Tools und Workflow-Managementsystemen	

Tabelle 4: Methoden und Sprachen der Prozessmodellierung  
in Anlehnung an [Yan u. a., 2007, 222] und [Gadatsch, 2005, 69ff.]

Nach [von Hagen und Stucky, 2004, 59] wird während der Modellierung das vorhandene Wissen geordnet, strukturiert, bewertet und anschließend innerhalb des Modells mit geeigneten Mitteln beschrieben. Je nach Modellzweck können verschiedene Modellarten unterschieden werden, beispielsweise Anforderungs-, Architektur-, Prozess-, Interaktions-, Entwurfs-, Daten-, Funktions-, Verhaltens- und Qualitätsmodelle. Von Hagen unterscheidet dabei drei zentrale Modellierungsfragen:

- Wodurch ist der abzubildende Realitätsausschnitt begrenzt und was beinhaltet er?
- Welche Konzepte und Beziehungen bestehen innerhalb des Realitätsausschnittes und durch welche Modellelemente können diese beschrieben werden?

- Welchen Detaillierungsgrad und Umfang, das heißt, welche Schichten und Ebenen werden unterschieden, auf denen sich die Modellelemente befinden?

Die im betrieblichen Umfeld vorhandenen Modellelemente sind: Aufgaben, Ablaufstrukturen (Sequenz, Alternative, Wiederholung, Verzweigung, Vereinigung), Ressourcen, Daten, Rollen, Organisationsstrukturen- und Einheiten, Zeitaspekte, Kostenaspekte, Ausnahme- und Fehlersituationen, Begriffe und Beziehungen, Geschäftsregeln<sup>1</sup>, Qualitätsanforderungen und Sicherheitsanforderungen [von Hagen und Stucky, 2004, 61].

Jede der oben genannten Prozessmodellierungssprachen unterstützt die Abbildung dieser Modellelemente mithilfe ihrer Sprachkonstrukte besser oder schlechter. Je nach Modellzweck ergeben sich an die Sprachen daher unterschiedliche Anforderungen. Dazu gehört beispielsweise die Ausdrucksmächtigkeit, Offenheit, Einfachheit, Verständlichkeit, Analysierbarkeit, Validierbarkeit, Ausführbarkeit, der Formalisierungs-, Präzisions- und Standardisierungsgrad, sowie die Herstellerunterstützung, aber auch Herstellerunabhängigkeit [von Hagen und Stucky, 2004, 63]. Nagl und Westfechtel ergänzen diese Anforderungen um die Sprachkonsistenz, sowie Steuerungs-, Kontroll- und Simulationsfähigkeit- bzw. Unterstützung dieser Sprache [Nagl und Westfechtel, 2003, 127].

Ziel einer Modellierungsmethode ist es demnach, die Anforderungen des Modellierungsziels mit den Fähigkeiten der verfügbaren Modellierungssprachen abzugleichen, die richtige Modellierungssprache auszuwählen und die Modellierung mithilfe dieser Sprache in der Art und Weise vorzunehmen, dass die Gegebenheiten des Realitätsausschnittes entsprechend den Bedingungen des Modellierungszwecks möglichst gut abgebildet sind. Für den Bereich der unternehmensübergreifenden Prozessmodellierung ergeben sich damit besondere Anforderungen an die zu verwendende Sprache. Auf diese Besonderheiten wird genauer eingegangen, wenn die unternehmensübergreifenden Prozesse näher erklärt wurden.

## 2.3 Zusammenhänge zwischen Prozess- und Servicemodellierung

### 2.3.1 Services als Bausteine von Prozessketten

In den vorangegangenen Abschnitten wurden die Möglichkeiten der Prozess- und Servicemodellierung in ihren Grundzügen besprochen. Einen wichtigen Bestandteil dieser Arbeit bildet der Vergleich beider Ansätze, hinsichtlich der Möglichkeiten, Prozesse durch die Verwendung von Services zu unterstützen oder in umgekehrter Sicht, Services in Prozesse einzubinden.

Bei Engels u. a. ergibt sich die Verbindung beider Ansätze bereits aus der Definition der elementare Prozessaktivität und des elementare Geschäftsservice:

*„Eine **elementare Prozessaktivität** (elementary process activity) ist eine Tätigkeit an einem oder mehreren Gegenständen oder Informationen, die auf die Erreichung eines bestimmten Ziels ausgerichtet ist. Diese beschreibt kontextfrei, wie Eingangsgrößen in Ausgangsgrößen umgewandelt werden. Sie ist die kleinste betriebliche Teilleistung, die ein Akteur in einer bestimmten Rolle immer vollständig oder gar nicht durchführt.“* [Engels u. a., 2008, 124]

*„Als **elementaren Geschäftsservice** (elementary business service) bezeichnet man einen Geschäftsservice mit folgenden Eigenschaften:*

1. *Er repräsentiert eine elementare Prozessaktivität.*
2. *Die Rolle des Akteurs, der die Aktivität ausführt, ist eindeutig bestimmt.*
3. *Das Geschäftsziel, das mit der Durchführung der Aktivität verbunden ist, ist eindeutig bestimmt.“* [Engels u. a., 2008, 125]

---

<sup>1</sup> Knolmayer, Gerhard; Endl, Rainer; Pfarher, Marcel: Modeling Process and Workflow by Business Rules, in [van der Aalst, 2000, S. 16ff.], Springer-Verlag, Berlin, Heidelberg, 2000.



Aus den beiden oberen Definitionen ergibt sich folgendes Bild: Jede elementare Prozessaktivität kann durch einen elementaren Geschäftsservice realisiert werden. Die Ein- und Ausgangsgrößen der Prozessaktivität dienen dabei als Ein- und Ausgabeparameter der Serviceschnittstelle. Der Service agiert in diesem Fall als Akteur, der für die Ausführung der Prozessaktivität eindeutig bestimmt ist. Das durch den Service erreichte Geschäftsziel entspricht der Funktionalität, die er kapselt und dem Prozess über seine Schnittstelle zur Verfügung stellt.

Wird der elementare Geschäftsservice durch eine Anwendungskomponente realisiert, bezeichnet dies einen Anwendungsservice [Engels u. a., 2008, 102].

*„Im Vergleich der beiden Begriffe Geschäftsservice und Geschäftsprozess zeigt sich, dass Geschäftsservices die Außensicht an der Schnittstelle zum Servicenehmer beschreiben und Geschäftsprozesse die Innensicht des Serviceerbringers.“* [Engels u. a., 2008, 97]

Gemäß der in Abschnitt 2.1.4 besprochenen Verteilung der vorhandenen Servicearten auf unterschiedliche Ebenen, lässt sich das Prinzip „Service realisiert Prozessaktivität“ auch auf Services mit höherem Abstraktionsgrad übertragen. Hierbei muss jedoch eine klare Trennung zwischen elementaren Geschäftsservices, die eine konkrete Aktivität realisieren, und Prozessservices, die einen bestimmten Abschnitt der Prozesskette kapseln, vorgenommen werden. Anders ausgedrückt kann sich ein Prozessservice einem oder mehreren anderen elementaren Geschäftsservices zur Erledigung seines Zieles direkt (ohne Zwischenservices) oder indirekt (mit Zwischenservices) bedienen und diese nacheinander oder parallel zueinander aufrufen. Dieses Konzept wird als Orchestrierung bezeichnet. Der Prozessservice kapselt in diesem Fall die Aufruffreihenfolge und -bedingungen entsprechend der in ihm festgelegten Prozesslogik. Gilt dieses Prinzip für Prozessservices, dann gilt es folglich auch für öffentliche Services, die insbesondere zum Zugriff auf die Systeme von Geschäftspartnern verwendet werden.

Vom Brocke hat sich intensiv mit dem Serviceorientierten Prozessmanagement (SOPM) beschäftigt. Dabei unterscheidet er zwischen der Strukturierung und der Institutionalisierung von Prozessen. Strukturiert wird der Prozess durch das Prozessmodell und institutionalisiert wird er durch die Subjekte, die den Prozess in der realen Welt ausführen [vom Brocke, 2008, 55]. Durch die Verwendung von Services und deren Unterstützung durch eine SOA, wird eine fachliche Kapselung der Aufgabenerfüllung vorgenommen, die nach prozessorientierter Sicht organisiert werden kann:

*„Ein **Service** bezeichnet eine Alternative zur Institutionalisierung einer Aktivität in einem Prozess.“* [vom Brocke, 2008, 55]

Der Service abstrahiert in diesem Fall von der Art der Institutionalisierung und damit auch der Art der Implementierung, denn diese wird durch die Serviceschnittstelle verborgen. Die Gesamtheit aller Services zur Unterstützung eines Prozesses wird als Service-Portfolio bezeichnet:

*„Ein **Service-Portfolio** bezeichnet die Menge an Services und Infrastrukturelementen, die zu einer Zeit für die Institutionalisierung eines Prozesses zu verwenden sind.“* [vom Brocke, 2008, 60]

Vom Brocke entwickelte in diesem Zusammenhang einen morphologischen Kasten zur Differenzierung von Services im Rahmen des Serviceorientierten Prozessmanagements, der folgender Tabelle entnommen werden kann.

Merkmal		Ausprägung		
Aktivität	Adressat	intern	in- und extern	extern
	Technologiebezug	unabhängig	wertabhängig	wesensabhängig
Infrastruktur	Durchführung	automatisiert	teilautomatisiert	nicht automatisiert
	Zeitbezug	ex-ante		ad-hoc
Service	Durchführung	automatisiert	teilautomatisiert	nicht automatisiert
	Organisationsbezug	intern	in- und extern	extern
	Struktur	elementar		komplex

Tabelle 5: Morphologischer Kasten zur Differenzierung von Services

[vom Brocke, 2007, 64]

### **2.3.2 Vor- und Nachteile bei der Verwendung von Services in Prozessketten**

Nachdem die Möglichkeit besprochen wurde, Services als Bausteine innerhalb von Prozessketten zu verwenden, wird im Folgenden die Frage geklärt, welche Vorteile, aber auch Probleme, daraus resultieren können. [Krcmar u. a., 2005a, 52] haben sich mit dem Konzept modularer Servicearchitekturen beschäftigt und sind der Frage nachgegangen, welche Vor- und Nachteile die Verwendung von Services bei der Erstellung komplexer Leistungen haben.

Die Möglichkeit der Wiederverwendung von Services in einem anderen Zusammenhang, als dem ursprünglich gedachten und geplanten, ist durch die offene Schnittstellenbeschreibung bereits implizit gegeben. Dadurch lässt sich ein fertiger Service nicht nur in einem, sondern theoretisch beliebig vielen Kontexten, in diesem Fall Geschäftsprozessen, verwenden und erhöht damit die Prozessflexibilität bei der Entwicklung neuer Prozesse. Damit einher geht die selektive Verwendung einzelner Services zur Erstellung kundenindividueller Servicekonfigurationen, um unterschiedlichen Kundenanforderungen gerecht werden zu können. Aufgrund der Abstraktionsfähigkeit einer Serviceschnittstelle gegenüber der Serviceimplementierung ist die individuelle Weiterentwicklung eines Service unabhängig von anderen Services oder der Prozesslogik möglich.

Mithilfe einer festgelegten Servicebeschreibungssprache können Services in der Regel besser standardisiert werden, als geschlossene Anwendungen, wodurch die Integrationskosten sinken. Indem für jeden Service separate Bewertungsfaktoren erfasst werden können, sind kontinuierliche Verbesserungen der Services im Hinblick auf Qualität, Zeit und Kosten möglich. Zu diesen Bewertungsfaktoren gehört beispielsweise die Nutzungshäufigkeit (Anzahl der Schnittstellenaufrufe pro Minute, Stunde oder Tag) auf deren Basis die Last- und Ressourcenverteilung für jeden Service einzeln gesteuert werden kann. Aus diesen technischen Vorteilen ergeben sich weitere betriebswirtschaftliche Vorteile durch die Parallelisierung von Entwicklungsaktivitäten bei der Erstellung neuer Prozesse, der beschleunigten Markteinführung, dem Angebot einer hohen Zahl von Produkt und der nachfragespezifischen Konfiguration der angebotenen Dienstleistungen [Krcmar u. a., 2005a, 53].

Nachteile und Risiken sehen Krcmar u. a. darin, dass durch die Standardisierung der Services auch eine Standardisierung der Geschäftsprozesse einher geht, wodurch der Kundennutzen sinken kann, da Konkurrenten die angebotenen Services gegebenenfalls schnell imitieren und möglicherweise spezielle Leistungen günstiger anbieten können [Krcmar u. a., 2005a, 54].

### **2.3.3 Probleme auf dem Weg vom Geschäftsprozess- zum Servicemodell**

Thalbauer u. a. haben eine Integrationsplattform zur Verknüpfung von Geschäftsprozessen mit IT-Services entworfen, bei der sie auf die Grenzen der Transformation von Geschäftsprozessmodellen in ausführbare Software bzw. Services aufmerksam machen. Sowohl der Bereich der Geschäftsprozessmodellierung, als auch der Softwareentwicklung sind jeweils stark ausgereift und unternehmensintern weitgehend erforscht [Thalbauer u. a., 2006, 63f.]. Jedoch sind beide Ansätze bisher unzureichend integriert, weshalb aus beiden Richtungen eine Integration der Gegenseite versucht wird. Thalbauer u. a. unterscheiden dabei Werkzeuge aus dem Bereich der Softwareentwicklung, mit deren Hilfe versucht wird Geschäftsprozessmodelle zu integrieren und Werkzeuge aus dem Bereich des Geschäftsprozessmanagements, bei denen die Modellierung softwaretechnisch unterstützt werden soll.

Im ersten Fall unterstützen die Softwareentwicklungswerkzeuge bisher kaum die Analyse, Bewertung und Simulation von Geschäftsprozessen, im zweiten Fall fehlt den Geschäftsprozessmanagementwerkzeugen oft die Möglichkeit, die Prozesslogik, ohne erhebliche manuelle Eingriffe, in Programmcode oder Serviceschnittstellenbeschreibungen zu überführen. Dies stellt einen interessanten Forschungsansatz im Bereich der Wirtschaftsinformatik dar.

Wie im Kapitel 2.2 bereits besprochen wurde, können Geschäftsprozessmodelle die Grundlage zur Softwareentwicklung-, Auswahl- und Integration innerhalb des Unternehmens bilden. Dabei lassen sich die fachlichen Anforderungen innerhalb der Geschäftsprozessmodelle nur schwer direkt in Software umwandeln, jedoch kann ein Teil der bereits erfassten Informationen mithilfe geeigneter Techni-

ken übertragen werden. Bisher erfolgt die Übertragung fachlicher Informationen auf Prozessebene oftmals durch eine textuelle Beschreibung im Lastenheft. Dieses wird dadurch zum „Flaschenhals“, bei dem viele Informationen verloren gehen, die bereits modelliert wurden. Es liegt damit häufig ein Medienbruch innerhalb der Transformationskette von den Geschäftsanforderungen zur Umsetzung in der IT vor. Daraus ergibt sich auch der bisher erhebliche Anpassungsaufwand bei Änderungen vorhandener Geschäftsprozesse, weil dieser Bruch auch anschließend noch vorhanden ist. [Thalbauer u. a., 2006, 67f.] führen dies auf vier zentrale Probleme zurück:

Durch die *Mehrsprachigkeit* innerhalb des Unternehmens modellieren die Mitarbeiter der Fachabteilungen ihre Anforderungen und Geschäftsprozesse meistens in deutscher Sprache, während die Implementierung der Software und Services in der Regel in englischer Sprache geschieht. Damit besteht von vorn herein ein Bruch der Spezifikationen, wodurch sich die Modelle nicht ohne erheblichem Übersetzungsaufwand und den damit verbundenen Probleme übertragen lassen. Das zweite Problem bilden die *unterschiedlichen Beschreibungsmethoden*. Während die Fachabteilungen ihr Wissen oft mithilfe von textuellen oder semi- bis nicht-formalen Beschreibungsmethoden erfassen, ist die Softwareentwicklung auf formale Modelle angewiesen. Hier sind entsprechende Schulungen der Fachabteilungen notwendig und sinnvoll, auch im Hinblick auf die Archivierung, Versionierung und Wiederverwendung der erstellten Modelle. [Dustdar u. a., 2007, 105] sprechen in diesem Zusammenhang vom Problem der unterschiedlich hohen Abstraktionsebene zwischen den Modellen auf fachlicher und technischer Ebene. Ein weiteres Problem besteht im *Fehlen eines einheitlichen Regelwerks*, in dem die Modellierungskonventionen- und Richtlinien verbindlich für alle Beteiligten festgelegt sind. Hierbei sei auf die Rolle der Modellierungsorganisation hingewiesen, wie sie im Abschnitt 2.2.1 beschrieben wurde.

Das vierte Problem besteht in der Notwendigkeit einer *koordinierenden Integrationsplattform*, die auf der einen Seite das gesamte Geschäftsprozessmanagement von der Modellierung, der Ausführung bis zur Optimierung unterstützt, auf der anderen Seite aber auch Generatoren für die Softwareentwicklung anbietet, mit deren Hilfe auf Basis der fachlichen Modelle Implementierungsvorlagen und Servicemodelle erzeugt werden können, beispielsweise mit fertigen Schnittstellenbeschreibungen. Aufgrund der sehr hohen Komplexität, die eine solche Integrationsplattform verwalten müsste, schlagen [Dustdar u. a., 2007, 105f.] die Unterteilung der prozessgetriebenen SOA in fünf verschiedene, plattformneutrale und standardisierte Sichten vor, deren Zusammenspiel die Generierung ausführbaren Codes, beispielsweise in BPEL, WSDL oder Java, ermöglicht. Dazu gehört die Prozess-, Kollaborations-, Informations-, Orchestrierungs- und Servicesicht, die im Folgenden jedoch nicht genauer erklärt werden und auf die Literatur verwiesen wird.

Einen weiteren Ansatz in dieser Richtung beschreiben [Gruber und Bzozowski, 2007, 54f.] mit dem Executable Business Process Model (EBPM). Grundlage dieses ausführbaren Geschäftsprozessmodells bildet die Business Modeling Architecture (BMA), als Bestandteil der IT-Systemarchitektur. Nach diesem Ansatz kann die semi-automatische Anbindung der IT-Technologien an eine ausführbare Systemarchitektur durch die Implementierung der Prozesse nach dem serviceorientierten Ansatz vorgenommen werden. Gruber und Bzozowski schlagen in diesem Fall einen Top-Down-Entwicklungsansatz vor, bei dem die Top-Level-Geschäftsprozesse in mehreren Schritten in elementare Prozessschritte untergliedert werden, deren Ausführung in jedem Fall durch einen Service geschieht.

In Abhängigkeit davon, ob es sich um einen automatisierbaren Service handelt, liegt dieser gegebenenfalls auch in Form einer Eingabe- oder Auswahlmaske vorliegt, die durch einen Menschen bedient wird. Sämtliche Services werden zunächst in einem hierarchischen Geschäftsprozessdiagramm und anschließend in einem Geschäftsprozessflussdiagramm erfasst. Eine konkrete Sprache ist für beide Diagramme nicht explizit vorgegeben.

Das „Herunterbrechen“ aller Prozessbestandteile auf den kleinsten gemeinsamen Nenner „Service“ bildet in diesem Fall die Grundlage zur vollständigen Abbildung der fachlichen Geschäftsprozessmodelle auf eine ausführbare Servicearchitektur. Die Überwachung der Ausführung von EBPM kann mithilfe eines Business Activity Monitors (BAM) geschehen, der das Nachrichten- und Datenmonitoring, die Überwachung von Metriken und KPI's, die visuelle Darstellung des Prozessablaufs- und fortschritts, sowie manuelle oder automatische Eingriffe bei Fehlern erlaubt. Damit dient

das EBPM als Brücke zwischen der Geschäftsprozessmodellierung und der Ausführungsplattform [Gruber und Bzozowski, 2007, 61].

## **2.4 Zusammenfassung**

Für die Integration von Services in Geschäftsprozesse ergeben sich mehrere grundlegende Voraussetzungen, die im Unternehmen geschaffen werden sollten. Auf der einen Seite muss das Unternehmen mit dem Geschäftsprozessmanagement vertraut sein, welches die Grundlage zur Geschäftsprozessmodellierung bildet. Im Rahmen der Geschäftsprozessmodellierung können konkrete Modellierungsrichtlinien- und Konventionen die Voraussetzungen zur Übertragung der Geschäftsprozessmodelle auf ausführbare Servicemodelle schaffen, worauf im vorherigen Abschnitt eingegangen wurde. Auf der anderen Seite muss das Unternehmen mit den Prinzipien der Servicemodellierung und damit der Entwicklung und dem Betrieb einer SOA vertraut sein, um die entstehende Komplexität der „Service-Welt“ beherrschbar zu machen. Beide Entwicklungsrichtungen erfordern entsprechendes Know-how, sowohl bei den Mitarbeitern in den Fachabteilungen, als auch den Software- und Serviceentwicklern.

Aus der Kombination der Geschäftsprozessmodellierung und der Servicemodellierung ergibt sich eine Reihe von Vorteilen für das Unternehmen. Der Kerngedanke der Geschäftsprozessmodellierung, Prozesse besser Erfassen, Anpassen, Visualisieren, Berechnen, Simulieren und Optimieren zu können wird durch den Einsatz einer SOA unterstützt. Indem ein Teil oder auch alle elementaren Prozessaktivitäten durch Services realisiert werden, erfolgt deren Erfassung innerhalb des Serviceverzeichnis, ihre Anpassung durch Service- und Prozessentwickler gleichermaßen und ihre Visualisierung, Simulation und Optimierung parallel oder gemeinsam auf fachlicher Prozessebene und technischer Serviceebene.

Das Prinzip der Orchestrierung kann zur Entwicklung von Prozessservices eingesetzt und aus den vorhandenen Geschäftsprozessmodellen übernommen werden. Dieser enge Zusammenhang zwischen den verfügbaren Servicekompositions- bzw. Ausführungssprachen wird anhand der großen Menge gemeinsamer Prozessabhängigkeiten (Und, Oder, Verzweigung, Join, ...) zur Modellierung der Prozesslogik deutlich, die sowohl für viele Geschäftsprozessmodellierungssprachen, als auch mehrere Serviceausführungssprachen, zum Beispiel BPEL oder WSCL, verfügbar sind. Ein weiterer, wichtiger Vorteil aus der Kombination beider Ansätze bildet die Möglichkeit, Geschäftsprozesse im Rahmen des Business Process Outsourcing (BPO) aus dem Unternehmen auszulagern, oder die Anbindung von Partnern zu verbessern, indem unternehmensübergreifende Geschäftsprozesse entwickelt werden, bei denen Services ebenfalls zum Einsatz kommen können. Darauf wird im nachfolgenden Kapitel genauer eingegangen. Es bleibt festzustellen, dass sich die beiden Ansätze der Geschäftsprozess- und Servicemodellierung gegenseitig befördern und zur Erreichung verschiedener Vorteile und Ziele miteinander kombiniert werden können, sofern innerhalb des Unternehmens die notwendigen Grundlagen geschaffen wurden.

## **3 Entwicklung organisationsübergreifender Prozesse und Services**

### **3.1 Motivation und Zweck unternehmensübergreifender Prozesse**

Geschäftsprozesse dienen ihrer Definition zufolge zur Erstellung einer Leistung für einen oder mehrere Kunden. Zu diesem Zweck können unternehmensinterne- oder externe Ressourcen, wie Menschen, Materialien und IT-Systeme zum Einsatz kommen. Durch die Verflechtung der unternehmensübergreifenden Leistungserstellung zeigt sich, dass Geschäftsprozesse nicht mehr starr an der Unternehmensgrenze enden sollten, sondern Kunden, Partnern und Lieferanten dynamisch miteinander verbinden. Daraus entsteht nicht nur die Möglichkeit, sondern auch die Notwendigkeit, die Optimierung der Wertschöpfungskette nicht mehr nur intern, sondern unternehmensübergreifend vorzunehmen. Zu diesem Zweck ist der Austausch von Daten und Koordinationsinformationen notwendig. Häufig werden diese Informations- und Kontrollflüsse an der Unternehmensgrenze unterbrochen oder ihre Über-

tragung gehemmt, die „*unternehmensinterne Datenautobahn*“ wird zum „*unternehmensexternen Feldweg*“ [Staud, 2006, 16f.].

Nach [Cannon, 1993] ergab eine Studie bereits im Jahr 1993, dass 70 % der manuell erfassten Informationen bei anderen Unternehmen bereits in elektronischer Form vorliegen, deren erneute Erfassung erhebliche Kosten verursacht, die durch ihren elektronischen Austausch gesenkt werden könnten. Es ist davon auszugehen, dass der Anteil manuell erfasster Informationen seit dem Jahr 1993 durch die stärkere Vernetzung der Unternehmen, beispielsweise über das Internet, abgenommen hat, jedoch besteht nach wie vor eine Diskrepanz zwischen bereits gespeicherten und an anderer Stelle benötigten Informationen, welche durch die Entwicklung öffentlicher Prozesse gemindert werden kann. Für die Entwicklung unternehmensübergreifender Prozesse ist eine Reihe von Treibern erkennbar. [Dressler, 2007, 20f.] nennt vier wesentliche *Treiber* aus technischer Sicht:

Durch den weit verbreiteten *Einsatz von Personalcomputern*, insbesondere an den Arbeitsplätzen der Mitarbeiter, ergibt sich ein hoher Digitalisierungsgrad unternehmensinterner Informationen, beispielsweise in Bezug auf Buchhaltungs-, Mitarbeiter-, Finanz-, und Kundendaten. Je mehr Informationen in digitaler Form vorhanden sind, je einfacher ist es, diese Informationen anderen Unternehmen zur Verfügung zu stellen, im Vergleich zur „Speicherung“ und Übertragung in Papierform. Das Internet und die damit verbundenen *Internettechnologien* bilden den zweiten wesentlichen Treiber, durch den der schnelle Austausch großer Datenmengen realisiert werden kann.

Nach [von Hagen und Stucky, 2004, 27] werden zur Entwicklung von öffentlichen Services und Prozessen insbesondere internetbasierte Technologien eingesetzt und ältere Technologien, wie zum Beispiel CORBA, zunehmend zurückgedrängt. Ein weiteres Standbein bildet die weite *Verbreitung von ERP-Systemen*, wodurch die Vereinheitlichung und Integration verschiedener Datenstrukturen und damit deren Übertragung zu anderen Unternehmen vereinfacht wird. Als vierten Treiber zählt Dressler die *modernen Kommunikationstechniken* in Bezug auf Mobiltelefone, Handhelds, Notebooks mit UMTS und WLAN, mit deren Hilfe der Zugriff auf die Informationssysteme des eigenen, oder auch anderer Unternehmen, vereinfacht wird [Dressler, 2007, 22f.]. Ziel des Einsatzes von Shared Services ist es, Unterstützungsdienstleistungen im Hinblick auf Kosten, Qualität, Durchlauf- und Prozesszeiten als wettbewerbliche Alternative zum Outsourcing anzubieten [Dressler, 2007, 25].

Als mögliche Ausprägung unternehmensübergreifender Zusammenarbeit nennt Dressler das Business Process Outsourcing. Dem BPO liegt die zentrale „Make-or-Buy“-Entscheidung zugrunde, ob eine konkrete unternehmerische Leistung selbst erbracht oder fremd bezogen wird. Dieser Grundgedanke leitet sich aus der Entwicklung der Automobilbranche ab, in der die Fertigungstiefe in den letzten Jahrzehnten immer weiter abnahm und die Auslagerung von Wertschöpfungsaktivitäten immer wichtiger wurde, um Kosten zu senken. Für das Outsourcing im Dienstleistungssektor hält Dressler insbesondere die Verwaltungsfunktionen, Lohn- und Gehaltsbuchhaltung, die Personalverwaltung, die Beschaffung indirekter Materialien und die Vertriebsunterstützung für gut geeignet [Dressler, 2007, 59]. Als wesentliche betriebswirtschaftliche Treiber zur Entwicklung öffentlicher Prozesse sieht Dressler die möglichen Kosteneinsparungen für das eigene Unternehmen, die Leistungsverbesserung durch Erhöhung des Service-Levels und Steigerung der Qualität der Leistungserstellung, die Prozessstandardisierung und -verbesserung, sowie die Erhöhung der Kundenzufriedenheit und damit der eigenen Wettbewerbsfähigkeit [Dressler, 2007, 38f.].

Scheer vertieft diese Betrachtung und nennt weitere Rahmenbedingungen als Grundlage der Entwicklung öffentlicher Prozesse (folgendes nach [Scheer u. a., 2002, 9f.]). Durch die Verbreitung des Internets wird der Aufwand zur *Globalisierung* für kleine und mittelständige Unternehmen gesenkt, weil niedrigere Kosten für die Ausbreitung auf externe Märkte anfallen. Im Rahmen der *Kundenorientierung* werden die Kunden zunehmend in die Gestaltung der Geschäftsprozesse eingebunden, um ihnen individuelle Waren und Dienstleistungen anbieten zu können. Dieser Vorgang erfordert die Integration des Kundenwissens in die Prozessgestaltung und den Kunden als aktiven Teil der Prozesskette im Rahmen des End-to-End-Business. Einen weiteren wichtigen Punkt bildet die Zunahme des Vernetzungsgrads zwischen Unternehmen durch die Entstehung von *kollaborativen Unternehmensnetzwerken*, wie den Business Webs. Dies wird unter anderem bedingt durch die Änderung des Wertes von *Informationen*, die nicht mehr nur als unterstützendes Element des Waren- und Dienstleistungs-

austauschs betrachtet werden, sondern als wertschaffender Faktor, der insbesondere im unternehmensübergreifenden Bereich für mehrere Partner einen Nutzen generieren kann. Einen anderen Ansatzpunkt bildet die größere Anzahl automatisierter Prozessschritte, bei denen die Automatisierung erst das Potenzial zu ihrer Veröffentlichung erschließt oder diese vereinfacht [von Hagen und Stucky, 2004, 27].

Abecker u. a. sehen die Vorteile der Entwicklung öffentlicher Prozesse in der Möglichkeit zur externen Nutzung unternehmensinternen Wissens. Geschäftsprozesse, insbesondere die Kernprozesse, bilden die Kernkompetenzen eines Unternehmens und stellen damit eine wichtige Wissensbasis dar. Geschäftsprozesse benötigen und generieren während ihres Ablaufes Wissen. Das Wissen über den effizienten Ablauf von Prozessen kann ein Unternehmen noch leistungsfähiger machen, wenn es dieses nicht nur zur internen Prozessoptimierung verwendet, sondern zusätzlich in Form öffentlicher Prozesse für Partner oder Kunden zur Verfügung stellt [Abecker u. a., 2002, 3f.]. Beispielsweise bietet die Deutsche Post AG ihren Kunden und Geschäftspartnern die Möglichkeit, Sendungen mithilfe eines Online-Formulars verfolgen zu lassen, um den Versandstatus und aktuellen Standort zu überprüfen. In diesem Falle wurde ehemals unternehmensinternes Wissen über die Schnittstelle eines öffentlichen Services für externe Verwender zur Verfügung gestellt, um damit zusätzlichen Nutzen für die Kunden zu erzeugen und die Kundenorientierung zu verbessern [DPAG, 2008].

Alonso u. a. verfolgen eine sehr technische Perspektive und sehen die Vorteile organisationsübergreifender Prozesse in der Integration autonomer, heterogener Softwaresysteme verschiedener Partner zur Automation der Geschäftsprozesse über diesen Systemen. Enterprise Application Integration (EAI) Systeme unterstützen zwar die Integration interner Geschäftsprozesse, scheitern jedoch am medialen Bruch der Unternehmensgrenze, das heißt der Übertragung von Informationen oder Daten in das System eines Partners. Keine der bisherigen EAI Systeme unterstützt die Lösung dieses Problems bisher ausreichend gut [Alonso u. a., 2004, 125ff.]. Im nächsten Kapitel werden öffentliche Prozesse und Services als potenzielle Lösung dieses Problems genauer beschrieben.

### 3.2 Grundlagen öffentlicher Prozesse

Zur Definition öffentlicher Prozesse ist zunächst deren klare Abgrenzung zu den im Unternehmen ablaufenden, nicht-öffentlichen Prozessen vorzunehmen. Diese Abgrenzung kann erst geschehen, wenn die unternehmensinternen Prozesse charakterisiert und daraufhin untersucht wurden, ob sie zur Veröffentlichung geeignet sind. Von Hagen und Stucky bieten zu diesem Zweck mehrere Klassifikationskriterien an, auf deren Grundlage verschiedene Prozessarten unterschieden werden können.

Klassifikationskriterium	Ausprägungen		
	Strukturiertheit	vollständig strukturiert	semi- bzw. teil-strukturiert
Art des Auftretens	regelmäßig	teilweise regelmäßig	unregelmäßig
Häufigkeit des Auftretens	sehr häufig	häufig	selten
Prozessdauer	lang (mehrere Stunden bis Tage)	mittel (mehrere Minuten bis Stunden)	kurz (mehrere Sekunden bis Minuten)
Automatisierungsgrad	vollständig automatisiert	teilweise automatisiert	nicht automatisiert
Verwendung	öffentlich	öffentlich und nicht-öffentlich	nicht öffentlich

Tabelle 6: Morphologischer Kasten zur Klassifizierung von Prozessarten

in Anlehnung an die textuelle Beschreibung von [Von Hagen und Stucky, 2004, 21ff.]

Gemäß der Art des Auftretens unterscheiden [Picot und Rohrbach, 1995, 25f.] Routineprozesse, Regelprozesse und einmalige Prozesse. *Routineprozesse* sind klar strukturiert und auf lange Sicht weitgehend planbar. Dadurch können sie sehr gut automatisiert werden. *Regelprozesse* sind nicht streng

determiniert, aber der Prozessablauf ist im Wesentlichen bestimmbar und kann durch den Eingriff von Mitarbeitern verändert werden. Sie beinhalten eine noch kontrollierbare Struktur und Komplexität. *Einmalige Prozesse* haben keinen festgelegten Prozessablauf und werden von einzelnen Teams oder Mitarbeitern individuell bearbeitet. Es zeigt sich an dieser Stelle, dass ein Prozess zur Automatisierung dann am besten geeignet ist, wenn er möglichst vollständig strukturiert ist und regelmäßig abläuft. Daraus ergeben sich auch direkte Vorteile für seine Veröffentlichung.

Abecker u. a. unterscheiden zwei unternehmensinterne Prozessarten, die Geschäftsprozesse und die ihnen zugehörige sekundäre Prozesse. Die Geschäftsprozesse liefern den wesentlichen Beitrag zur Wertschöpfung im Unternehmen, haben einen externen Auslöser und liefern diesem ein konkretes Ergebnis nach außen. Sekundäre Prozesse stellen Abfolgen von Aktivitäten dar, welche die Ausführung von Geschäftsprozessen unterstützen, indem sie die Ressourcen zur Bearbeitung von Geschäftsprozessen zur Verfügung stellen [Abecker u. a., 2002, 164f.]. Die Autoren Melano und Pidd bezeichnen die Geschäftsprozesse als Kernprozesse und sekundäre Prozesse als Unterstützungsprozesse. Weiterhin unterscheiden sie Managementprozesse zum Management der Kern- und Unterstützungsprozesse, sowie interorganisatorische Prozesse, die mindestens zwei rechtlich selbstständige Unternehmen tangieren [Melano und Pidd, 2008, 45].

Meyer u. a. unterscheiden Management-, Produktions-, und Unterstützungsprozesse [Meyer u. a., 2005, 43]. Engels u. a. unterscheiden Kerngeschäftsprozesse, unterstützende Geschäftsprozesse (Unterstützungsprozesse), strategische Geschäftsprozesse (Managementprozesse) und innovative Geschäftsprozesse zur Erarbeitung und Erprobung von Innovationen [Engels u. a., 2008, 74ff.]. Auf Grundlage der oben beschriebenen Prozessarten und deren Eigenschaften können folgende Begriffsdefinitionen vorgenommen werden:

*„Ein **organisationsübergreifender Prozess** besteht aus einer Anzahl organisationsinterner, privater Prozesse sowie aus zwischen ihnen liegenden öffentlichen Prozessen, so genannte Geschäfts- oder B2B-Protokolle, welche die privaten Prozesse miteinander verbinden.“* [Masak, 2007, 212]

*„Ein **öffentlicher (Geschäfts-)Prozess** ist Teil eines oder mehrerer interner Geschäftsprozesse, der zum Zweck der überbetrieblichen Zusammenarbeit in einen organisationsübergreifenden Prozess integriert ist. Er wird über öffentliche Services bereitgestellt.“* [eigene Definition]

In diesem Sinne bilden interne Geschäftsprozesse die Grundlage öffentlicher Geschäftsprozesse und diese wiederum die Grundlage organisationsübergreifender Prozesse. Managementprozesse dienen der Planung, Steuerung und Optimierung interner und damit auch öffentlicher Prozesse. Unterstützungsprozesse dienen ihrerseits zur Unterstützung mit den benötigten Ressourcen. Das Zusammenspiel der verschiedenen Prozessarten wird durch die folgende Abbildung verdeutlicht.

### Organisationsübergreifende Prozesse

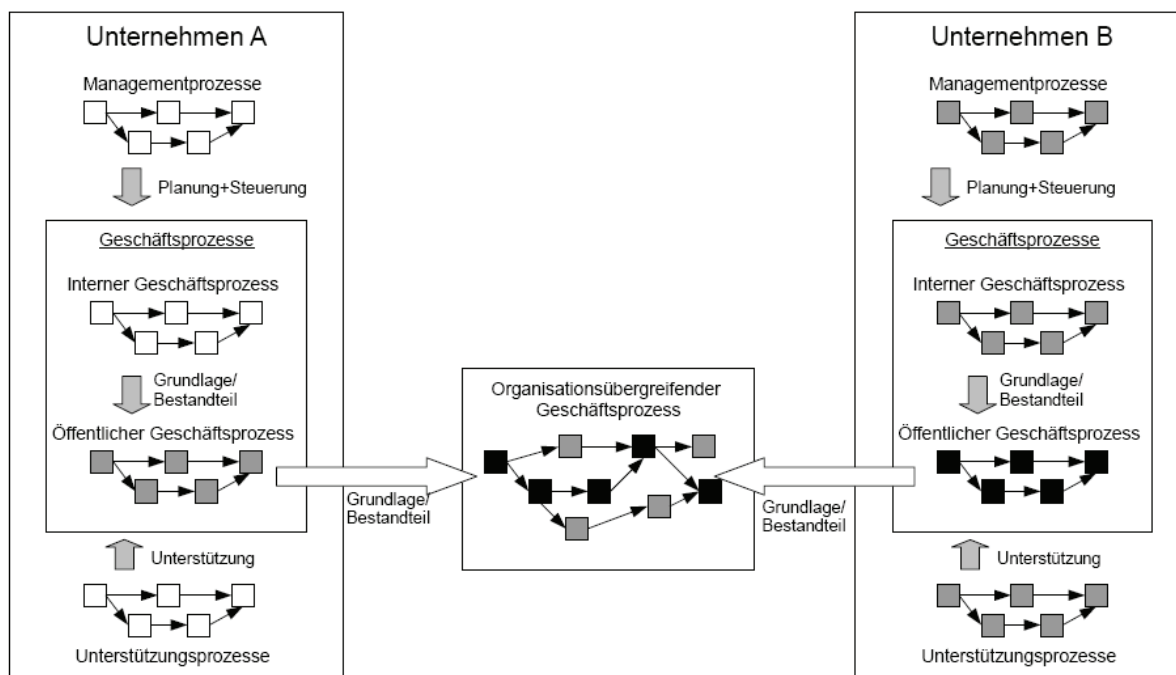


Abbildung 6: Schemata organisationsübergreifender Prozesse

In der Literatur finden sich noch andere Begriffe, mit denen organisationsübergreifende Geschäftsprozesse beschreiben werden. [Greiner, 2006, 5f.] bezeichnet sie als „kooperative Prozesse“ („cooperative processes“), [Andonoff u. a., 2005, 120] als „inter-organizational workflows“ (IOW), [Hoyer u. a., 2007, 185ff.] als „collaborative e-Business processes“ (CBP) oder „interorganisatorischen Prozessen“ und [Ziemann u. a., 2007, 87] als „cross-organizational business processes“ (auch CBP). Aus der Trennung interner und externer Geschäftsprozesse ergeben sich nach [Masak, 2007, 214f.] zahlreiche Vorteile. Die lokalen Prozessteile können in beliebigen, heterogenen Umgebungen vorliegen und unabhängig von den Prozessen der Partner und dem Gesamtprozess jederzeit lokal geändert werden, solange die Schnittstellen zu den öffentlichen Services unverändert bleiben. Der Zugriff auf öffentliche Prozesse wird über öffentliche Services bereitgestellt, worauf im nachfolgenden Abschnitt genauer eingegangen wird.

Ein weiterer Vorteil besteht in der nahezu beliebigen Kombination bzw. Komposition lokaler, öffentlicher Prozesse einzelner Partner zu globalen Prozessen über mehrere Partner, ohne die Autonomie der einzelnen Prozessteilnehmer zu beeinträchtigen. Jeder Prozessteilnehmer nimmt damit eine bestimmte Rolle innerhalb des globalen, organisationsübergreifenden Prozesses ein und kann gegebenenfalls durch einen anderen Teilnehmer ersetzt werden, wenn dies notwendig wird. Dadurch wird die Modellierung flexibler Geschäftsbeziehungen ermöglicht, bei denen die Partner wechseln können. Auf die Probleme und Hindernisse, die sich bei der Umsetzung organisationsübergreifender Geschäftsprozesse ergeben, wird im Abschnitt 3.3.1 genauer eingegangen.

Mit Bezug auf die Prozessklassifikation am Anfang dieses Abschnittes ergeben sich für organisationsübergreifende Geschäftsprozesse mehrere Eigenschaften beziehungsweise Eignungskriterien. Gut geeignet für die überbetriebliche Zusammenarbeit sind Prozesse, die möglichst vollständig in Form von Geschäftsprozessmodellen strukturiert und formalisiert sind. Dies setzt ein gutes Verständnis der eigenen, internen Geschäftsprozesse voraus und bildet aus fachlicher und technischer Sicht die Grundlage für deren partielle Veröffentlichung. Ist ein Geschäftsprozess gut formalisiert, werden dessen Automatisierung und damit die Einbindung in globale Prozessketten erleichtert. In Bezug auf die Art und Häufigkeit des Auftretens erscheint es nicht sinnvoll, seltene und unregelmäßige Geschäftsprozesse öffentlich abzubilden, das Gegenteil gilt entsprechend für häufige und regelmäßige Geschäftsprozesse. Die Prozessdauer kann bei öffentlichen Prozessen bis zu mehreren Tagen dauern, woraus sich besondere Anforderungen an das Servicedesign und die Prozessausführungsumgebungen ergeben,



worauf im nächsten Abschnitt genauer eingegangen wird. Nagl und Westfechtel ergänzen diese Eigenschaften und nennen weiterhin einen niedrigen Anteil unscharfer, nichtformaler Informationen zur Prozesskontrolle und eine geringe Entscheidungsintensität als Grad der Beeinflussung des Prozessablaufs durch menschliche Eingriffe [Nagl und Westfechtel, 2003, 54f.].

Dressler nennt weitere Prozesseigenschaften, die seine Nutzung im überbetrieblichen Bereich sinnvoll erscheinen lassen [Dressler, 2007, 41ff.]. Organisationsübergreifende Prozesse sollten eine beherrschbare Komplexität besitzen, die sich aus der Verwendung weniger, dafür jedoch gut strukturierter, definierter und beschriebener Serviceschnittstellen ergibt. Der Zugriff auf Basis- oder Zwischenservices fremder Unternehmen bringt eine bedeutend größere Prozesskomplexität und Schnittstellenanzahl mit sich, als die Verwendung von Enterpriseservices, die auf grobgranulare Prozessservices zurückgreifen, wie Abschnitt 2.1.4 entnommen werden kann. Zur Automatisierung eignen sich insbesondere Prozesse mit einer geringen Prozessqualität, einer langen Ausführungsdauer, hohen Prozesskosten, sowie eine unzureichenden oder fehlenden technischen Unterstützung. An genau diesen Stellen ergeben sich durch die Entwicklung organisationsübergreifender Prozesse für die Beteiligten die größten Einsparungs- und Rationalisierungspotenziale.

### 3.3 Grundlagen öffentlicher Services

Im Abschnitt 2.1.4 wurde bereits auf die verschiedenen Servicearten eingegangen, die innerhalb einer SOA auftreten können. Als die komplexeste Form gelten dabei die öffentlichen Services, die anderen Unternehmen zur Verfügung gestellt werden können. Zunächst ist zu klären, wie der Begriff „öffentlich“ zu verstehen ist. Nissen u. a. unterscheiden für diesen Zweck drei Servicekategorien für öffentliche Services, die während ihrer Entwicklung festzulegen sind [Nissen u. a., 2007, 63f.]:

- *Rein interne Services* werden nur innerhalb des Unternehmens genutzt und sind nicht dafür entworfen, um anderen Unternehmen zur Verfügung zu stehen.
- *Öffentliche Services* werden vom Unternehmen selbst erbracht und sind ausreichend gut beschrieben und leistungsfähig genug, um von externen Unternehmen genutzt werden zu können, beispielsweise im Rahmen einer organisationsübergreifenden Prozesskette.
- *Externe Services* sind von einem anderen Unternehmen fremd bezogene Services, die aufgrund fehlender Datengrundlage, Kostengründen oder organisatorischen Einschränkungen selbst nicht erbracht werden können oder sollen. Sie stellen damit öffentliche Services eines anderen Unternehmens dar.

Das Adjektiv „öffentlich“ bezieht sich damit nicht ausschließlich auf die Verfügbarkeit eines Service gegenüber anderen Services innerhalb der SOA eines Unternehmens über seine „offene“ Schnittstellenspezifikation (diese Eigenschaft erfüllen nur die rein internen Services), sondern auf seine Verfügbarkeit im unternehmensübergreifenden Bereich, das heißt dem Serviceaufruf über die Unternehmensgrenze hinweg. Zu den rein internen Services zählen damit zwangsläufig alle Basis-, Zwischen- und Prozessservices. Sowohl öffentliche Services, die das Unternehmen selbst erbringt und anderen bietet, als auch externe Services, die das Unternehmen fremd bezieht, sind damit „öffentlich“ beziehungsweise „public“, abhängig vom Standpunkt der Betrachtung.

Die Verwendung von öffentlichen Services als Bestandteil einer organisationsübergreifenden Prozesskette erfordert eine Reihe von Servicemerkmalen, die rein interne Services aufgrund ihrer Einsatzumgebung nicht im gleichen Maße aufzuweisen brauchen. Nach [Masak 2007, S. 77f.] gehört dazu die ausführliche Spezifikation aller funktionalen und nicht-funktionalen Eigenschaften, worunter auch eine semantische Beschreibung des Service gehört, um die gegebene Servicespezifikation in den konkreten Kontext eines Geschäftsprozesses einzubinden. Weiterhin müssen Services, die im Rahmen organisationsübergreifender Prozesse verwendet werden, ihren Status teilweise mehrere Tage lang behalten können, da diese oft sehr langlebig sind. [Krafzig u. a., 2007, 98] nennen zahlreiche weitere Anforderungen für öffentliche Services. Die Serviceschnittstellen auf Unternehmensebene haben ein sehr hohes Granularitätsniveau, welches mit dem Umfang von Geschäftsdokumenten verglichen werden kann. Mit der langen Ausführungsdauer geht die Verwendung asynchroner Serviceaufrufe einher, wodurch die Speicherung des Prozess- und Servicezustandes notwendig wird. Öffentliche Services

bewirken den Zugriff auf unternehmensinterne Daten und Anwendungen, weshalb hohe Standards für die Sicherheitsmaßnahmen und -mechanismen erforderlich sind. Dazu gehörten die Authentifizierung, Autorisierung, Verschlüsselung, Vertraulichkeit, Integrität, Nachweisbarkeit und Zugriffskontrolle [Margolis und Sharpe, 2007, 16]. Da öffentliche Services auch kostenpflichtig angeboten werden können, sind gegebenenfalls Abrechnungsmodelle zur Rechnungsstellung notwendig, die den Service-Cashflow überwachen und verwalten. Alle diese nichtfunktionalen Anforderungen können in Service-Level-Agreements (Servicequalitätsvereinbarungen) hinterlegt werden, wodurch jedoch eine Infrastruktur notwendig wird, um diese Vereinbarungen zu überwachen. Dies kann beispielsweise mithilfe von Monitoring-Werkzeugen geschehen [Masak, 2007, 112ff.].

Im Abschnitt 2.1.2 wurde bereits auf die verschiedenen Reifegrade einer SOA hingewiesen. Um die eben beschriebenen Anforderungen, an öffentliche Services, erfüllen zu können erscheint es sinnvoll, die SOA bereits bis zu höchsten Form der Prozessfähigkeit entwickelt zu haben. Erst in diesem Stadium sind Prozessservices in der Lage, die Komplexität der Back-End-Systeme zu kapseln, die Verarbeitung langlebiger Prozesse zu unterstützen, die in der Prozessschicht gekapselte Prozesslogik unabhängig von den anderen Schichten und Services verwalten und durch den Einsatz öffentlicher Services auch anderen Unternehmen anbieten zu können [Krafzig u. a., 2007, 106f.]. In umgekehrter Sicht ist die Integration fremder, öffentlicher, prozesszentrierter Services erst möglich, wenn das eigene Unternehmen mithilfe seiner SOA in der Lage ist, Prozesse auf Grundlage eigener und fremder Services zu unterstützen.

Nach [Opincaru, 2007, 61f.] haben viele Unternehmen eine Scheu davor, ihre „Geschäftsressourcen“ öffentlich verfügbar zu machen. Der Grund dafür liegt oftmals im geringen Know-how zur Einrichtung einer Sicherheitsinfrastruktur für die öffentlichen Services, um die möglichen Sicherheitsrisiken zu verringern. Auf dem Gebiet der Webservice Standards wurde für diesen Zweck eine Unmenge von sicherheitsrelevanten WS-\*-Standards entwickelt, wie WS-Security, WS-Trust, WS-Federation, WS-SecureConnection-, WS-Privacy, WS-Federation, WS-Authorisation, WS-Policy, WS-(Atomic)Transaction, WS-TXM LRA-, BP- und ACID, WS-ReliableMessaging, und so weiter. Diese Standards sind zwar dafür geeignet, um Webservices auf technischer Ebene sicher zu gestalten, geben jedoch keine Hinweise für die auf fachlicher, organisatorischer Ebene benötigte Infrastruktur zur Überwachung und Durchsetzung der festgelegten Sicherheitsbestimmungen. Erfahrungen zeigen weiterhin, dass Webservices durch die den Einsatz der oben genannten Standards sehr schnell an Komplexität zunehmen, wodurch deren Wiederverwendbarkeit und Übertragbarkeit erschwert wird [Opincaru, 2007, 61].

Sinnvoller kann es erscheinen, eine serviceorientierte Sicherheitsarchitektur zu entwickeln, mit deren Hilfe die oben genannten Sicherheitsanforderungen durch den Einsatz spezialisierter „Sicherheits“-Services abgedeckt werden. In diesem Fall werden die Sicherheitsbestimmungen aus den eigentlichen, öffentlichen Services ausgelagert und in Sicherheitsservices eingelagert, wodurch diese auch für andere öffentliche Services zur Verfügung stehen. Dadurch erstet oberhalb der öffentlichen Servicezugriffsschicht eine weitere Sicherheitsschicht, welche die tatsächliche „Außenansicht“ der SOA eines Unternehmens darstellt. Opincaru nennt zu diesem Zweck eine Reihe möglicher, öffentlicher Sicherheitsservices [Opincaru, 2007, S67f.]:

- *Authentifizierungsservices* dienen der Verifizierung und Identifizierung eingehender Nachrichten und kennzeichnen diese mit einer konkreten Rolle, damit nachfolgende Services wissen, wie die Nachricht verarbeitet werden soll.
- *Autorisierungsservices* dienen zur Prüfung der Berechtigungen einer Nachricht. Dabei können die Konzepte der Policy Administration-, Information-, Enforcing-, und Decision Points (PAP, PIP, PEP, PDP) zum Einsatz kommen.
- *Audit Services* können zum passiven Auditing (Logging der Vorgänge im System) und dem aktiven Auditing (Benachrichtigung bei Fehlern und Ausnahmen) verwendet werden.
- *Cryptographic Services* dienen der Verschlüsselung und digitalen Signierung ein- und ausgehender Nachrichten.

- *Accounting Services* können als Abrechnungsmechanismus verwendet und in eine Prozesskette eingebunden werden, um Buchungen vorzunehmen. In Kombination mit *Charging Services* können Unternehmen ein Servicenutzungsguthaben erwerben, um die Abrechnung zu erleichtern.

Es bleibt festzuhalten, dass die mit der Verwendung öffentlicher Services einhergehenden nichtfunktionalen Anforderungen, insbesondere im Bereich der Sicherheit, entweder in den öffentlichen Services selbst, oder in einer zusätzlichen Sicherheitsschicht untergebracht werden können. Der letzte Fall wird durch die folgende Abbildung verdeutlicht.

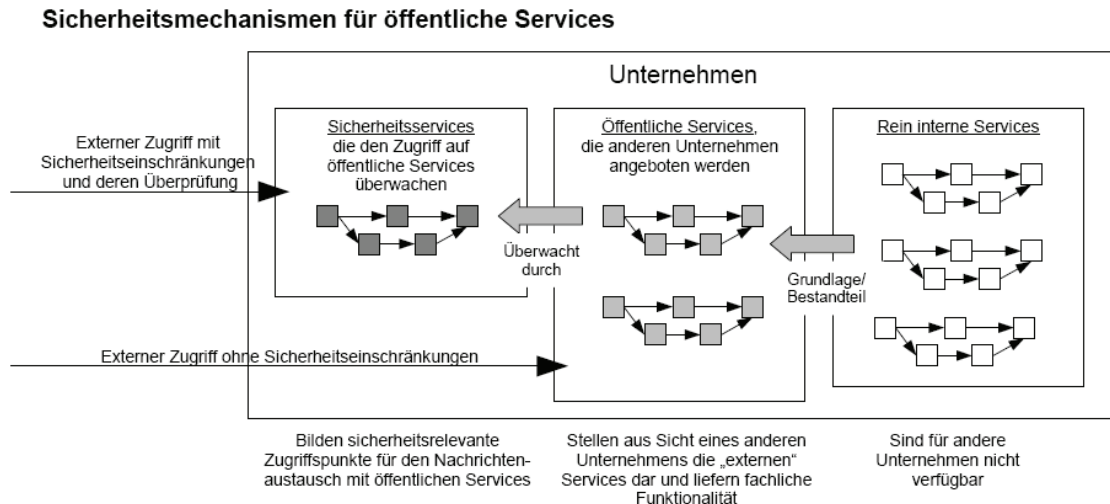


Abbildung 7: Sicherheitsarchitektur für öffentliche Services

Huvar u. a. empfehlen zur Verbesserung der Flexibilität einer solchen Servicearchitektur die Unterscheidung von Service-Konsumentengruppen. Für jeden öffentlichen Service werden die potenziellen Konsumenten charakterisiert und zu logischen Gruppen gebündelt und diesen Rechte und Pflichten zuweisen, die fortan für die Gruppenmitglieder gültig sind. Damit kann der Serviceanbieter für jede Gruppe individuelle Einstellungen vornehmen, beispielsweise hinsichtlich der für diese Gruppe benötigten Sicherheitsservices. Dadurch lässt sich die Konfiguration neuer Kunden leicht vornehmen, indem diese den bereits vorhandenen Gruppen zugewiesen oder diese angepasst werden. Die Konsumentengruppe spiegelt dadurch den Kontext des Abnehmers wieder und vereinfacht die Sicherheitsarchitektur bei einer großen Anzahl von Abnehmern [Huvar u. a., 2008, 199].

Wie das Zusammenspiel aus öffentlichen Services und Prozessen gestaltet werden kann, wird im folgenden Kapitel genauer erklärt, nachdem die Rahmenbedingungen des organisationsübergreifenden Prozessmanagements erklärt wurden, die als Grundlage der Entwicklung öffentlicher Service und Prozesse dienen.

### 3.4 Organisation und Architektur unternehmensübergreifender Prozesse und Services

#### 3.4.1 Unternehmensexterne Lösungsansätze für die Prozessgestaltung

Bei der Gestaltung der organisationsübergreifenden Prozessmodellierung unterscheidet [Steinbuch, 1998, 193ff.] mehrere verschiedene Lösungsansätze, die im Folgenden kurz umrissen werden.

Während der *Ausgründung* werden Geschäftsprozesse in rechtlich selbstständige Tochterunternehmen ausgelagert und anschließend von diesen bearbeitet. Dabei kann es sich um komplette Unternehmensteile handeln, wenn mehrere Geschäftsprozesse einer Geschäftseinheit betroffen sind. Das Dachunternehmen kann sich dadurch besser auf die Kerngeschäftsprozesse und geänderte Marktbedingungen konzentrieren, das Risiko bei unrentablen Geschäftsprozessen mindern, sowie eine bessere Kostenrechnung und -steuerung vornehmen. Für das Tochterunternehmen bieten sich unabhängige Optimierungs- und Spezialisierungsmöglichkeiten.

- Beim *Outsourcing* geschieht die Verlagerung prozessspezifischer Aufgaben an einen externen Anbieter auf Grundlage der „Make-or-Buy“-Analyse [Dressler, 2007, 68]. Auch dabei ist eine höhere Prozess- und Dienstleistungsqualität durch bessere Spezialisierung des Outsourcing-Partners möglich.
- Im Rahmen von *Joint Ventures* oder *Kooperationen* können Geschäftsprozesse unter Beteiligung der Kooperationspartner gemeinsam erbracht werden.
- Bei der *Geschäftsprozessweiterung* erfolgt eine Vorwärts- oder Rückwärtsintegration entlang der Wertschöpfungskette, bei der die Übernahme von Prozessen oder Prozessschritten zum Beispiel von Kunden, Lieferanten oder Finanzdienstleistungsunternehmen geschieht. Das Gegenteil bildet die *Geschäftsprozesseliminierung* bei der komplette Prozessketten, beispielsweise durch Verkauf oder Auflösung im Rahmen einer Firmenfusion entfallen können [Steinbuch, 1998, 193ff.].

Karle und Oberweis unterscheiden vier grundlegende Formen überbetrieblicher Zusammenarbeit anhand der Tiefe beziehungsweise Intensität der Verbindung zwischen zwei Unternehmen. Die unterste Stufe bildet die reine *Kommunikation* zwischen Unternehmen zum Zweck des Informationsaustausches. Sie bildet die Grundlage aller weiteren Stufen und umfasst auf technischer Ebene den Nachrichtenaustausch zur Datenübertragung. Auf der nächsten Stufe liegt die *Koordination*, die einen Mechanismus für den steuernden Einfluss auf Aktivitäten und Prozesse darstellt. Ziel dabei ist die Abstimmung gemeinsamer Aktivitäten zur Erreichung eines bestimmten Ergebnisses und damit der Verbesserung der Zusammenarbeit. Bei der *Kooperation* handelt es sich um ein Prinzip der arbeitsteiligen Leistungserbringung, das meistens durch Verträge oder Abmachungen zwischen rechtlich selbstständigen Personen oder Unternehmen zustande kommt. Die *Kollaboration* stellt einen Spezialfall der Kooperation dar, bei dem die Arbeit am gleichen Artefakt durch räumlich verteilte Aufgabenträger durchgeführt wird [Karle und Oberweis, 2006, 80].

Unabhängig davon, in welcher Form die organisationsübergreifende Zusammenarbeit rechtlich geregelt wird, ergeben sich für das Prozessmanagement öffentlicher Prozesse neue Anforderungen an die beteiligten Unternehmen. Der nachfolgende Abschnitt widmet sich diesem Aufgabenfeld.

### 3.4.2 Organisationsübergreifendes Prozessmanagement

Im vorherigen Abschnitt wurden Szenarien aufgezeigt, bei denen die Entwicklung und Verwaltung organisationsübergreifender Geschäftsprozesse notwendig ist. Als wichtigsten Bestandteil dessen nennen Scheer u. a. die standardisierte Beschreibung und Modellierung des internen und externen Leistungsspektrums der beteiligten Unternehmen, sowie den Schnittstellen zum Austausch der erbrachten Leistungen [Scheer u. a., 2002, 10].

Dadurch wechselt der Fokus zur Entwicklung, Ausführung und Überwachung von Geschäftsprozessen vom innerbetrieblichen auf den überbetrieblichen Bereich und stellt neue Herausforderungen an die beteiligten Partner. Scheer u. a. unterscheiden die entstehenden Anforderungen in drei Kategorien:

Als erstes ist ein *Wandel der Unternehmenskultur* notwendig, bei dem Widerstände und Unverständnis bei Mitarbeitern auf Fach- und Führungsebene beseitigt werden müssen, die der Freigabe von Wissen über die internen Strukturen, Prozesse und Daten kritisch gegenüber stehen. Der Einsatz einer *zentralen Organisation* für die Prozessgestaltung bewirkt die transparente Verteilung der Abhängigkeiten und dient der Entwicklung klarer Machtstrukturen und Kontrollinstanzen für die Prozessausführung. Damit einher geht die Erstellung eines Berechtigungskonzepts, in dem Aufgaben, Rollen, Rechte und Pflichten der beteiligten Organisationen festgelegt sind. Im letzten Schritt geschieht die Entwicklung der *zukünftigen, technischen Integrationsplattform*. Diese umfasst sowohl Sprachen für die Prozess- und Servicemodellierung, als auch technische Standards für den Nachrichtenaustausch, um die eigenen Prozess- und Anwendungsarchitekturen für die organisationsübergreifende Prozessausführung vorzubereiten [Scheer u. a., 2002, 11ff.]. Im vierten Kapitel dieser Arbeit wird auf RosettaNet als Rahmenwerk vorhandener Integrationsstandards genauer eingegangen.

Das organisationsübergreifende Prozessmanagement steht damit vor der Aufgabe, die Interoperabilität von mindestens zwei selbstständigen Unternehmen zu ermöglichen, die heterogene Softwaresysteme

im Einsatz haben können. Dabei können mehrere Interoperabilitätsebenen unterschieden werden, auf denen die Kooperation bzw. Kollaboration der Unternehmen vorliegt:

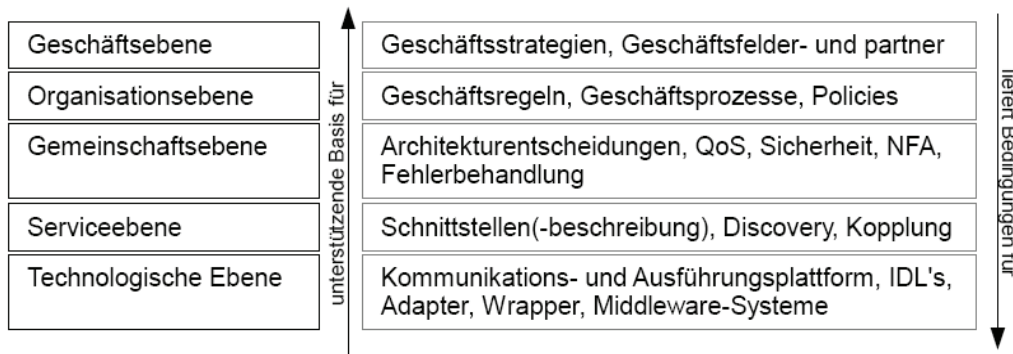


Abbildung 8: Integrationsebenen organisationsübergreifender Prozesse  
in Anlehnung an die textuelle Beschreibung von [Roukolainen und Kutvonen, 2005, 317ff.]

[Roukolainen und Kutvonen, 2005, 317] sprechen in diesem Zusammenhang auch von der Notwendigkeit zur Integration auf technischer, semantischer und pragmatischer Ebene, um die fehlerfreie Zusammenarbeit zu ermöglichen. Wenn in den beteiligten Unternehmen die Voraussetzungen für die Zusammenarbeit geschaffen wurden, erfolgt im nächsten Schritt die Modellierung der organisationsübergreifenden Prozesse. Damit beschäftigt sich der folgende Abschnitt.

### 3.4.3 Modellierung organisationsübergreifender Prozesse

Die Geschäftsprozessmodellierung stellt einen Teilbereich des Geschäftsprozessmanagements dar und bildet die Grundlage der späteren Prozessimplementierung. Zur Modellierung und Implementierung stehen heutzutage dutzende verschiedener Standards und Sprachen mit unterschiedlichen Konzepten zur Verfügung [Ziemann u. a., 2007, 87f.]. Die Modellierungssprachen unterstützen verschiedene Anforderungen unterschiedlich gut, beispielsweise hinsichtlich ihrer Flexibilität, Erlernbarkeit, Aussagekraft, Ausführbarkeit und ihrem Abstraktions-, Hierarchie- und Granularitätsebenen. Welche Anforderungen die zu verwendende Sprache erfüllen sollte, ergibt sich aus den besonderen Anforderungen organisationsübergreifender Prozesse, auf die im Folgenden eingegangen wird. Der vorherige Abschnitt beschäftigte sich mit der Notwendigkeit der organisationsübergreifenden Integration auf Geschäftsebene, auf deren Basis die technische Integration realisiert wird. Wenn unternehmensübergreifende Prozesse implementiert werden, sollten diese auch unternehmensübergreifend modelliert werden, wobei sich die zwischenbetriebliche Prozessmodellierung erheblich von der innerbetrieblichen unterscheidet. (Folgendes in Anlehnung an [Ziemann u. a., 2007, 88ff.]):

Die zwischenbetriebliche Prozessmodellierung liegt im Spannungsfeld zwischen Prozesstransparenz und Informationsgeheimhaltung. Dies entspricht der Freigabe firmeninterner Informationen über die vorhandenen Softwaresysteme, Sprachen, Schnittstellen und Prozesse und der Notwendigkeit nur die Informationen freizugeben, die für den öffentlichen Prozess unbedingt benötigt werden. Werden zu viele Informationen freigegeben, entstehen Sicherheitsrisiken, werden zu wenige Informationen freigegeben, wird die Prozessmodellierung unnötig erschwert. Konzepte zur Behandlung dieses Problems sind beispielsweise die Unterscheidung statischer und dynamischer Schnittstellen oder die Trennung öffentlicher und interner Sichten/Ebenen als Bestandteil der Modellierungssprache.

Die formale Spezifikation der öffentlichen Serviceschnittstellen erfordert Konzepte zur Prüfung von Berechtigungen. *Wann* benötigt *welcher* Partner *welche* Informationen von *welchem* Service? Im unternehmensinternen Bereich müssen solche Berechtigungen häufig nicht geprüft werden, da alle Zugriffe innerhalb einer geschlossenen Ausführungsumgebung (Intranet, LAN) stattfinden. Im überbetrieblichen Bereich ergeben sich diese Berechtigungen auf technischer Ebene direkt aus dem Kontext des Geschäftsprozesses und müssen deshalb bei dessen Modellierung berücksichtigt werden. Bei der Verbindung mit vielen Partnern ist dafür ein Rollen- oder Benutzerkonzept notwendig, mit dessen Hilfe die Anfragen der Partner identifiziert, entsprechend ausgewertet und an den zuständigen Service delegiert werden können.

Bei der Abbildung des Datenflusses zwischen Unternehmen gelten ebenfalls andere Bedingungen, als innerhalb eines Unternehmens. Unternehmensinterne Zugriffe sind in der Regel nicht kostenpflichtig, im Sinne eines abzurechnenden Dienstes gegenüber einer anderen Abteilung. Jedoch müssen sämtliche Datenein- und -ausgaben bei der Verbindung mit dem Partner zumindest verifiziert oder protokolliert, gegebenenfalls aber auch abgerechnet werden, wofür ein Abrechnungssystem einzurichten und in Betrieb zu nehmen ist. Einen weiteren Punkt bildet die Analyse und Optimierung organisationsübergreifender Prozesse, die sowohl unternehmensintern, als auch übergreifend geschehen sollte. Dies beinhaltet das Monitoring laufender und abgeschlossener Geschäftsprozesse zur Erkennung und Reaktion auf Schwachstellen und Verbesserungspotenziale im eigenen, aber auch bei fremden Unternehmen. Welche Leistungsfähigkeit von den Partnern innerhalb der Prozesskette gefordert ist, sollte ebenfalls während der Modellierung berücksichtigt und festgelegt werden.

Den schwierigsten Schritt bildet jedoch die Übertragung von fachlichen Geschäftsprozessen auf ausführbare Prozesse auf technischer Ebene. Nach [Hoyer u. a., 2007, 187] scheitert diese Transformation häufig an der großen Komplexität organisationsübergreifender Prozesse auf fachlicher Ebene, unzureichenden, global akzeptierten e-Business-Standards zu ihrer Modellierung und den Beschränkungen möglicher Ausführungsumgebungen auf technischer Ebene. Um organisationsübergreifende Geschäftsprozesse erstellen zu können, muss die Heterogenität auf semantischer Ebene durch den Einsatz von Dictionaries, Terminologien und Ontologien und auf Beschreibungsebene durch die Verwendung einer ausreichend mächtigen Modellierungsnotation und -sprache gewährleistet werden, mit deren Hilfe sich alle wichtigen Aspekte abbilden lassen [Ziemann u. a., 2007, 89].

Dabei ist es jedoch nicht zwingend notwendig, dass alle Partner ihre Geschäftsprozesse in dieser Sprache abbilden, sondern lediglich ein Mechanismus oder Transformationskonzept zur Verfügung steht, um die unternehmensintern verwendete Sprache auf die organisationsübergreifend verwendete Sprache abzubilden. Die Einigung auf eine einheitliche Serviceschnittstellenbeschreibungssprache abstrahiert dadurch sowohl die intern, als auch extern zu verwendende Sprache. Ob ein Service als Aktionsinstanz eines UML Aktivitätsdiagramms, einer eEPK oder innerhalb des Knotens eines Petri-Netzes aufgerufen wird, erscheint irrelevant, solange dem Service alle zur Ausführung benötigten Informationen übergeben werden<sup>2</sup>. Eine mögliche „zentrale“ Ausführungssprache für organisationsübergreifender Prozesse ist die Business Process Execution Language (BPEL) bei der die Transformation mittels EPC2BPEL, BPMN2BPEL oder UML2BPEL geschehen kann [Hoyer u. a., 2007, 189f.]. Hoyer beschreibt zu diesem Zweck folgendes Transformationskonzept:

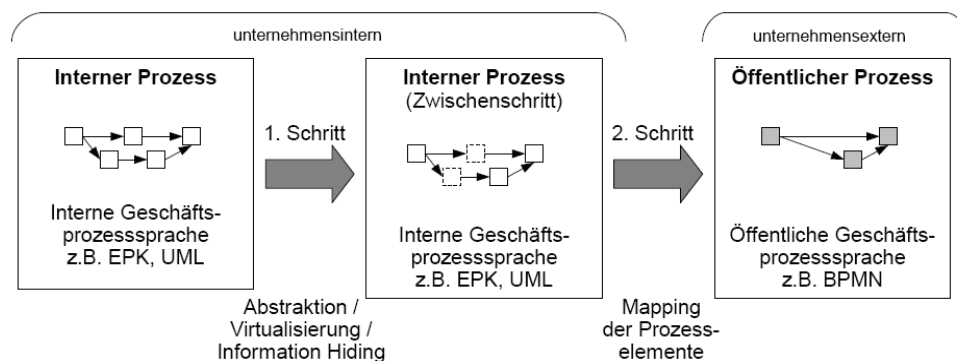


Abbildung 9: Transformationskonzept privater zu öffentlichen Prozessen  
in Anlehnung an [Hoyer u. a., 2007, 190]

Der Transformation liegen dabei verschiedene Regeln zugrunde (in Anlehnung an [Hoyer u. a., 2007, 191ff.]):

- alle für das Partnerunternehmen nicht-relevanten, trivialen Ereignisse werden eliminiert

<sup>2</sup> Siehe dazu auch: Huth, Stefan; Wieland, Thomas: Geschäftsprozessmodellierung mittels Software-Services auf Basis der EPK, in [Nissen, 2007, S. 61ff.], Deutscher Universitätsverlag, Wiesbaden, 2007.

- alle statischen Ausführungsinformationen werden gemäß dem Prinzip der Service-Kapselung entfernt (wodurch wird der öffentliche Service X erbracht/implementiert), um lose Bindungen zu ermöglichen
- nur Services, die Daten nach außen senden oder von dort empfangen, werden veröffentlicht, alle internen Verarbeitungsschritte (Servicekopplungen) bleiben verborgen
- Prozess-Unterprozess-Beziehungen bleiben verborgen, für den Partner ist es nicht relevant, wie ein Prozess im Unternehmen abläuft
- alle öffentlichen Services werden um Ausführungsbedingungen ergänzt, gemäß den Geschäftsregeln und der vorhandenen Sicherheitsarchitektur
- alle Entscheidungsknoten (AND, OR, XOR) werden mithilfe der zentralen Ausführungssprache abgebildet, wobei die Transformation an dieser Stelle für jede interne verwendete Sprache spezifisch ist

Durch die Unterscheidung verschiedener Betrachtungsweisen ergeben sich verschiedene Arten von Geschäftsprozessmodellen, die im organisationsübergreifenden Kontext auftreten [Ziemann u. a., 2007, 90]. *Globale Prozessmodelle* beinhalten alle Geschäftspartner, deren angebotene und benötigte Leistungen und regeln alle Interaktionen zwischen den Partnern. Die Erstellung eines globalen Prozessmodells bildet die eine Seite des organisationsübergreifenden Prozessmanagements. Auf der anderen Seite stehen *öffentliche Prozessmodelle*, die einen Teilbereich der privaten Prozessmodelle beinhalten, die für das Unternehmensnetzwerk in Form eines globalen Prozessmodells, relevant sind. Sie werden aus der Sicht jedes beteiligten Unternehmens individuell beschrieben und enthalten alle öffentlichen Services, die angeboten werden und externe Services, die verwendet werden.

Die Grundlage der eben genannten Modelle bilden die *internen, privaten Prozessmodelle*, die ebenfalls aus individueller Sicht beschrieben und nur innerhalb des Unternehmens bekannt sind und dort verwendet werden. Sie enthalten dementsprechend alle rein internen Services und deren Zusammenspiel mit den öffentlichen Services. Folgende Abbildung verdeutlicht den Zusammenhang:

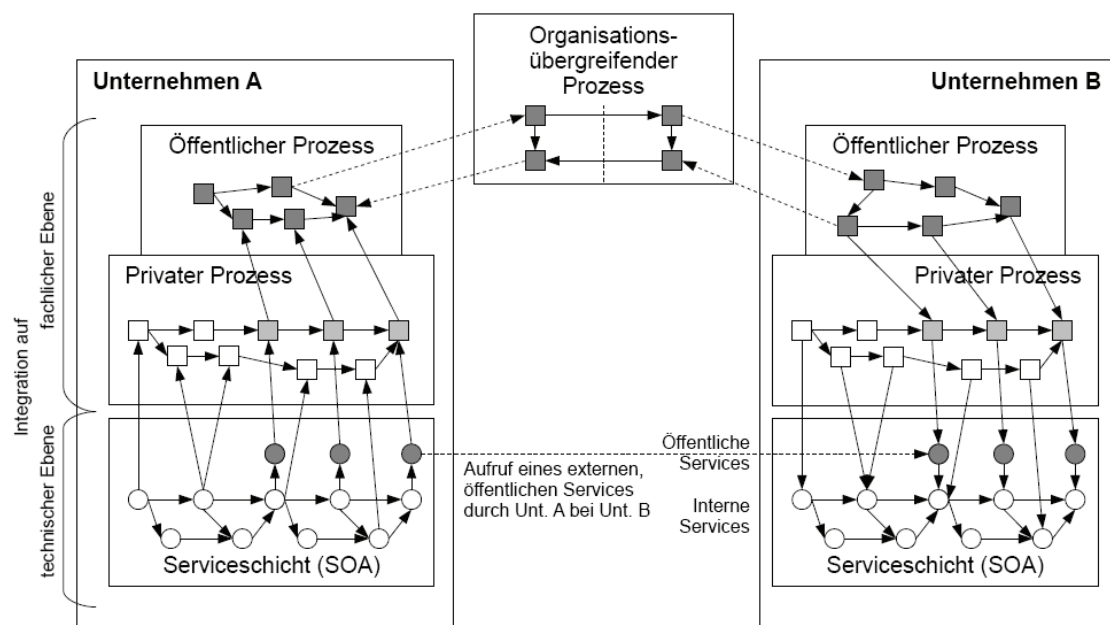


Abbildung 10: Organisationsübergreifende Prozessintegration

Greiner unterscheidet dabei zwei verschiedene Kooperationsmodelle [Greiner, 2006, 8f.]. Beim *einfachen Kooperationsmodell* kennen die Partner den gesamten Prozess nicht und sind daher nicht gleichberechtigt. Die Prozesskontrolle unterliegt dem Partner, der den Prozess im Moment ausführt und sich zur Erledigung der Aktivitäten seines Prozesses den öffentlichen Services eines anderen Partners bedient. Damit geht eine lose Kopplung des Netzwerkes und hohe Autonomie der Teilnehmer einher.

Im Falle des *komplexen Kooperationsmodells* sind die Partner in das zentrale, globale Prozessmodell eingebunden und stimmen diesem zu. Dadurch sind alle Teilnehmer gleichberechtigt, teilen sich die Kontrolle über die Prozesse, sind eng gekoppelt und haben dadurch eine geringere Autonomie. Die obere Abbildung veranschaulicht dieses Kooperationsmodell mit zwei Unternehmen, würde jedoch auch mit mehr Unternehmen funktionieren. Wie die genannten Kooperationsmodelle ausgeführt werden können, wird im nachfolgenden Abschnitt beschrieben.

Tao und Yang empfehlen zur besseren Verwaltung und Konfiguration der Geschäftsprozesse ein differenziertes Servicedesign. Die Differenzierung bezieht sich in diesem Falle nicht auf die Art und Position des Service innerhalb der Servicelandschaft, sondern die Differenzierung öffentlicher Services gegenüber den potenziellen Verwendern. Auf das Prinzip des Service Oriented Computing wurde bereits im Abschnitt 2.1.4 eingegangen, dort zur Unterscheidung von Servicearten. Häufig benötigen mehrere Verwender den gleichen Service und umgekehrt muss ein Service Ergebnisse für verschiedene Verwender im gleichen oder einem anderen Geschäftsprozess liefern. Diese Anforderung bringt die Notwendigkeit mit sich, unterschiedliche Qualitätsniveaus oder Interaktionsmuster für die Services zu implementieren, wenn die Verwender mit verschiedenem Verhalten und Semantik auf den gleichen Service zugreifen. Als Beispiel sei hier ein Service zur Abrechnung des Warenkorbes für einen Webshop genannt, der in Abhängigkeit der Kundengruppe des Verwenders (Stammkunde, Neukunde) einen Rabatt gewährt [Tao und Yang, 2007, 194].

Das Konzept der Verarbeitung eines Serviceaufrufs unter einem bestimmten Verwendungshintergrund, wird als „Service Differentiation“ bezeichnet und kann dazu dienen, die Leistungsfähigkeit der Services innerhalb des Geschäftsprozesses zu erhöhen, die Anzahl der öffentlichen Services zu verringern und die Flexibilität des Geschäftsprozesses insgesamt zu erhöhen. Jeder Serviceaufruf geschieht in diesem Falle durch einen identifizierbaren Verwender, für den Geschäftsregeln (business policies) hinterlegt sind. In der Vergangenheit wurden diese Geschäftsregeln oftmals hart in den Services codiert oder Services von vorn herein auf einen spezifischen Verwendungskontext ausgelegt, ohne diesen zur Laufzeit unterscheiden zu können. Sinnvoller ist es jedoch, verschiedene Interfaces für einen Service zu unterscheiden, die dem Verwendungskontext entsprechen und gemäß den für die Verwender hinterlegten Geschäftsregeln aktiv werden. Demgemäß wird mithilfe der Abstraktion eine abstrakte Servicebeschreibung geschaffen, die im äußersten Fall zur Modellierung vollständig abstrakter Geschäftsprozesse (abstract business process, ABP) führen kann. Tao und Yang sprechen in diesem Zusammenhang von „policy configured business processes“ (PCBP)<sup>3</sup> [Tao und Yang, 2007, 196]. BPEL unterstützt als Ausführungssprache die Definition von abstrakten und konkreten Servicechnittstellen, worauf im Abschnitt 3.4.6 genauer eingegangen wird. Nach der Modellierung kann mit der Implementierung begonnen werden.

### 3.4.4 Implementierung organisationsübergreifender Prozesse

Bussler hat sich ausführlich mit der Implementierung öffentlicher und privater Prozesse beschäftigt, unabhängig von der verwendeten Software und deren Herstellern, und führt für beide Begriffe eine Reihe von Implementierungsrichtlinien an, wie deren Verhalten auf technischer Ebene beschrieben werden kann. Private und öffentliche Prozesse werden durch die Verarbeitung von Ereignissen an den Endpunkten der verwendeten B2B-Integrationstechnologie spezifiziert. Die verwendete Technologie, z. B. RosettaNet liefert Vorgaben, für die Art und Repräsentation der Nachrichten und der Daten innerhalb der Nachrichten. Alle Schnittstellen werden mithilfe einer Schnittstellenbeschreibungssprache spezifiziert und dienen der Implementierung eines Schnittstellenprozesses. Dadurch bewirkt der Aufruf einer Schnittstelle den Beginn eines Prozesses, wobei ein Geschäftsprozess aus einer Vielzahl von Unterprozessen bestehen kann, die jeweils durch eine bestimmte Schnittstelle zur Verfügung gestellt und durch sie abgewickelt werden [Bussler, 2003, 123].

Während der Prozessbindung wird ein fachlich definierter Prozess oder Teilprozess auf eine eindeutig identifizierbare Schnittstelle abgebildet, die damit innerhalb des Prozesses Verwendung findet. Zu

---

<sup>3</sup> Siehe auch: Chun, Soon A.; Atluri, Vijayalakshmi; Adam, Nebil R.: Using Semantics for Policy-Based Web Service Composition, Springer Science and Business Media Inc., Niederlande, 2005



diesem Zweck muss ein Mapping von den Geschäftsprozessmodellschritten auf die jeweilige Schnittstelle durchgeführt werden, die später die Grundlage zum Routing der Nachrichten bildet. Für jede Schnittstelle wird dabei Namenseindeutigkeit gefordert, um Fehler beim Routing zu verhindern. Dabei kann jede Schnittstelle der Theorie zufolge in beliebig vielen Prozessen verwendet werden, aber jeder Prozessschritt kann je nach Art der Anfrage auch durch verschiedene Schnittstellen bearbeitet werden. Jede Schnittstelle und damit der durch sie ausgelöste Prozess wird durch die Prozessbindung eindeutig einem Geschäftsprozess zugewiesen [Bussler, 2003, 124].

Wie bereits angedeutet, kommunizieren die Endpunkte untereinander mithilfe von Nachrichten und verwenden dafür besonders im überbetrieblichen Bereich sehr häufig asynchrone Kommunikation mit persistenten Warteschlangen. Die Verarbeitung der Nachrichten geschieht „hinter“ dem Endpunkt und ist von außerhalb des Endpunktes nicht sichtbar, dies entspricht der Kapselung von Wissen durch die Verwendung von Services. Um die zuverlässige Kommunikation der Endpunkte zu gewährleisten, müssen sich die Endpunkte miteinander verstehen und in ihrem Verhalten übereinstimmen. Das Verhalten der Endpunkte ergibt sich aus der innerhalb der Geschäftsprozesse definierten Prozesslogik. Dafür ist die Ausführungsumgebung als Bestandteil der B2B-Integrationstechnologie zuständig, die nicht nur den Nachrichtenaustausch zwischen den Endpunkten, sondern auch zwischen den Endpunkten und ihren zugehörigen Anwendungen abwickelt. Während der Ausführungszeit muss sichergestellt sein, dass die Endpunkte auf die Eingaben der Prozessschnittstelle reagieren und die Daten in den Nachrichten an die entsprechende Anwendung weiterleiten. Alle Abweichungen vom gewünschten Verhalten, zum Beispiel durch Ausfall einer Anwendung oder eines Endpunktes, müssen an den Auslöser der Anfrage (Servicenutzer) und den Serviceerbringer gemeldet und von diesen entsprechend behandelt werden. Dafür muss der Endpunkt darüber Bescheid wissen, von wem er seine Daten empfangen hat und wer im Fehlerfall zu benachrichtigen ist [Bussler, 2003, 121f.].

Ein öffentlicher Prozess stellt definitionsgemäß die Möglichkeit zur Verfügung, Nachrichten gemäß einem festgelegten Nachrichtenaustausch und -datenformat zwischen zwei Organisationen über ein Netzwerk auszutauschen. Die genaue Definition des Matchings wird mithilfe von B2B-Protokollen realisiert, von denen RosettaNet Partner-Interface-Processes im Abschnitt 3.5.4 genauer erklärt werden. Die folgende Abbildung verdeutlicht den Zusammenhang.

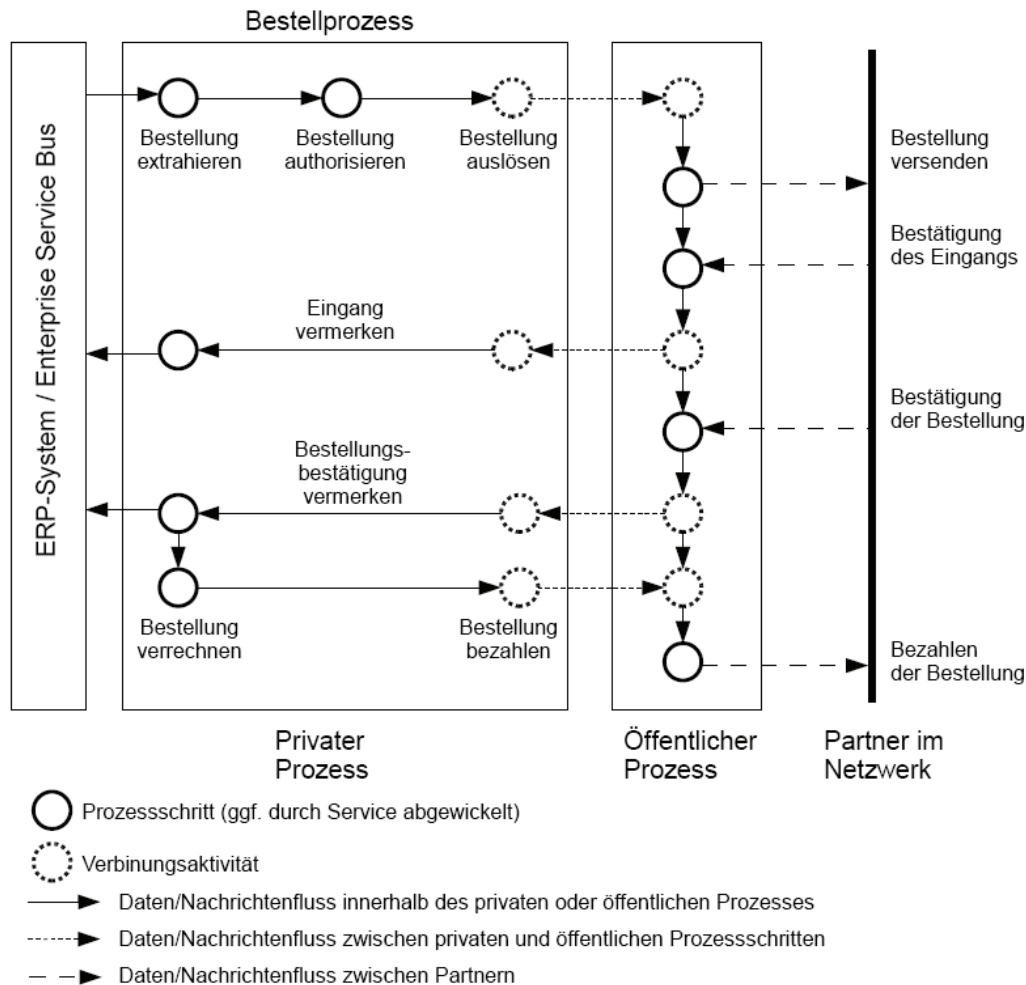


Abbildung 11: Implementierung überbetrieblicher Prozesse am Beispiel der Bestellung in Anlehnung an [Bussler, 2001, 21]

[Casati u. a., 2001, 18ff.] unterscheiden zur Modellierung öffentlicher Prozesse eine Reihe von Kernkonzepten von B2B-Protokollen, die sich in den verschiedenen Sprachen und Frameworks wieder finden. Die Kommunikation zwischen den Partnern geschieht mithilfe von Nachrichten, die ein akzeptiertes *Nachrichtenformat* haben müssen, um von beiden Seiten gleichermaßen verstanden zu werden. Das Nachrichtenformat definiert den Kontext der Nachricht innerhalb des Geschäftsablaufs und wird häufig durch XML-DTD oder XML-Schema spezifiziert. Zum Austausch der Nachrichten sind *Aktivitäten* notwendig, die angeben, wann welche Nachrichten gesendet und empfangen werden.

Dabei werden *Geschäftsnachrichten* zur Übertragung von Geschäftsdaten und *Bestätigungsnachrichten* unterschieden, mit denen der Empfang der Geschäftsnachrichten bestätigt wird (siehe vorherige Abbildung). Ein Kernkonzept bildet die Verwendung von *Rollen*, die den beteiligten Partnern zugewiesen werden und die sowohl zur Beschreibung von Rechten und Pflichten der Partner (business policies), als auch zur Identifikation der Nachrichten unerlässlich sind. Weitere Konzepte sind die Time-Out- und Wiederholungslogik zur Sicherstellung des Nachrichtentransfers über unsichere Netzwerke, aber auch die Duplikatprüfung und ggf. Ablehnung zur Gewährleistung der exactly-once Fehlersemantik, die in den meisten Fällen verwendet wird. RosettaNet wurde speziell für den unternehmensübergreifenden Nachrichtenaustausch entwickelt.

Im Abschnitt 3.4.2 wurde auf die Organisation des Prozessmanagements auf *fachlicher Ebene* eingegangen. Die Implementierung öffentlicher Prozesse umfasst auf *technischer Ebene* eine Reihe von Tätigkeiten, die im Folgenden kurz beschrieben werden, in Anlehnung an [Bussler, 2001, 25]:

- Unter dem Begriff *Advertisement* wird das Angebot von öffentlichen Services gegenüber den Partnern verstanden. Die Definition des öffentlichen Prozesses muss dort gespeichert sein, wo sie von den Partnern gefunden werden kann. Dabei wird bekannt gegeben, mithilfe welcher formalen Sprache und welchen Standards der Service implementiert wurde.
- Mit dem Begriff *Discovery* ist das Auffinden der Definition und Schnittstellenbeschreibung eines öffentlichen Service gemeint, wofür ein zentrales Verzeichnis verwendet werden kann, in dem alle Services erfasst und öffentliche Services entsprechend gekennzeichnet sind.
- Bei der *Selection* geschieht die Auswahl eines öffentlichen Service, zu dem ein Vertrag (*Contract*) vereinbart wird, der beispielsweise die bei der Verwendung entstehenden Kosten, die Qualität des Service und andere Anforderungen enthält. Diese Angelegenheiten können oftmals nur im persönlichen Gespräch geklärt werden.
- Mithilfe der *Composition* können sich die Partner mehrere Services zusammenstellen und zu einer komplexen Dienstleistung integrieren. Beispielsweise kann ein Automobilkonzern die öffentlichen Services mehrerer Zulieferer komponieren, um Anfragen starten zu können, welche Bauteile zu welchem Preis verfügbar sind.
- Beim *Monitoring* geschieht zum einen die Überwachung des Ausführungsstatus oder Service, zum anderen die Einhaltung der mit dem Service vereinbarten Leistungen, den Service-Level-Agreements. Das Monitoring sollte dabei von beiden Partnern durchgeführt werden, um die Prozessqualität erhöhen und überprüfen zu können.

Bussler hat sich weiterhin mit verschiedenen B2B-Implementierungsmethodologien beschäftigt, die sich grundlegend in Top-Down- und Bottom-Up-Ansätze unterscheiden lassen.

Der *Top-Down-Ansatz* beginnt mit der Modellierung der Geschäftsprozesse auf hoher, fachlicher Ebene und verfeinert die dabei entstandenen Modelle Schritt für Schritt über die notwendigen Workflows zur Ausführung der Prozesse, Services zur Unterstützung der Workflows, Bindungen und Interfaces zur Kommunikation zwischen den Services und, auf niedrigster Ebene, der Definition von Endpunkten und dem Nachrichtenaustausch zwischen diesen Endpunkten. Im letzten Schritt kommen alle notwendigen Regeln zur Gültigkeitsprüfung (Constraints), Qualitätsmessung und Gewährleistung der Ausführungs- und Zugriffssicherheit hinzu, die sich ebenfalls aus den Geschäftsprozessmodellen und den darin enthaltenen Rollen und Berechtigungen ableiten lassen.

Der *Bottom-Up-Ansatz* bildet das Gegenteil zum Top-Down-Ansatz und beginnt mit der Definition aller notwendigen Daten, dann den Ereignissen und Nachrichten zur Transformation der Daten, deren Zusammensetzung zu Prozessketten und so weiter. In diesem Fall werden zuerst die Endpunkte definiert und anschließend festgelegt, welche davon in welchem Geschäftsprozess Verwendung finden, [Bussler, 2003, 307ff.]. Beide Ansätze können erhebliche Probleme mit sich bringen, beispielsweise wenn beim Top-Down-Ansatz bereits vorhandene Anwendungen oder Prozesse vernachlässigt werden, oder sich die Komposition vorhandener Endpunkte und Services zu komplexen Prozessketten beim Bottom-Up-Ansatz wegen technischer Probleme als schwierig oder gar undurchführbar erweist.

### 3.4.5 Ausführung organisationsübergreifender Prozesse

Bevor mit der Ausführung begonnen werden kann, müssen die fachlichen Prozessmodelle in eine technische Beschreibungsform überführt werden, die so genannten technischen Prozessmodelle. Im Abschnitt 2.2.3 wurden fachliche Prozessmodelle bereits beschrieben. Sie dienen der Sammlung, Präzisierung und Abstimmung von Regeln und Vorgehensweisen zur Abwicklung von geschäftlichen Vorfällen [Störrle und Glock, 2007, 84]. Fachliche Prozessmodelle liefern damit den wesentlichen Input für technische Prozessmodelle, zu denen auch Servicemodelle gehören können, wenn die Prozesse durch Services unterstützt werden sollen. Die technischen Prozessmodelle zeigen damit die geplante Umsetzung eines fachlichen Modells auf einer möglichen Ausführungsplattform, beispielsweise in Form einer SOA.

Zu diesem Zweck wird das fachliche Prozessmodell um verschiedene Architekturkonzepte erweitert und verfeinert [Störrle und Glock, 2007, 87f.]. Dazu gehört die Transaktionskontrolle (welche Ände-

rungen innerhalb der Prozesskette müssen im Fehlerfall zurückgenommen werden), Zugriffsbeschränkungen, Protokollierung, Monitoring, Verschlüsselung und so weiter. Das Ergebnis bildet ein ausführbares Modell mit fertigem Quellcode, beispielsweise in Form von BPEL- oder Java-Code. Diese Transformation verändert die Struktur des fachlichen Modells im Hinblick auf bestimmte Ziele, in Bezug auf die Menge und Art des Datenaustauschs oder die Zuordnung vorhandener oder neu zu entwickelnder Services zur Unterstützung eines Prozessschrittes.

Weiterhin kann es notwendig sein, die Struktur des fachlichen Prozesses zu ändern, um seine Ausführbarkeit zu verbessern. Bei der Datenflussoptimierung ist die *Ein- oder Ausgliederung* einzelner Tätigkeiten in den Prozess möglich, je nachdem, ob eine Teilaktivität noch für einen anderen Prozess benötigt wird, sowie die *Umgliederung* bzw. Umordnung, das heißt der zeitlichen oder logischen Änderung der Reihenfolge bei der Prozessausführung. Im Rahmen der *Spezialisierung* werden grobgranulare, fachliche Aktivitäten auf technischer Ebene weiter verfeinert und gegebenenfalls weitere Prozessschritte oder Services eingefügt. Bei der *Sequenzialisierung* wird die fachlich modellierte Nebenläufigkeit aufgelöst, falls diese technisch nicht realisierbar ist oder die Laufzeit unterschiedlicher Wege zu stark schwankt und dadurch Probleme verursachen würde [Störrle und Glock, 2007, 89f.]. Jedes Unternehmen nimmt diese Transformation selbst vor und beachtet währenddessen, welche Prozesssteile und damit Services öffentlich verfügbar sein sollen. (Siehe dazu auch Abschnitt 3.3.3)

Nach [Andonoff u. a., 2005, 120f.] stellen fachliche Prozessmodelle jedoch nur eine von drei grundlegenden Modellarten dar, die zur Ausführung notwendig sind. Weiterhin notwendig ist ein Unternehmensmodell. In diesem werden alle beteiligten Partner und deren Rechte und Pflichten bei der Prozessverarbeitung erfasst, woraus die notwendigen Sicherheitsmechanismen und -bestimmungen resultieren, die über die ausführende Workflow-Schicht bis zur Serviceschicht durchgreifen. Zuletzt gibt es ein Informationsmodell, mit dessen Hilfe sich die Partner auf standardisierte Dokumentstrukturen und -formate einigen. Erst das Zusammenspiel dieser drei Modellarten ermöglicht die konsequente Planung und spätere Ausführung der Geschäftsprozesse.

Nach [Masak, 2007, 79ff.] werden Workflow-Managementsysteme (WFMS) sehr häufig zur Steuerung und Koordination von Prozessen und Services eingesetzt. Für den organisationsübergreifenden Bereich können dabei zwei Kategorien zur Verbindung von WFMS unterschieden werden, die sogenannte *serviceorientierte* Kopplung auf technischer Ebene und die *aktivitätsorientierte* Kopplung auf fachlicher Ebene.

Bei der *serviceorientierten Kopplung* werden alle Aktivitäten vollständig in Services gekapselt und somit durch Services bereitgestellt. Ihre Verbindung geschieht auf niedriger, technischer Ebene, ohne Zuhilfenahme eines übergeordneten, fachlichen Modells. Hierbei wird zwischen zentralisierter und verteilter serviceorientierter Ausführung unterschieden. Bei der verteilten Ausführung werden die dem Service zugrunde liegenden Aktivitäten nur vom lokalen System gesteuert und es erfolgt keine organisationsübergreifende Koordination der Serviceausführung. Das Gegenteil bildet die zentralisierte Ausführung, bei der ein zentrales System die Steuerung und Kontrolle des Nachrichtenaustausches zwischen den Partnern übernimmt und zu diesem Zweck ein organisationsübergreifendes Prozessmodell benötigt und verwendet. Die von den Partnern angebotenen Services werden mithilfe entsprechender Beschreibungssprachen zu einem komplexen Ablauf kombiniert.

Bei der *aktivitätsorientierten Kopplung* basiert die Verbindung der lokalen Systeme auf der Spezifikation des organisationsübergreifenden Prozesses, der die Grundlage eines kooperativen, ausführbaren Workflows bildet. Im Gegensatz zur serviceorientierten Kopplung ist die Ausführung der Prozessschritte durch die höhere Abstraktionsebene nicht auf Services beschränkt. [Greiner, 2006, 12ff.] unterscheidet in diesem Zusammenhang ebenfalls zwischen *zentralen* und *dezentralen* Ansätzen. Bei der zentralen Ausführung übernimmt ein interorganisatorisches Ausführungssystem die Prozessausführung und enthält zu diesem Zweck ein globales, organisationsübergreifendes Prozessmodell, wie in Abbildung 10 zur organisationsübergreifenden Prozessintegration zu erkennen ist. In diesem Fall „bedient“ sich die zentrale Ausführungsumgebung den öffentlichen Prozessen der Partner zur Steuerung der organisationsübergreifenden Prozesse. Sie dient damit sowohl als fachliche, als auch technische Integrationsplattform der beteiligten Partner. Dadurch ist beispielsweise eine Fehlerbehandlung über mehrere Partner hinweg möglich. [Masak, 2007, 79] bezeichnet diesen als *zentralen oder föderalen*

*Prozesskoordinator*, der die korrekte Ausführung des organisationsübergreifenden Prozesses auf Basis der vorhandenen Prozessspezifikationen durch den Aufruf der Services der beteiligten Subsysteme übernimmt. Die Subsysteme der Partner können dabei einen *Koordinationsagenten* beinhalten, der die Kommunikation der eigenen Ausführungsumgebung mit dem Prozesskoordinator abwickelt. Der Koordinationsagent muss dafür Steuerungs- und Überwachungsfunktionalitäten bereitstellen.

Bei der verteilten, aktivitätsorientierten Prozessausführung verwendet jeder Partner seine eigene Ausführungsumgebung und führt einen Teil der gesamten, globalen Prozesskette aus. Hat er seine Aufgaben bearbeitet, wird die Prozessausführung an den nächsten Partner übergeben, indem ein entsprechender Aufruf beim Partner geschieht. In diesem Falle gibt es zwar ebenfalls ein globales, organisationsübergreifendes Prozessmodell, aber dessen Ausführung geschieht dezentral. Eine Fehlerbehandlung über mehrere Partner hinweg ist damit bedeutend schwieriger zu realisieren. Dies wird besonders deutlich, wenn betrachtet wird, dass ein organisationsübergreifender Prozess eine Transaktion auf hoher, fachlicher bzw. semantischer Ebene darstellt, der aus mehreren Transaktionen auf niedriger, technischer Ebene der einzelnen Partner besteht. In der folgenden Tabelle werden die beschriebenen Ansätze zusammengefasst.

	<b>dezentraler Ansatz</b>	<b>zentraler Ansatz</b>
<b>fachliche Ebene</b>	verteilte, aktivitätsorientierte Ausführung	zentrale aktivitätsorientierte Ausführung
<b>technische Ebene</b>	verteilte, serviceorientierte Ausführung	zentrale, serviceorientierte Ausführung

Tabelle 7: Ausführungsvarianten organisationsübergreifender Prozesse

Die Auswahl des entsprechenden Ansatzes hat Auswirkungen auf die Autonomie, Flexibilität und Kontrollmöglichkeiten der Partner und bildet eine wichtige Entscheidung im Rahmen des organisationsübergreifenden Prozessmanagements [Greiner, 2006, 32]. Bei den zentralen Ansätzen ergibt sich die Frage, welcher Partner das zentrale System steuert und welche Einflussmöglichkeiten jeder Partner auf dieses System haben soll. Unabhängig vom verwendeten Ausführungsansatz ergibt sich die Notwendigkeit, Zugriffs- und Sicherheitsmechanismen für öffentliche Prozesse festzulegen, in denen jeder Partner konkret festlegen kann, welche eigenen, öffentlichen Services innerhalb einer organisationsübergreifenden Prozesskette verwendet werden können. Die Mechanismen auf Serviceebene, wie sie im Abschnitt 3.2. zu den Sicherheitsservices beschrieben wurden, sind dafür verwendbar, liegen jedoch auf einer niedrigen Abstraktionsebene. Ein Ansatz auf höherer Ebene beschreiben [Zhao u. a., 2005, 20ff.] mit dem Begriff der „relativen“ Workflows, mit denen die Sichtbarkeit eigener Services kontrolliert werden kann. Hierbei können fertige, öffentliche Workflows durch jeden Partner bereits vordefiniert und mit verschiedenen „Sichtbarkeits-Attributen“ versehen werden. Je nach Berechtigung der Partner ergeben sich verschiedene Abstufungen:

- Mit „*invisible*“ gekennzeichnete Workflows sind für alle Partner versteckt und nur für den zentralen Prozesskoordinator verfügbar. (Dieser kann ggf. ein anderer Partner sein, falls diesem die Rolle des Prozesskoordinators zukommt.)
- Ist ein Workflow als „*trackable*“ gekennzeichnet, können die Partner Ausführungs- und Statusinformationen beziehen, selbst aber keine neuen Anfragen an die Services des Workflows stellen oder in dessen Verarbeitung eingreifen. (Der Prozesskoordinator kann diese Informationen jederzeit beziehen.)
- Mit „*contractable*“ gekennzeichnete Workflows sind vollständig öffentlich und können von allen Partnern verwendet werden. Dazu ist, wie der Name bereits sagt, gegebenenfalls ein individueller Vertrag zwischen den beiden Partnern notwendig, parallel zu oder anstelle von seiner Verwendung im Rahmen eines organisationsübergreifenden Workflows durch den Prozesskoordinator.

Jeder Partner ist dadurch in der Lage, seine eigenen, öffentlichen Services zu fertigen Prozessketten zusammenzusetzen, mit Ausführungseigenschaften zu versehen und damit in verschiedenen Szenarien seinen Partnern zur Verfügung zu stellen. Die Zugriffsbestimmungen auf fachlicher Ebene lassen sich

durch Berechtigung für ganze Workflows besser abbilden, als auf niedrigster, technischer Ebene in Form von Service-Verträgen<sup>4</sup>. Diese sind jedoch weiterhin notwendig und bilden das technische Grundgerüst. Die Ausführung der Prozessmodelle unterliegt den dafür entwickelten Systemen, auf deren Verwendung sich die Partner ebenfalls im Rahmen des Prozessmanagements einigen müssen. [Greiner, 2006, 24] nennt den ORACLE Process Manager, IBM WebSphere, Microsoft BizTalk Server und die ARIS Plattform als weit verbreitete Umgebungen. Der nächste Abschnitt beschäftigt sich mit den derzeit vorhandenen Sprachen und Ansätzen zur Prozessmodellierung.

### 3.4.6 Sprachen und Ansätze zur Prozessmodellierung

In den vergangenen Jahrzehnten wurden zahlreiche Sprachen, Methoden und Notationen zur Beschreibung von Prozessen entwickelt, jeweils im Hinblick auf ihre spätere Verwendung in einer bestimmten Branche, zu einem bestimmten Zweck oder unter einem bestimmten Aspekt. Jede der vorhandenen Sprachen und Methoden, von denen einige bereits in Tabelle 4 in Abschnitt 2.2.4 genannt wurden, unterstützt verschiedene Modellierungsaspekte unterschiedlich gut, beispielsweise die Abbildung von Aktivitäten, Verzweigungen, Rollen, Daten- und Informationsverarbeitung, Ressourcenverwendung und so weiter. Die vorhandenen Sprachen unterscheiden sich hinsichtlich des Grades, wie sehr sie bestimmte Aspekte berücksichtigen, von denen zahlreiche bereits in Abschnitt 2.2.4 genannt wurden.

Während stark formalisierte Prozessbeschreibungssprachen eine bessere Simulation, Automatisierung und spätere Implementierung unterstützen, lassen sich betriebswirtschaftliche Anforderungen teilweise nur schlecht abbilden, erfordern einen hohen Einarbeitungsaufwand oder sind aus fachlicher Sicht zur Prozessmodellierung ungeeignet. Dies gilt beispielsweise für Zustandsübergangsdiagramme, Abstract Status Machines oder pi-calculus Diagramme. Demgegenüber sind abstrakte, fachliche Sprachen und Notationen gegebenenfalls schlecht implementierbar und müssen erst in eine Implementierungssprache auf technischer Ebene überführt werden, wie es bei BPMN/ BPML zu BPEL der Fall ist. Das Hauptproblem besteht in den vielen verschiedenen Arten von Informationen, die innerhalb des Modells erfasst werden müssen, damit es für den praktischen Einsatz einen Nutzen stiftet, jedoch verliert das Modell immer mehr an Wert, je unübersichtlicher und komplexer es wird. Wenn die Geschäftsprozessmodelle als Grundlage zur Softwareentwicklung dienen, führen fehlerhafte oder unzureichend beschriebene Modelle zwangsläufig zu fehlerhafter Software, wenn wichtige Aspekte außer acht gelassen wurden, weil sie mit der vorhandenen Modellierungssprache nicht erfasst werden konnten [Korherr, 2008, 8f.].

Korherr hat mehrere Sprachen hinsichtlich ihrer Eignung zur Modellierung von Geschäftsprozessen untersucht und Anforderungen erkannt, die durch die bisher vorhandenen Sprachen nur unzureichend oder nicht abgebildet werden können. Oftmals unterstützen die Sprachen weder qualitäts- noch quantitätsbasierte Elemente, um die Häufigkeit des Eintritts von Aktionen und Aktivitäten, sowie die Sicherheit und Zuverlässigkeit auszudrücken, mit der eine Aktion oder Aktivität ausgeführt werden muss. Dies gilt gleichfalls für die Angabe von zeitlichen Restriktionen, wie der maximalen Ausführungsdauer einer Aktivität beziehungsweise eines Prozesses insgesamt. Dazu gehört weiterhin die Angabe der Priorität von Prozessschritten gegenüber anderen Prozessschritten des gleichen oder anderer Prozesse, um zu kennzeichnen welche Aktivitäten in jedem Fall ausgeführt werden müssen, oder welche im Fehlerfall, sowie bei Zeit- oder Geldmangel zunächst ausgelassen werden können [Korherr, 2008, 39].

Einige Sprachen besitzen kein offiziell standardisiertes Metamodell, wie zum Beispiel die Meta Object Facility für UML2. Dadurch sind die Hersteller von Modellierungstools gezwungen eigene Metamodelle zu entwickeln müssen, wodurch die Integration und der Austausch von Prozessmodellen über Unternehmensgrenzen hinweg erschwert werden. Zu diesem Zweck müssten sich die beteiligten Part-

---

<sup>4</sup> Weiterführende Literatur zum organisationsübergreifenden Workflow-Management siehe: [Schulz, K. and Orłowska, M.: Facilitating Cross-organisational Workflows with a Workflow View Approach. Data & Knowledge Engineering, 2004] und [Wetzel, I. and Klischewski, R.: Serviceflow beyond Workflow? IT Support for Managing Inter-organizational Service Processes. Information Systems, 2004]

ner auf ein gemeinsames Metamodell einigen, wodurch ein Mapping der Prozessmodellelemente oder Metamodelle notwendig wird [Korherr, 2008, 40].

Ein weiteres Problem besteht darin, oftmals die Prozesskosten nur unzureichend im Modell abbilden zu können, wodurch ihre Erfassung für die einzelnen Aktivitäten erschwert wird. Die Frage, wann wer was zu tun hat, lässt sich in fast allen Geschäftsprozessmodellierungssprachen abbilden, jedoch nicht, welche Kosten für diese Person oder diesen Service während des Prozesses anfallen. Für das Prozesscontrolling würden sich völlig neue Möglichkeiten ergeben, wenn die entstehenden Kostenarten, wie Ressourcenverbrauch und Zeitdauer implizit für jede Aktivität festgelegt und während der Prozessausführung bereits aufgrund des Modells erfasst werden könnten. Insgesamt, so bemerkt Korherr, können die vorhandenen Prozessmodellierungssprachen zwar gut abbilden, *wie* ein Prozess abläuft (dafür stehen zahlreiche Prozessbausteine- und Konstrukte zur Verfügung), jedoch nicht unter *welchen betriebswirtschaftlichen Aspekten* hinsichtlich Zeit, Kosten, Effektivität, Effizienz, Qualität oder Zuverlässigkeit dies geschehen kann oder sollte.

Anders ausgedrückt mangelt es den bisherigen Geschäftsprozessmodellierungssprachen an der Möglichkeit, die betriebswirtschaftlich relevanten Prozessziele und deren Messung zu modellieren, wodurch das Metamodell der vorhandenen Sprachen erweitert werden muss, damit die Prozesskennzahlen während der Prozessausführung erfasst werden können. Diese Erweiterung geschieht ebenfalls herstellerspezifisch und erschwert die Übertragung von Prozessmodellen.

Im Folgenden werden verschiedene Ansätze und Sprachen zur Beschreibung von Geschäftsprozessen vorgestellt, bevor im nächsten Kapitel das RosettaNet Framework genauer erklärt wird. Einen hohen Bekanntheitsgrad im Bereich der Geschäftsprozessmodellierung genießt die Business Process Modeling Notation (BPMN), die im Jahr 2002 von Stephen White, einem IBM-Mitarbeiter erarbeitet und anschließend durch die Business Process Management Initiative (BPMI) weiterentwickelt wurde. 2005 wurde die Entwicklung der Notation an die Object Management Group (OMG) übergeben und zählt seit 2006 als offizieller OMG-Standard mit der aktuellen Versionsnummer 1.1 [OMG, 2008]. Zahlreiche Kontrollflusselemente weisen durch den Einfluss der OMG eine hohe Ähnlichkeit mit UML2 Aktivitätsdiagrammen auf [Korherr, 2008, 21].

Ein auf Grundlage von BPMN erstelltes Diagramm wird als „Business Process Diagram“ (BPD) bezeichnet. Aufgrund seines ursprünglichen Entwurfes besitzt BPMN jedoch kein festes Metamodell, wodurch der Austausch zwischen Prozessmodellierungswerkzeugen erschwert wird. Für die neue Version 2.0 ist daher eine einheitliche XML-Syntax zur Standardisierung der Metadaten vorgesehen [OMG, 2007]. Während BPMN eine grafische Notation für Geschäftsprozessmodelle darstellt, bildet die Business Process Modeling Language (BPML) eine XML-basierte Metasprache zur Beschreibung von Geschäftsprozessen. BPML wurde ebenfalls von der BPMI entwickelt, später in die OMG aufgenommen und liegt in der aktuellen Version 1.0 von November 2002 vor. Die Weiterentwicklung von BPML und BPMN unterliegt der „Business Modeling & Integration Domain Task Force“ (BMI DTF) als Teilbereich der OMG [OMG, 2008].

Bei BPMN werden sieben grundlegende Arten von Notationselementen unterschieden. *Gateways* stellen Verzweigungen des Prozessablaufs dar und dienen der Modellierung des Prozessablaufes. Ein weiteres Notationselement sind *Ereignisse*, beispielsweise für den Versand oder das Eintreffen von Nachrichten, bei bestimmten Zeitpunkten, Fehlern, Signalmeldungen oder Bedingungen. *Rollen* werden mithilfe von *Pools* und *Swimlanes* dargestellt, um Organisationen, Verantwortlichkeiten, Abteilungen, IT-Systeme oder Stelle abzubilden. Die Darstellung von *Aktivitäten* erfolgt durch die Verwendung von *Tasks*, die Schleifen, mehrfache Instanzen und Unterprozesse enthalten können. Weitere Notationselemente sind Datenobjekte und Transaktionen, auf die hier nicht näher eingegangen wird. Eine Übersicht der wichtigsten Notationselemente befindet sich im Anhang B, die von der Business Process Technology Group erarbeitet wurde.

Mithilfe von BPMN erstellte Geschäftsprozessdiagramme sind aufgrund ihrer Struktur nicht als maschinell lesbare Prozessbeschreibung geeignet und müssen zu diesem Zweck in eine Ausführungssprache übersetzt werden. Für BPMN sind die Business Process Execution Language (BPEL) und die XML Process Definition Language (XPDL) sehr gut geeignet [Korherr, 2008, 58f.]. Der BPMN-Standard liefert eine Reihe von Richtlinien für die Übertragung der in BPMN modellierten Prozessbe-

standteile in den BPEL-Standard, die von der OMG erarbeitet wurden. Im Rahmen der modellgetriebenen Softwareentwicklung bezeichnet dies die Übertragung des plattformunabhängigen BPMN-Modells (PIM) auf ein plattformspezifisches BPEL-Modell (PSM) [Korherr, 2008, 75]. Für XPD L hat die Workflow Management Coalition ein Übertragungsschema erstellt. Die Übertragung von BPMN auf ebXML ist geplant, aber noch nicht ausgeführt.

BPEL ist eine XML-Sprache zur Spezifikation des Verhaltens von Geschäftsprozessen auf Basis von WebServices, die im April 2007 in der Version 2.0 zuletzt veröffentlicht wurde [OASIS, 2007a]. Sie unterstützt die Beschreibung von abstrakten und ausführbaren Geschäftsprozessen, sowie deren Orchestrierung mittels einer Workflow-Engine. Die Instanzierung von BPEL-Prozessen in einer Java-Umgebung ist mittels BPEL4J möglich. Nach [Korherr, 2008, 103] können BPEL-Prozesse auch mithilfe von UML2 Aktivitätsdiagrammen modelliert werden, wobei jedoch die Object Constraint Language (OCL) zur Einbindung der betriebswirtschaftlichen Aspekte in das Aktivitätsdiagramm notwendig ist, aber bisher kaum unterstützt wird.

Neben den Prozessbeschreibungs- und -ausführungssprachen BPEL und BPMN/BPML liefert ebXML eine Reihe von Vorgaben für die Gestaltung von XML-basierten Dokumenten für elektronische Geschäftsprozesse. Gegenüber den vorher genannten Sprachen und Notationen stellt ebXML keinen geschlossenen Standard, sondern eine Sammlung von Standards der UN/CEFACT und der OASIS dar, die ebXML entwickelten [EBXMLa, 2006]. Dazu gehören Spezifikationen für Gestaltung von Geschäftsprozessen („Business Process Specification Schema“, BPSS), ein Informationsmodell für die Speicherung der Geschäftsprozesse („Registry Information Model“), ein Informationsmodell zur Speicherung der in den Geschäftsprozessen eingebundenen Services („Registry Services Specification“), Hinweise zur Gestaltung einer technischen Infrastruktur (ebXML Technical Architecture Specification“), sowie Vorgaben für den Entwurf von Protokollen für den Nachrichtenaustausch zwischen den Geschäftspartnern („Collaboration-Protocol Profile and Agreement Specification“, CPP- und CPA-Spezifikation).

EbXML bildet damit keine Sprache oder Notation zur Modellierung oder Ausführung von Geschäftsprozessen, sondern ein Framework für die XML-basierte Kopplung der Anwendungssysteme von Geschäftspartnern unter Verwendung von Metamodellen und Implementierungsvorgaben für Interaktionsprotokolle, Verzeichnisse, Nachrichtenformate und öffentliche Services [EBXMLb, 2006]. Die nachfolgende Abbildung stellt das Design-Time und Run-Time Referenzmodell für ebXML dar, ist jedoch auch auf andere, ähnliche Ansätze übertragbar.

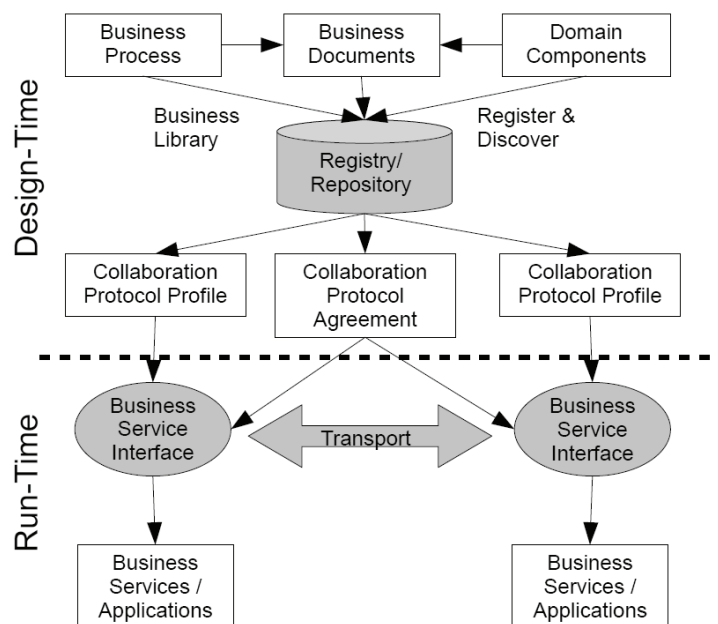


Abbildung 12: ebXML Design-Time und Run-Time Referenzmodell in Anlehnung an [EBXML, 2001, 13]



Die während der Entwurfsphase festgelegten Geschäftsprozesse und die durch sie erzeugten und veränderten Geschäftsdokumente werden in einem zentralen Verzeichnis erfasst. Die Strukturierung der Dokumente und ihre enthaltenen Daten können zuvor als Kernkomponenten und Domainkomponenten modelliert und bei der Erstellung von Geschäftsprozessen wieder verwendet werden. Im Rahmen von RosettaNet entspricht dies dem Business Dictionary und dem Technical Dictionary, worauf im Kapitel 3.5.2 genauer eingegangen wird. Aus den im Repository hinterlegten Daten kann das Collaboration Protocol Profile abgeleitet werden, das als wesentlicher Input zur Erstellung der Serviceschnittstellen dient. Zur Modellierung der Geschäftsprozesse wird bei ebXML eine Untermenge der im UML-Standard vorhandenen Modellierungselemente und Modellarten verwendet, beispielsweise UML2 Zustandsdiagramme oder Aktivitätsdiagramme [OASIS, 2007b]. Auf Grundlage von ebXML entwickelte OASIS die *Universal Business Language*.

*“The Universal Business Language is designed to provide a universally understood and recognized commercial syntax for legally binding business documents and to operate within a standard business framework such as ISO 15000 (ebXML) to provide a complete, standards-based infrastructure that can extend the benefits of existing EDI systems to businesses of all sizes.” [OASIS, 2009]*

Die UBL konzentriert sich dabei auf die Standardisierung von Geschäftsdaten als den ersten Schritt zur globalen E-Commerce Integration, unter Verwendung der in ebXML enthaltenen Core Components. Dabei wird strikt zwischen der Standardisierung von Geschäftsdaten/-dokumenten und Geschäftsprozessen getrennt, indem beide auf verschiedenen Standardisierungsschichten verteilt werden. Im Anhang D ist ein Teil des mit UBL standardisierten Geschäftsprozesses für den Vertrieb mithilfe eines UML2 Aktivitätsdiagramms dargestellt. Nachdem in diesem Kapitel auf einige Ansätze zur Modellierung von Geschäftsprozessen eingegangen wurde, erfolgt im nächsten Kapitel eine ausführliche Beschreibung des RosettaNet Frameworks, mit dessen Hilfe organisationsübergreifende Prozesse implementiert werden können.

### 3.5 Organisationsübergreifende Prozesse am Beispiel von RosettaNet

#### 3.5.1 Entstehung und Ziele des RosettaNet Konsortiums

RosettaNet ist ein globales, nichtkommerzielles Konsortium von mehr als 500 Unternehmen verschiedener Branchen, insbesondere der IT, Computer- und Halbleiterindustrie, Telekommunikation und Logistik. Es wurde im Februar 1998 von 40 Unternehmen gegründet mit dem Ziel, die elektronische Datenübertragung zwischen Unternehmen zu standardisieren. Der Schwerpunkt liegt hierbei in der Entwicklung und Verbreitung standardisierter Schnittstellen für den Bereich des Electronic Commerce beziehungsweise Collaborative Commerce zur Unterstützung von Geschäftsprozessen über Unternehmensgrenzen hinweg. Heutzutage bildet RosettaNet eine Unterorganisation des GS1, welches aus dem Uniform Code Council (UCC) hervorgegangen ist und eine der einflussreichsten Standardisierungsorganisationen darstellt [Alonso u. a., 2004, 234].

RosettaNet wurde mit dem Ziel entwickelt, zahlreiche Probleme des elektronischen Datenaustauschs (EDI) durch dessen stärkere Standardisierung zu beheben. Dazu zählten die hohen Kosten für kleine und mittelständige Unternehmen beim Einsatz von EDI und der daraus entstandene Barriere zur Teilnahme am Electronic Commerce, die hohe Komplexität zur Entwicklung von EDI-Anwendungen und den hohen Wartungs- und Administrationsaufwand der damaligen EDI-Systeme [ORACLE, 2008a].

Durch die Standardisierung des elektronischen Datenaustauschs wird die Automatisierung von Geschäftsprozessen unterstützt. Diese Vorteile ergeben sich aus der Möglichkeit, flexible und dynamische Wertschöpfungsketten zwischen den Partnern zu etablieren, in dem sie das Internet zum Austausch von Geschäftsdaten und Dokumenten nutzen und dabei auf die durch RosettaNet definierten Standards zurückgreifen [RosettaNet, 2008]. RosettaNet verfolgt damit das Ziel, eine weltweit einheitliche Sprache für das Electronic Business zu verbreiten.

*„RosettaNet aims to align the business processes of supply chain partners. This goal is achieved by the creation of RosettaNet Partner Interface Processes or PIPs and it helps to define business processes between trading partners. Each PIP defines how two specific processes,*

*running in two different partners' organizations, will be standardized and interfaced across the entire supply chain. PIP includes all business logic, message flow, and message contents to enable alignment of the two processes.*“ [RosettaNet, 2009]

### 3.5.2 Standardisierungsbereiche

RosettaNet spezifiziert eine Reihe von Standards in verschiedenen Bereichen, die zur Entwicklung von Geschäftsbeziehungen verwendet werden können. Dazu gehören Richtlinien für plattformübergreifende Anwendungen und deren Netzwerkkommunikation, basierend auf offenen XML-Standards. Die Standardisierung findet dabei in drei Bereichen statt. Die weiteren Angaben wurden [Alonso u. a., 2004, 234f.] entnommen.

Im Bereich der *Geschäftsprozesse* umfasst RosettaNet die Spezifikation von Koordinationsprotokollen für die beteiligten Partner. Zu diesem Zweck werden Partner-Interface-Processes (PIPs) verwendet, die sowohl horizontale als auch vertikale Protokolle definieren, die von den Partnern ausgeführt werden. Jeder PIP umfasst dabei die zu übertragenden Geschäftsdokumente, ein zugehöriges Vokabular und die Choreografie des Nachrichtenaustauschs. Alle PIPs werden innerhalb des PIP Directory gesammelt und in einem PIP Specification Package gespeichert.

Bei den *Datenformaten* dient RosettaNet zur Bildung einer einheitlichen Terminologie für die Bestandteile des Nachrichtenaustauschs. Dadurch wird sichergestellt, dass eine Anfrage vom Empfänger auf die gleiche Weise verstanden wird, wie es vom Absender vorgesehen war. RosettaNet dient damit der Überbrückung von technischen und semantischen Sprachbarrieren durch die Entwicklung zweier Wörterbücher. Das Business Dictionary befasst sich mit der Beschreibung von Geschäftsaktivitäten, das Technical Dictionary bietet eine einheitliche Terminologie, die zur Definition von Produkten und Services verwendet wird.

Den dritten Bereich bilden die *Nachrichtendienste* (messaging services). Wenn ein PIP vollständig spezifiziert wurde, gibt das RosettaNet Implementation Framework (RNIF) vor, wie dieser Prozess ausgeführt wird. Es übernimmt dabei die Ver- und Entpackung der Nachrichten, sowie deren Transfer, Routing, Verschlüsselung und Authentifizierung während der Kommunikation zwischen den Partnern. Das RNIF definiert zu diesem Zweck eine Container-Struktur, das Nachrichten-Header-Format, die Protokollbindung und Choreografie-Muster (synchron, asynchron) für den Austausch der Nachrichten.

### 3.5.3 Sichten und Cluster des RosettaNet Frameworks

Zur Beherrschung der Komplexität innerhalb des RosettaNet-Frameworks wurden verschiedene Sichten und Cluster entwickelt, mit deren Hilfe die überbetrieblichen Beziehungen organisiert und entwickelt werden können. RosettaNet unterscheidet in diesem Zusammenhang drei verschiedene Sichten (Views) auf unterschiedlichem Abstraktionsgrad, entsprechend einer fachlichen, funktionalen und technischen Betrachtung der Geschäftsbeziehungen. [Alonso u. a., 2004, 234ff.] beschreiben die Sichten folgendermaßen:

Der *Business Operational View* (BOV) wird auch als „Aktionsschicht“ bezeichnet und beschreibt den Fluss der Geschäftstransaktionen zwischen den beteiligten Geschäftseinheiten. Sie wird durch Geschäftsanalysten erarbeitet und enthält alle Rollen, die an den Transaktionen beteiligt sind, zum Beispiel Menschen und Anwendungen und deren Einfluss auf die Wertschöpfungskette. Innerhalb des BOV werden Blueprints für die PIPs entworfen, die mithilfe von UML Aktivitätsdiagrammen notiert werden. Das Ergebnis bilden ein oder mehrere „PIP Business Process Flow Diagrams“, die rollenbasierte Diagramme darstellen, welche den Nachrichtenaustausch und die Protokollschnittstellen beschreiben. Jede Rolle besitzt ihre eigene Bahn (Swimlane), wodurch klar ersichtlich ist, *welcher* Partner *wann welche* Aktion ausführt. Jedes Diagramm hat einen Startpunkt, der den Beginn des Geschäftsprozesses kennzeichnet und einen oder mehrere Endpunkte, je nach Ergebnis des Prozesses. Rechtecke mit abgerundeten Ecken repräsentieren die zu verarbeitenden Geschäftsdokumente. Zur Darstellung von Aktivitäten werden Rechtecke mit abgerundeten Seiten verwendet, von denen jede einen eindeutigen Namen hat.

Der *Functional Service View* (FSV) wird auch als „Transaktionsschicht“ bezeichnet und direkt aus dem BOV abgeleitet. Er spezifiziert die Geschäftstransaktionen zwischen den Geschäftseinheiten im Sinne des Nachrichtenaustauschs auf technischer Ebene. Der FSV legt damit die konkreten Koordinationsprotokolle und Kontrollinformationen für den Nachrichtenaustausch fest, das heißt die Aktionen und Signale während einer Konversation innerhalb des PIP Business Process Flows. Die innerhalb des FSV interagierenden Netzwerkkomponenten werden auch als „RosettaNet Services“ bezeichnet.

Der *Implementation Framework View* (IFV) wird auch „Serviceschicht“ genannt und definiert die Anforderungen an die Kommunikationsprotokolle und Nachrichtenformate. Beispielsweise kann festgelegt werden, dass der sichere Transport über HTTPS geschieht und eine bestimmte XML-DTD für den Nachrichtenaustausch verwendet wird. Zukünftig ist vorgesehen den W3C-XML-Standard zur Spezifikation der Struktur von Aktionen und Signalen zu verwenden, um die Kompatibilität mit Web-Services zu verbessern [ORACLE, 2008a].

Neben den drei beschriebenen Sichten werden alle standardisierten Geschäftsbereiche in acht verschiedene Cluster unterteilt, die jeweils mehrere Segmente beinhalten. Die folgende Tabelle gibt dazu einen Überblick. Im Anhang C befindet sich diese Übersicht zu den Clusters and Segments in englischer Sprache.

## Entwicklung organisationsübergreifender Prozesse und Services

Cluster	Segment	Beschreibung
0: RosettaNet Support	0A: Administration 0C: Tests und Support	PIPs für administrative Funktionalitäten (z. B. Fehlerbehandlung)
1: Partner Produkt & Service Review	1A: Partnerbesprechung 1B: Produkt & Service Besprechung	PIPs zur Erfassung und Sammlung von Produktinformationen, Management des Aufbaus von Produktbeschreibungen
2: Produktinformationen	2A: Vertriebsvorbereitung 2B: Benachrichtigung bei Produktwechsel 2C: Produktentwurfinformationen 2D: Kollaborativer Entwurf und Entwicklung	PIPs zur Verteilung und Aktualisierung von Produktbeschreibungen
3: Bestellungsverwaltung	3A: Auftragserfassung 3B: Transport und Vertrieb 3C: Bezahlung und Finanzen 3D: Produktkonfiguration	PIPs zur Bestellungsverwaltung, Bestellungsabwicklung und Statuskontrolle
4: Bestandsverwaltung	4A: Gemeinschaftliche Verkaufsprognose 4B: Bestandseinteilung (Planung) 4C: Bestandsbericht 4D: Bestandsauffüllung 4E: Verkaufsbericht 4F: Preisvereinbarung	PIPs Abfrage und Änderung von Inventarinformationen zur Lagerbestandsverwaltung- und planung
5: Marketing- Informationsmanagement	5A: Lead Opportunity Management 5B: Marketingkampagnen-Management	PIPs zur Unterstützung des Marketing
6: Reparatur und Unterstützung	6A: Unterstützung und Administration von Garantien, Services und Verträgen 6B: Unterstützung und Administration der Anlagenverwaltung 6C: Technischer Support und Service-Management	PIPs für den technischen Support im Schadensfall
7: Herstellung (Produktion)	7A: Übertragung von Entwürfen 7B: Fabrikationsmanagement 7C: Verteilung von Herstellungs- Herstellungsinformationen	PIPs zur Unterstützung von Supply-Chains bei Produktionsunternehmen

Tabelle 8: RosettaNet Clusters und Segmente

Übersetzung von [Microsoft, 2008] und ergänzt durch [Alonso u. a., 2004, 235f.]

Jeder PIP besitzt durch die Verwendung dieser Kategorisierungshierarchie eine eindeutige Kennung, die sich aus der Nummer des Clusters und des Segmentes zusammensetzt, beispielsweise für den PIP „3A2: Preisanfrage und Verfügbarkeit“ im Cluster „3: Bestellungsverwaltung“ und „A: Auftragserfassung“. Die Spezifikation von PIPs wird im nächsten Abschnitt genauer betrachtet. Folgendes Beispiel verdeutlicht den Zusammenhang:

<pre> CLUSTER 3: Order Management   Segment A: Quote and Order Entry     PIP 3A1: Request Quote       Activity: Request Quote         Action: Quote Request Action         Action: Quote Confirmation CLUSTER 4: Inventory Management         </pre>
--

Abbildung 13: RosettaNet Clusters, Segments, PIPs, Activities und Actions  
in Anlehnung an [ORACLE, 2008a]

### 3.5.4 Partner-Interface-Process Spezifikationen

Ein Partner-Interface-Process beschreibt, wie ein gemeinschaftliches Kommunikationsprotokoll gestaltet werden kann. Jeder PIP enthält zu diesem Zweck ein Technical Dictionary, in dem alle am Prozess beteiligten Komponenten beschrieben sind, und das Business Dictionary, welches als Geschäftsspezifikation des Protokolls dient. Es werden zwei Arten von Nachrichten beim Austausch von Geschäftsdokumenten innerhalb eines PIP unterschieden. Die Geschäftsaktionsnachricht (business action message, BAM) beinhaltet Geschäftsdokumente (Geschäftsdaten) wie zum Beispiel Rechnungen oder

Bestellungen. Die Geschäftssignalnachricht (business signal message, BSM) dient der Bekanntgabe, ob eine Geschäftsaktionsnachricht positiv oder negativ ist, sie überträgt jedoch keine Geschäftsdaten [Alonso u. a., 2004, 235]. Nach [Khalaf, 2005, 364] wird dieser Nachrichtenaustausch mithilfe von Message Sequence Charts (MSC) abgebildet.

Die PIPs stellen somit das Ergebnis intensiver Forschung dar, wie Geschäftsprozesse innerhalb einer Wertschöpfungskette auf verschiedenen Ebenen spezifiziert werden können. Sie bilden eine Sammlung generischer, standardisierter Prozesse, die als Grundlage für Prozesse in der realen Welt dienen können, um das Business-to-Business-Alignment zu verbessern. PIPs kapseln zu diesem Zweck die Bestandteile von Geschäftsprozessen durch die Spezifikation der Struktur und des Formates von Geschäftsdokumenten und deren Zusammenhang mit Aktivitäten, Aktionen und Rollen für jeden Partner [ORACLE, 2008a].

*“In simple terms **Partner-Interface-Processes** can be defined as the choreography and message content exchanged with trading partners. We should remember that PIPs are a specification and not an implementation; this gives the trading partners that adopt RosettaNet the flexibility to implement the PIP specifications themselves or to purchase third-party products that reduce the development overhead.”* [ORACLE, 2008a]

Die Spezifikation eines PIP geschieht in vier Schritten, beginnend mit der abstrakten, fachlichen Sicht, die anschließend bis zur technischen Sicht verfeinert und konkretisiert wird. (Folgendes in Anlehnung an [Alonso u. a., 2004, 235f.]). Im *ersten Schritt* erfolgt die Beschreibung des aktuellen Geschäftsmodells in dem erfasst wird, wie die beteiligten Partner (supply chain business entities) ihre Geschäfte zum jetzigen Zeitpunkt abwickeln. Dies bewirkt die Erfassung des Istzustandes und entspricht einer Analyse der bisherigen Geschäftsprozesse. Der *zweite Schritt* dient dem Reengineering des aktuellen Geschäftsprozessmodells mit dem Ziel, eine Spezifikation zu entwerfen, mit deren Hilfe die Partner internetbasierte Interaktionen zur Abwicklung ihrer Geschäfte verwenden können. In diesem Schritt wird die Frage beantwortet, welche Geschäftsaktions- und Signalnachrichten die Partner miteinander austauschen müssen, um einen unternehmensübergreifenden Geschäftsprozess etablieren zu können.

Sobald das Reengineering abgeschlossen ist, erfolgt im *dritten Schritt* die Erstellung eines PIP Blueprints, in dem beschrieben ist, welche Aufgaben die Partner innerhalb des organisationsübergreifenden Geschäftsprozesses übernehmen. Erst wenn alle Partner diesem Blueprint zustimmen, wird mit dem vierten Schritte begonnen. Dieser PIP Blueprint entspricht dem in Abschnitt 3.4.5 beschriebenen, komplexen Kooperationsmodell mit einem zentralen, global entwickelten organisationsübergreifenden Geschäftsprozessmodell.

Im letzten, *vierten Schritt* erfolgt die Erstellung eines PIP-Protokolls, welches den Systemarchitekten und Softwareentwicklern vorgibt, über welche Schnittstellen die ans Unternehmensnetzwerk angeschlossenen Anwendungen interagieren müssen, das heißt, in welcher Art und Weise das PIP-Protokoll implementiert wird.

Dabei müssen zahlreiche Fragen zur konkreten Implementierung der PIPs besprochen werden [ORACLE, 2008b]:

- Welche Informationen sind vom Partner für die Implementierung notwendig?
- Welche Version der vorhandenen PIPs werden implementiert?
- Welche RNIF-Version (1.1 oder 2.0) wird für den Nachrichtenaustausch verwendet?
- Wie gestaltet sich die Back-End-Integration für jede PIP? Das heißt: welche privaten Prozesse sollen mit den PIPs gekoppelt werden?
- Welche Datenformate und Kommunikationsmittel unterstützen die Back-End-Systeme?

Die beschriebene Vorgehensweise zur Entwicklung der PIPs entspricht dem Top-Down-Ansatz der Entwicklung einer verteilten, aktivitätsorientierten Ausführung, wie sie im Abschnitt 3.4.5 erläutert wurde. Die Ausführung geschieht mit Hilfe des RosettaNet Implementation Frameworks, worauf im nächsten Abschnitt genauer eingegangen wird.

### 3.5.5 Das RosettaNet Implementation Framework

Das RosettaNet Implementation Framework (RNIF) wurde zur Unterstützung der Entwickler von E-Business Lösungen entworfen, die Anwendungen bzw. Anwendungskomponenten implementieren, die PIPs ausführen sollen. Es dient somit als eine Art „Middleware-Spezifikation“ zur Implementierung von PIPs. Durch ihre Einigung auf die RNIF-Spezifikation können die Partner sicherstellen, dass ihre Anwendungen fehlerfrei mit den Anwendungen der Partner kommunizieren. RNIF definiert zu diesem Zweck die Verpackung von PIPs, die Authentifizierung, Autorisierung, Verschlüsselung und Zuverlässigkeit (exactly-once-Semantik).

*“The **RosettaNet Implementation Framework (RNIF)** standard defines how systems transport a RosettaNet message. The RNIF standard is a robust transfer, routing, packaging, and security standard. All RosettaNet messaging systems must comply with the RNIF standard to achieve RosettaNet certification.”* [Microsoft, 2008]

*„The **RNIF** is an open, common networked-application framework designed to allow RosettaNet supply chain and solution partners to collaborate in executing RosettaNet PIPs.“* [ORACLE, 2008b]

RosettaNet Business Messages stellen demnach standardisierte, transportprotokollunabhängige Spezifikationen zur Verpackung von Bestandteilen der PIPs (Aktivitäten, Aktionen, Geschäftsdaten) dar. Der Nachrichtenaustausch kann durch die Verwendung von S-MIME, HTTPS und SSL sicher gestaltet werden [ORACLE, 2008a]. Die folgende Abbildung enthält das RNIF-Nachrichtenformat als Verpackung der Aufrufe innerhalb des PIP (Process-, Activity-, Action-Control) und dessen Zusammenhang mit den tiefer liegenden Schichten des Netzwerk-Referenzmodells.

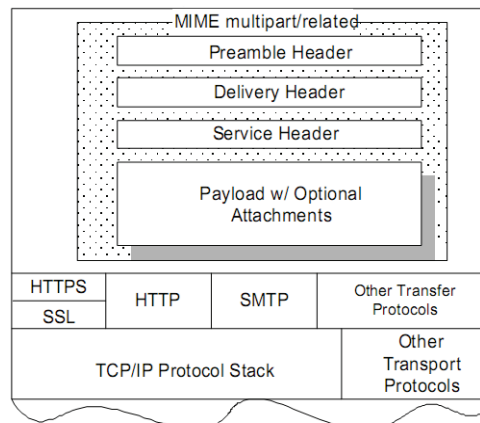


Abbildung 14: RosettaNet Application Protocol Stack View [Yendluri, 2000]

Die Entwicklung eigenständiger Sicherheitservices, wie im Abschnitt 3.3 beschrieben wurde, entfällt in diesem Fall, jedoch müssen weiterhin die Rechte und Pflichten der Partner festgelegt werden. Im Anhang A sind viele der beschriebenen Bestandteile des RosettaNet-Frameworks im Zusammenhang mit öffentlichen und privaten Prozessen abgebildet. Eine Übersicht zu den grundlegenden Bestandteilen einer RosettaNet-Architektur zwischen einem Unternehmen (B) und einem Kunden kann der folgenden Abbildung entnommen werden.

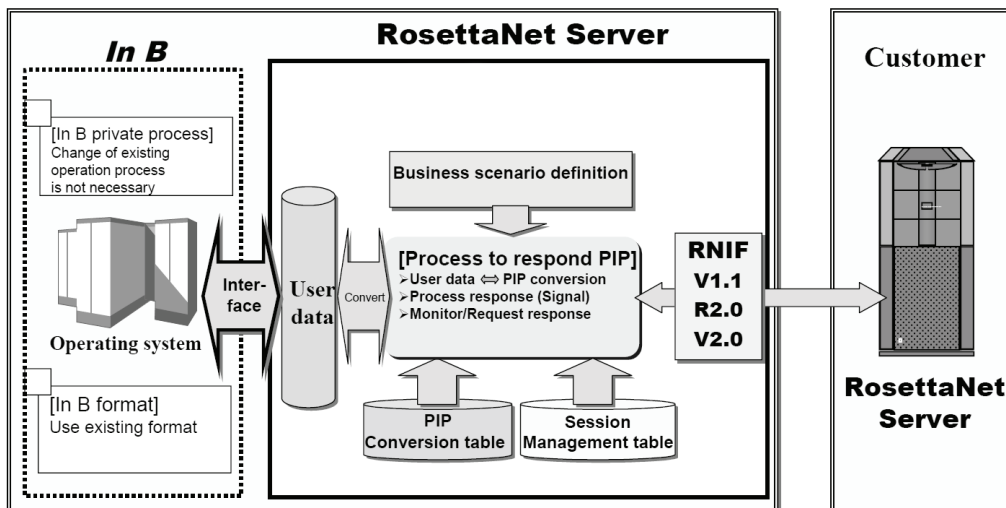


Abbildung 15: RosettaNet B2C-Server-Architektur  
[Fujitsu, 2002]

Der Einsatz des RNIF entspricht, in Bezug auf die Möglichkeiten zur Ausführung organisationsübergreifender Prozesse, der *dezentralen, aktivitätsorientierten* Ausführung auf Basis eines *zentralen* Geschäftsprozessmodells, dem PIP Blueprint, dem alle Partner zustimmen müssen, um die Interaktionsfähigkeit ihrer Anwendungen zu gewährleisten, worauf im Abschnitt 3.3.5 eingegangen wurde.

### 3.5.6 RosettaNet im Rahmen des Geschäftsprozessmanagements

RosettaNet stellt einen Ordnungsrahmen zur Entwicklung organisationsübergreifender Prozesse dar, der aus einer Vielzahl von Standards auf mehreren Abstraktionsebenen besteht, mit deren Hilfe einheitliche Geschäftsvorfälle abgebildet werden können. Im Kapitel 3.4 wurden öffentliche und private Prozesse und deren Zusammenspiel bereits erläutert. Bei RosettaNet wird diese Trennung ebenfalls aufrecht erhalten:

*„RosettaNet business processes follow the RosettaNet design strategy of separating public and private business processes. Public business processes provide for the exchange of business messages between trading partners, while private business processes interact with internal, back-end systems. Public business processes have standardized, highly structured PIP choreographies, while private business processes are highly customized to a trading partner’s internal environment. Private processes communicate with public processes via well-defined interfaces, typically based on JMS queues.“* [ORACLE, 2008b]

Yi u. a. haben sich mit den Problemen bei der Verwendung von RosettaNet im Rahmen komplexer, organisationsübergreifender Geschäftsprozesse zwischen mehreren Partner beschäftigt. Mit der Anzahl der Geschäftspartner und der Geschäftsprozesse zu diesen Geschäftspartnern, steigt zwangsläufig die Menge der benötigten PIPs. Je mehr PIPs ein Unternehmen managen muss, je sinnvoller ist deren Integration in einem zentralen System für deren Verwaltung. Dieser Zusammenhang wird als „Multi-PIP-Support“ bezeichnet.

Das RosettaNet-Framework liefert selbst leider keine Hinweise oder Richtlinien, wie ein solches, zentrales System gestaltet werden kann. RosettaNet gibt nur vor, welche Daten und Dokumente innerhalb eines PIPs verwendet werden sollten, jedoch nicht, wie mehrere PIPs zu einem komplexen Geschäftsprozess auf höherer Ebene zusammengesetzt werden kann. Das Problem besteht in der verhältnismäßig niedrigen Abstraktionsebene, in der die PIPs bei den Partnern ausgeführt werden, wobei jeder PIP eine einzeln stehende, bidirektionale Verbindung dieser Partner darstellt. Deren Komposition zu einem komplexen Geschäftsprozess über mehrere Partner ist schwierig und kompliziert, weil dies beim Entwurf des RosettaNet Implementation Frameworks nur unzureichend beachtet wurde [Yi u. a., 2006, 698]. Durch die „Abschottung“ der PIPs untereinander bietet das RosettaNet Framework nur

mangelhafte Möglichkeiten zum Monitoring und der Kontrolle der PIPs, wenn diese als Teil eines umfangreicheren Prozesses über mehrere Partner eingesetzt werden sollen.

Eine Lösung dieses Problems wäre die Verwendung eines zentralen oder mehrerer dezentraler Workflow-Managementsysteme, die einen organisationsübergreifenden Geschäftsprozess auf hoher, fachlicher Ebene ausführen, bei dem einzelne PIPs als Prozessbestandteile auf technischer Ebene ausgeführt werden. [Yi u. a., 2006, 700] empfehlen zu diesem Zweck ein kontentbasiertes Dokumentenrouting zur Verwaltung der Beziehungen sowohl zwischen einzelnen PIPs, als auch zwischen PIPs und anderen BPM-Instanzen. Theoretisch könnte die PIP-Prozesskette an einer Stelle unterbrochen sein und die Partner dort mithilfe anderer Technologien kommunizieren, beispielsweise dem Nachrichtenaustausch über WebServices. Mit den bisherigen RosettaNet-B2B-Systemen lässt sich dies noch nicht realisieren [Yi u. a., 2006, 701].

Ein grundlegendes Problem bei der Verwendung von PIPs im Rahmen komplexer Geschäftsprozesse über mehrere Partner liegt in der Identifikation der einzelnen PIP-Instanzen. RosettaNet verwendet zur Identifikation der PIPs eine „PIP-Instance-ID“, wobei sich diese ausschließlich auf die Instanz zu einem festen Partner im Rahmen einer „Zwei-Parteien-Kommunikation“ bezieht, bei der der Partner nicht zu identifiziert werden braucht, sondern lediglich die aktuelle Prozessinstanz zur Verarbeitung der aktuellen Nachrichten. Bei einer Kommunikation über mehrere Partner reicht die Identifikation der Prozessinstanz nicht aus, da der Partner selbst ebenfalls identifiziert werden muss. Anders gesagt, sieht der RosettaNet-Standard die Unterscheidung von Nachrichten, mehrerer, verschiedener Partnern als Teilnehmer des gleichen Prozesses nicht vor [Yi, 2006, 702]. Im nächsten Kapitel wird auf einen möglichen Lösungsansatz für dieses Problem eingegangen.

### **3.5.7 RosettaNet-basierte WebServices mit WSDL und BPEL4WS**

Die Verwendung von RosettaNet in komplexen Geschäftsprozessen ist mit mehreren verschiedenen Partnern und unter Verwendung unterschiedlicher PIPs möglich, wenn RosettaNet-basierte WebServices implementiert werden, die mithilfe von BPEL4WS die Erstellung umfangreicher Interaktionsszenarien unterstützen.

Um die Kombination aus RosettaNet-PIPs und WebServices nachvollziehen zu können, ist es notwendig, sich die Ziele beider Ansätze vor Augen zu halten und miteinander zu vergleichen. Zur Entwicklung organisationsübergreifender Geschäftsprozesse sind mehrere Komponenten erforderlich, die als Vergleichskriterien aufgefasst werden können. Dazu gehören *öffentliche Schnittstellen*, über die ein oder mehrere, andere Unternehmen auf die angebotenen Dienste des eigenen Unternehmens zugreifen können. Dabei ist es empfehlenswert standardisierte Schnittstellen- und Protokollformate zu verwenden. Als nächstes ist die Verwendung eines *einheitlichen Vokabulars* notwendig, in dem festgelegt ist, wie die Nachrichten aufgebaut sind, die zwischen den Schnittstellen ausgetauscht werden und was diese Nachrichten beinhalten können. Alle Wörter innerhalb der Nachrichten benötigen eine festgelegte Bedeutung, um von allen Unternehmen auf die gleiche Weise verstanden zu werden. Die nächste Komponente ist ein Mechanismus, um die *Rollen und Choreografie des Nachrichtenaustauschs* abbilden zu können. Die Choreografie dient der Festlegung des Kommunikationsmusters zwischen den am öffentlichen Geschäftsprozess beteiligten Partnern und legt damit fest, wie der Dialog zwischen den Unternehmen geführt wird. Den letzten Bestandteil bilden die Vorkehrungen für eine *sichere und vertrauenswürdige Kommunikation* zwischen den Partnern durch die Verwendung von Zertifikaten und digitalen Signaturen für den Nachrichtenaustausch [Masud, 2003].

Das RosettaNet-Standardisierungsframework deckt zwar diese Komponenten vollständig ab, jedoch ergeben sich einige Einschränkungen, die durch die Verwendung von WebServices aufgehoben werden können.

Die öffentlichen Schnittstellen bei RosettaNet sind zwar „öffentlich“ gegenüber dem Partner, für den diesen PIP eingerichtet wurde, jedoch nicht gegenüber weiteren Geschäftspartnern. Für diese muss gegebenenfalls der gleiche PIP erneut eingerichtet werden, wodurch PIPs eher als „halb-öffentlich“ bezeichnet werden müssten. Webservice-Schnittstellen sind der Definition zufolge immer für mehrere Partner verfügbar, beziehungsweise erscheint es unzweckmäßig eine Webservice-Schnittstelle für nur genau einen Partner zu implementieren [Masud, 2003].



Das vordefinierte Vokabular (RosettaNet Business Dictionary und Technical Dictionary) enthält zehntausende Begriffe und Definitionen für eine Vielzahl von geschäftlichen Vorfällen. Die Kommunikation über WebServices ist auf ein solches Vokabular angewiesen und kann durch geeignete Mechanismen auf diese Wörterbücher zurückgreifen. Die Standardisierung der RosettaNet-Nachrichten mittels XML-DTD und XML-Schema erleichtert dabei die Verwendung im XML-basierten WebService-Umfeld [RosettaNet, 2003, 4]

Bei der Choreografie des Nachrichtenaustauschs bietet RosettaNet mehrere Hundert vordefinierter PIPs. Das Problem liegt darin, dass diese PIPs nicht in Form einer ausführbaren Geschäftsprozessmodellierungssprache, sondern durch UML-Aktivitätsdiagramme, Tabellen, textuellen Beschreibungen, DTDs und XML-Schema-Definitionen beschrieben sind. Sinnvoller wäre es, die Nachrichten-choreografie der PIPs mit einer standardisierten Prozessausführungssprache wie BPEL oder BPEL4WS zu beschreiben, wodurch sich zahlreiche neue Möglichkeiten bieten, da BPEL eine Vielzahl von logikbasierten Kontrollflusselementen unterstützt, wie SWITCH- und WHILE-Anweisungen [Masud, 2003]. Auf dem Gebiet der Sicherheit und Vertrauenswürdigkeit liefert RosettaNet zwar Vorgaben zur Implementierung einer sicheren Nachrichtenübertragung innerhalb eines PIP, nicht jedoch zur Prüfung der Gültigkeit von Nachrichten bei komplexen Geschäftsprozessen mit mehreren Partnern [Masud, 2003].

Die nachfolgende Abbildung verdeutlicht die Zusammenhänge und zeigt die durch RosettaNet abgedeckten Bereiche zur Ausführung von Geschäftsprozessen im Vergleich zu Standards bei WebServices und ebXML. Zwar bietet RosettaNet keine Mechanismen für die Beschreibung von privaten Prozessen oder die Verwendung zentraler Verzeichnisse (proprietäre PIP-Verzeichnisse bleiben unbeachtet), schließlich bildet dies nicht den Kern des Standardisierungsframeworks, jedoch bietet RosettaNet Hunderte vordefinierter Geschäftsprozessschablonen und die zugehörigen Wörterbücher.

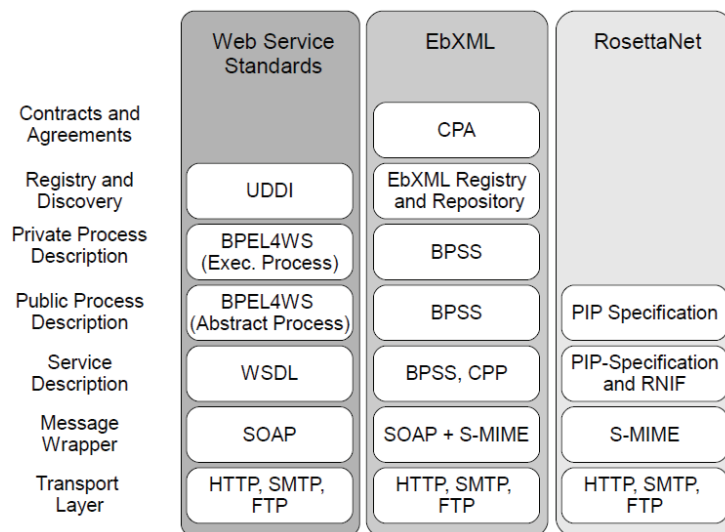


Abbildung 16: Vergleich von RosettaNet, WebServices und EbXML in Anlehnung an [Masud, 2003]

Welche Gründe sprechen noch für die Kombination von RosettaNet und WebServices?

[Masud, 2003] gibt einen schnelleren ROI für die Implementierung von RosettaNet, aber auch für WebServices an, sofern nicht nur Geschäftsprozesse mit einem einzelnen Partner implementiert werden sollen, sondern das eigene Unternehmen in ein Netzwerk eingebunden werden soll. In diesem Fall kann sowohl von der großen Akzeptanz und Verbreitung von RosettaNet, als auch den Vorteilen bei der Verwendung von WebServices profitiert werden. Durch den Mehraufwand, der aus der Kombination beider Ansätze resultiert, ergeben sich jedoch zahlreiche neue Möglichkeiten. Weiterhin ist dieser Ansatz in den USA besonders interessant, wo viele amerikanische Unternehmen RosettaNet zur Koppelung ihrer IT-Systeme verwendet haben und die vorhandenen PIP-Spezifikationen nicht verwerfen müssen, wenn sich in Richtung serviceorientierter Architekturen und WebServices weiterentwickeln wollen. In diesem Falle können die PIPs weiterhin funktionieren, aber darüber hinaus auch in BPEL-

Prozesse integriert werden, weil diese eine Abstraktionsebene höher liegen, als die PIPs, die von ihrer Verwendung in einem umfassenderen Kontext nichts erfahren. Der Vorteil dieses Ansatzes besteht darin, die zwischen zwei Partnern vorhandenen PIPs noch für beliebig viele weitere Partner über die BPEL-Schnittstellen zur Verfügung stellen zu können, ohne dass die neuen Partner mit den beiden erstgenannten ihre PIPs erneut definieren müssen. Die „bidirektionale Beschränkung“ des RosettaNet-Ansatzes wird durch dieses Konzept teilweise aufgehoben [Khalaf, 2005, 371].

Sowohl WebServices, als auch RosettaNet verwenden XML als Basissprache für Nachrichten und internetbasierte Kommunikationsprotokolle, wie SMTP, HTTP und FTP für den Nachrichtenaustausch [RosettaNet, 2003, 3]. Während RosettaNet ein elektronisches, dokumentenbasiertes Interaktionsmodell für die Geschäftspartner unterstützt, bieten WebServices die Möglichkeit, Anwendungsfunktionalitäten zu kapseln und über wohl definierte Schnittstellen zur Verfügung zu stellen. Zur Unterstützung eines elektronischen Geschäftsprozesses kann die Prozessausführungsumgebung Nachrichten an diese Schnittstellen senden und WebServices als Bausteine innerhalb der Prozesskette verwenden, worauf im Abschnitt 2.3 bereits eingegangen wurde.

Bei RosettaNet werden die Schnittstellen jedoch innerhalb des PIP von vorn herein festgelegt und gelten damit als statisch. Eine Änderung der Schnittstellen ist nur durch die Änderung der PIP möglich. Anders gesagt, legt der PIP genau fest, von wem wann welche Daten kommen, bei der Verwendung von WebServices kann im Extremfall zur Entwurfszeit des Prozesses noch keine einzige Schnittstelle feststehen, weil diese dynamisch während der Laufzeit gefunden und konfiguriert werden. Ein höherer Grad an Flexibilität erfordert jedoch eine komplexere Infrastruktur, als es bei RosettaNet der Fall ist [RosettaNet, 2003, 3].

*„RosettaNet can leverage Web services technology today in multiple ways to offer an even higher level of functionality for the cross-enterprise supply chain communications it serves. Specifically, Web services, itself a standards-based technology, could be incorporated as the underlying infrastructure to enable the deployment of RosettaNet-compliant documents directly onto the Internet, in full accordance with the RosettaNet Architecture. RosettaNet e-business processes or PIPs could then be advertised, discovered and accessed in the similar public settings as other services.“* [RosettaNet, 2003, 6]

Um RosettaNet-PIPs in BPEL-Prozessen verwenden zu können, muss zunächst die PIP-Spezifikation transformiert und in die Webservice-spezifischen Formate umgewandelt werden. [Masud, 2003] beschreibt zu diesem Zweck ein Transformationskonzept. Im ersten Schritt werden die Nachrichten eines PIP als WSDL-Operationen neu abgebildet. Dafür wird die DTD aus der PIP-Spezifikation benötigt, welche für alle PIPs zur Verfügung steht. Diese DTD muss zur Verwendung in WebServices jedoch in XML-Schema überführt werden. Da viele RosettaNet-PIP-DTDs separat entwickelt wurden, ohne ein übergeordnetes Metamodell zu verwenden, müssen einige Elemente aus den DTDs in der XML-Schema-Beschreibung neu definiert werden.

Hierbei offenbart sich eine weitere Schwäche von RosettaNet bei der Verwendung mehrerer PIPs in einem komplexen Geschäftsprozess, jedoch entwickelt das RosettaNet-Konsortium derzeit einheitliche XML-Schema-Spezifikationen für alle PIPs. Nachdem die XML-Schema-Spezifikation zur Verfügung steht, kann diese zur Erzeugung der WSDL-Operationen verwendet werden. Diese können wiederum durch BPEL4WS in komplexe Geschäftsprozesse eingebunden werden. Die Choreografie des Nachrichtenaustauschs in der PIP-Spezifikation kann als nächstes als abstrakter und ausführbarer Geschäftsprozess in BPEL4WS modelliert werden, bei dem auf die WSDL-Operationen zugegriffen wird. Die Exception Messages des RNIF werden durch den Exception Handling Mechanismus von BPEL4WS aufgefangen und entsprechend verarbeitet. Es gibt jedoch auch PIP-Bestandteile, die nicht nach BPEL4WS übertragen werden können. Dazu gehören die Kontrollparameter für den Nachrichtenaustausch, wie zum Beispiel die Anzahl der Wiederholungsversuche bei Fehlschlägen oder die maximale Ausführungsdauer für eine Aktivität. Manuelle Anpassungen und Eingriffe in die Prozessspezifikationen sind in diesem Fall unumgänglich.

Im Kapitel 4.2 wird ein Fallbeispiel für einen konkreten PIP vorgestellt und zum einen die Verwendung von Services im unternehmensinternen Bereich und zum anderen die Verwendung von WebSer-

vices für den unternehmensübergreifenden Bereich dargestellt. Dort wird das eben beschriebene Transformationskonzept aufgegriffen.

## 4 Unterstützung des Vertriebs im Finanzdienstleistungsbereich durch RosettaNet

### 4.1 Referenzprozess Vertrieb als Beispiel eines unternehmensübergreifenden Prozesses

Nachdem in den vorangegangenen Kapiteln auf verschiedene Möglichkeiten und Ansätze der Geschäftsprozess- und Servicemodellierung eingegangen wurde, beschäftigt sich der vierte Teil der Arbeit mit einem konkreten Beispiel eines Geschäftsprozesses für den Dienstleistungsbereich. Dabei handelt es sich um den Vertriebsprozess dessen Referenzprozess dem Sourcing-Modell im Zahlungsverkehr von Frei und Reitbauer entnommen wurde [Frei et. Reitbauer, 2006, 19]. Im Abschnitt 3.5.3 wurden die für RosettaNet verfügbaren Cluster und Segmente zur Unterstützung des Austauschs elektronischer Geschäftsdokumente auf Grundlage von PIP-Spezifikationen bereits erwähnt. Die vorhandenen RosettaNet-Spezifikationen sind in erster Linie für Unternehmen der Produktionsindustrie, Halbleiterfertigung, Logistik und Telekommunikation entwickelt worden. Welche Möglichkeiten sich bei der Verwendung von RosettaNet für den Dienstleistungsbereich, insbesondere den Bankensektor bieten, bildet den Gegenstand der folgenden Untersuchung. Zu diesem Zweck erfolgt ein Abgleich der für RosettaNet verfügbaren PIPs mit den im „Referenzprozess Vertrieb“ vorhandenen Prozessschritte. Es soll die Frage beantwortet werden, welche Prozessbestandteile des Vertriebs gegebenenfalls durch die Verwendung von RosettaNet unterstützt werden können, das heißt, welche Potenziale RosettaNet für die Unterstützung des Vertriebs bietet.

Das RosettaNet Konsortium veröffentlicht in regelmäßigen Abständen einen Teil der vorhandenen PIPs für das Publikum, um Interessenten einen Einblick in das Framework und die unterstützten Geschäftsprozesse zu geben [RosettaNet, 2009]. Die neueste Version dieses Dokumentes mit einem Überblick zu den verfügbaren Clustern, Segmenten und PIPs wurde im Januar 2009 veröffentlicht. Leider ist die vollständige Spezifikation nur für registrierte Mitglieder von RosettaNet möglich, weshalb sich die folgende Betrachtung auf einen Teilbereich beziehungsweise Ausschnitt der spezifizierten Geschäftsprozesse beschränken muss. Die nachfolgende Abbildung zeigt eine Übersicht des Referenzprozess für den Vertrieb, im Anhang ist eine detaillierte Beschreibung der einzelnen Prozessschritte aufgeführt.

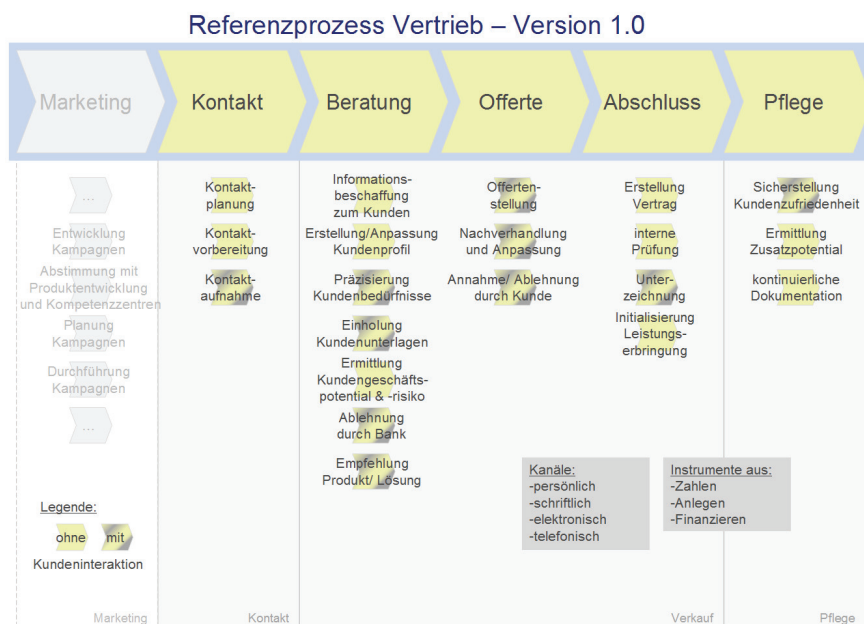


Abbildung 17: Referenzprozess Vertrieb [Eckert u. a., 2009]

Die folgende Tabelle enthält alle im Kernbereich des Vertriebs vorhandenen Prozessschritte in Kombination mit den in Anhang F beschriebenen PIP-Spezifikationen. Der Kombination beider Bereiche liegen folgende Annahmen zugrunde:

Jeder Prozessschritt (siehe obere Abbildung) wird entweder *intern* oder *in Zusammenarbeit mit dem Kunden* durchgeführt. Für alle internen Prozessschritte sind die PIP-Spezifikationen nicht anwendbar, weil sie ihrem Prinzip zufolge für die Kommunikation zwischen zwei Parteien ausgelegt sind und bei internen Prozessschritten keine Kommunikation zwischen der Bank und dem Kunden stattfindet. Eine Untersuchung der Verwendungsmöglichkeiten von RosettaNet bei der Kommunikation zwischen den Banken ist nicht Teil dieser Untersuchung. Ein Teil der Prozessschritte setzt eine Aktion des Anbieters voraus, worauf der Kunde reagiert, oder umgekehrt. Diese Vorgehensweise wird zunächst nicht betrachtet. Wer der Auslöser eines speziellen Prozessschrittes ist, muss im Einzelfall geprüft werden.

Einige Prozessschritte beziehen sich auf eine Menge von (unbekannten) Kunden (Leads), während andere Prozessschritte auf einen spezifischen, identifizierbaren Kunden gerichtet sind. Im ersten Fall könnte RosettaNet den Prozessschritt durch Verwendung eines anderen elektronischen (Werbe-) Mediums verwendet werden, beispielsweise bei der Erstellung von Werbe-E-Mails an potenzielle Kunden, bereits vorhandene Kunden können über die schon bestehenden PIPs mit (neuen) Produktinformationen beliefert werden. Der Referenzprozess Vertrieb wird für den Dienstleistungsbereich angewendet, insbesondere für den Bankensektor. Dadurch müssen die Bestandteile der PIP-Spezifikationen von der „materiellen Basis“ von RosettaNet, die sich häufig auf physische Güter und Waren bezieht, auf die „immaterielle Basis“ des Dienstleistungssektors teilweise abstrahiert werden. Eine Erklärung dazu ist an entsprechender Stelle gegeben.

Die Reihenfolge der genannten PIPs entspricht nicht zwangsläufig der zeitlichen Reihenfolge, in der die PIPs während des Prozessschrittes aktiv werden. Es handelt sich lediglich um eine alphanumerische Aufzählung der möglichen PIPs für diesen Prozessschritt. Die folgende Tabelle enthält einen Ausschnitt der Kombination des Vertriebsprozesses mit den vorhandenen RosettaNet PIPs. Eine vollständige Übersicht ist im Anhang G enthalten.

<b>Kombination des Referenzprozess Vertrieb mit den PIP-Spezifikationen von RosettaNet</b>	
Prozesse	Definition und unterstützende PIPs
Kontakt	siehe Anhang G
Beratung	siehe Anhang G
Informationsbeschaffung	laut Referenzprozess rein interner Prozess, daher: keine PIP-Unterstützung
Erstellung/Anpassung Kundenprofil	Erstellung (Neukunden)/Anpassung (Bestandskunden) des Kundenprofils: 1A1: Request Account Setup 1A2: Maintain Account
Präzisierung Kundenbedürfnisse	Präzisierung der geäußerten Kundenbedürfnisse: 2A3: Query Marketing Information 2A4: Query Sales Promotion & Rebate Information 2A6: Query Product Lifecycle Information 2A7: Query Product Discontinuation Information 2A13: Distribute Material Composition Information (Beschreibung der Zusammensetzung/Konfiguration eines Produktes) 2A15: Request Material Composition Information (Anfrage der Zusammensetzung/Konfiguration eines Produktes)
Einholung Kundenunterlagen	Einholung von Kundenunterlagen, die das Kundenprofil bestätigen/ergänzen/ausbauen 1A2: Maintain Account (Aktualisierung/Erweiterung des Kundenprofils)
Ermittlung Kundengeschäftspotenzial und -risiko	Interner Prozess, keine PIP-Unterstützung
Ablehnung durch Bank	Ablehnung von weiterer Beratung für gesamten Kunden oder einzelne Produkte 2B7: Notify of Product Change (Kunde wird darüber benachrichtigt, dass ein Produkt nicht mehr zur Verfügung steht)
Empfehlung Produkt/ Lösung	Empfehlung eines Produkts/einer Produktklasse/eines Lösungspakets 2A1: Distribute Product Catalog Information (Versand des aktuellen Produktkatalogs/Produktvorschlags an die Kunden) 5C1: Distribute Product List (Werbeangebot mit Produktvorschlägen an die Kunden)
Offerte	siehe Anhang G
Abschluss	siehe Anhang G
Pflege (Betreuung)	siehe Anhang G

Tabelle 9: Kombination des Referenzprozess Vertrieb mit den PIP-Spezifikationen von RosettaNet

Die Kombination der Prozessschritte des Vertriebs mit den vorhandenen PIP-Spezifikationen liefert folgendes Ergebnis: Von den 20 Prozessschritten des Referenzprozesses erfordern genau die Hälfte (10) eine Interaktion mit dem Kunden und stellen damit einen organisationsübergreifenden Geschäftsprozess dar. Für diese Prozessschritte sind zwischen einer und neun verschiedener PIPs vorhanden. Damit werden alle Prozessschritte des Vertriebs, die eine Interaktion mit dem Kunden erfordern, zumindest theoretisch durch RosettaNet unterstützt. Wie diese Unterstützung im Einzelfall konkret gestaltet werden kann bildet nicht den Kern dieser Untersuchung, jedoch wird im letzten Abschnitt der Arbeit (Kapitel 4.3) ein Implementierungsmuster für einen PIP aufgezeigt, die zur Auslösung einer Bestellung des Kunden dient und bei/nach der Bestätigung der Offerte eingesetzt werden könnte. Im

Anhang G befindet sich die vollständige Übersicht aller Prozessschritte des Vertriebs mit zusätzlichen Anmerkungen, in welchem Zusammenhang die genannten PIPs innerhalb der Prozessschritte verwendet werden könnten.

## 4.2 Fallbeispiel Bestellung im Rahmen des Vertriebsprozesses

Nachdem die allgemeine Verfügbarkeit von PIPs zur Unterstützung des Vertriebs untersucht wurde, soll ein konkreter PIP und deren Auswirkungen beim Dienstleistungsunternehmen und seinem Kunden genauer beschrieben werden. Der PIP 3A4 ist für diesen Zweck geeignet und dient dem Kunden dazu, eine Bestellung auszulösen. Wie im nachfolgenden Geschäftsprozessdiagramm zu sehen ist, können vor und nach dem Versand der Bestellung noch weitere PIPs aktiv werden, beispielsweise um die Verfügbarkeit und den Preis zu testen, die Bestellung zu ändern oder zu stornieren oder sich über den Status der Bestellung zu informieren. Die Beschreibung des PIP 3A4 lautet folgendermaßen:

*„PIP 3A4, Purpose: To support a process between trading partners that involves issuing a purchase order and acknowledging that purchase order. The PIP also supports the capability to cancel or change the purchase order based on the acknowledgment response and to acknowledge, at the line level, if the order is accepted, rejected, or pending.“ [RosettaNet, 2009]*

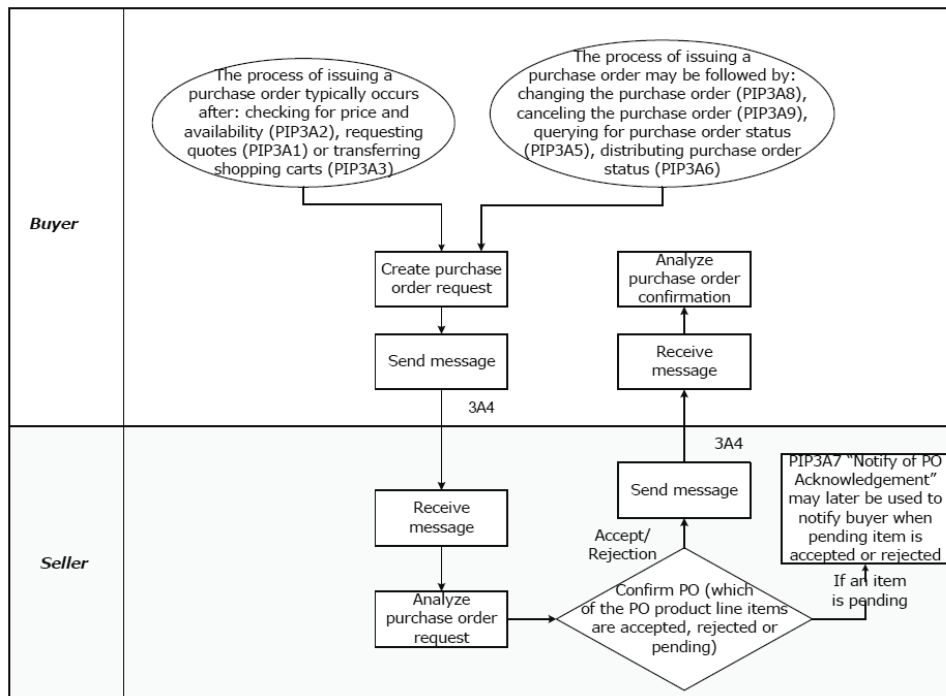


Abbildung 18: RosettaNet PIP 3A4: Request Purchase Order [RosettaNet, 2009]

Nachdem die Bestellung durch den Kunden aufgegeben wurde, wird innerhalb der Bank eine Reihe von Teilprozessen ausgelöst, die durch den Referenzprozess zum Wertpapiergeschäft (Fokus Abwicklung) beschrieben werden, der in folgender Abbildung dargestellt ist.

## Unterstützung des Vertriebs im Finanzdienstleistungsbereich durch RosettaNet

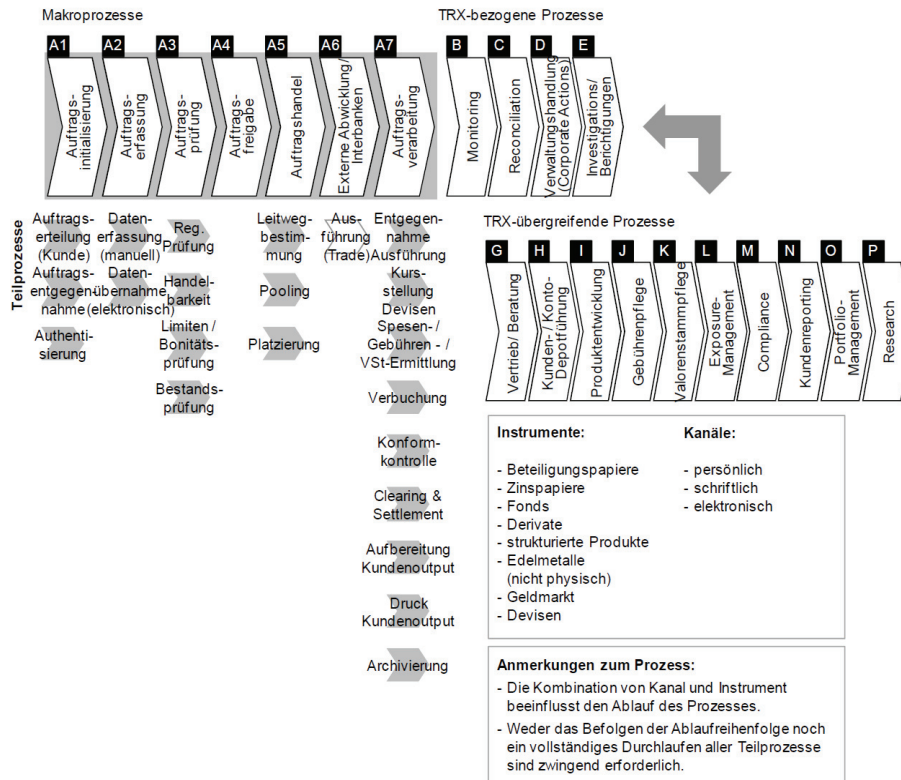


Abbildung 19: Referenzprozess zum Wertpapiergeschäft [Hoffmann und Reitbauer, 2009, 74]

Die RosettaNet Spezifikation zum Auslösen von Bestellungen bezieht sich in diesem Fall auf den Kauf von Wertpapieren durch einen Kunden der Bank. Der Referenzprozess beginnt mit der Auftragserteilung durch den Kunden und findet sein Gegenstück im Schritt „Create Purchase Order Request“ und „Send Message“ auf der Seite des Kunden. Der Eingang der Nachricht bei der Bank wird im Referenzprozess als Auftragsentgegenahme bezeichnet. Die Authentifizierung des Kunden, in diesem Falle der RosettaNet-Nachricht, erfolgt elektronisch. Das Gleiche gilt für den Schritt der Auftragsfassung, denn hier verläuft die Datenübernahme für den aktuellen Auftrag ebenfalls elektronisch, in Form einer RosettaNet Business Action Message. Innerhalb der Bank beginnen anschließend eine Reihe von internen Prozessschritten, wie die Auftragsprüfung, Auftragsfreigabe, Auftragshandel und die (externe) Abwicklung (Trade), bei denen Services zur Unterstützung dieser Prozessschritte eingesetzt werden können, und gegebenenfalls weitere RosettaNet-PIPs für die Interbankenkommunikation. Am Ende erfolgt die Auftragsverarbeitung bei der dem Kunden über den PIP eine Nachricht zum Ausführungsstatus der Bestellung gesendet wird. Innerhalb der PIP-Spezifikation ist eine Abwicklung vom gewünschten Prozessverlauf ebenfalls enthalten, wenn es auf der Seite des Anbieters zu Problemen kommt, beispielsweise wenn der Käufer von der Bestellung zurücktritt oder die Bestellung nicht ausgeführt werden kann (siehe Abbildung).

Als nächstes soll ein Teilbereich des eben beschriebenen Szenarios genauer betrachtet werden. Im Abschnitt 3.5.7 wurde die Möglichkeit angesprochen, RosettaNet-PIPs in Form von WebServices abzubilden und durch die Verwendung von BPEL4WS komplexe Geschäftsprozesssszenarien zwischen mehreren Unternehmen auf Basis von RosettaNet zu implementieren. Zumindest für die Interaktion zwischen zwei Unternehmen soll die Verwendung von RosettaNet, WebServices und BPEL4WS exemplarisch aufgezeigt werden. In Abbildung 18 ist bereits ein UML Aktivitätsdiagramm für den Aufbau des PIP 3A4 für den Bestellprozess enthalten. Um die Choreografie des Nachrichtenaustausches präziser darzustellen, empfiehlt sich die Verwendung eines UML Sequenzdiagramms, welches in folgender Abbildung dargestellt ist.



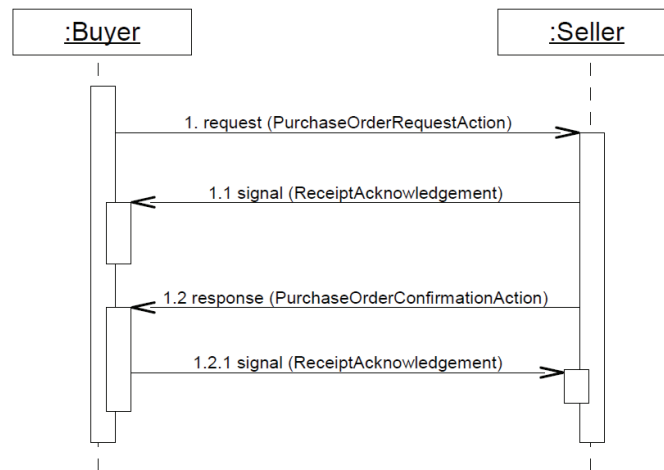


Abbildung 20: UML Sequenzdiagramm der RosettaNet-PIP 3A4 in Anlehnung an [Masud, 2003]

Als nächstes werden die Grundkonzepte der Geschäftsprozessmodellierung mit BPEL, der Verwendung von WebServices und der Nachrichtenaustausch mittels RosettaNet anhand des PIP 3A4 dargestellt.

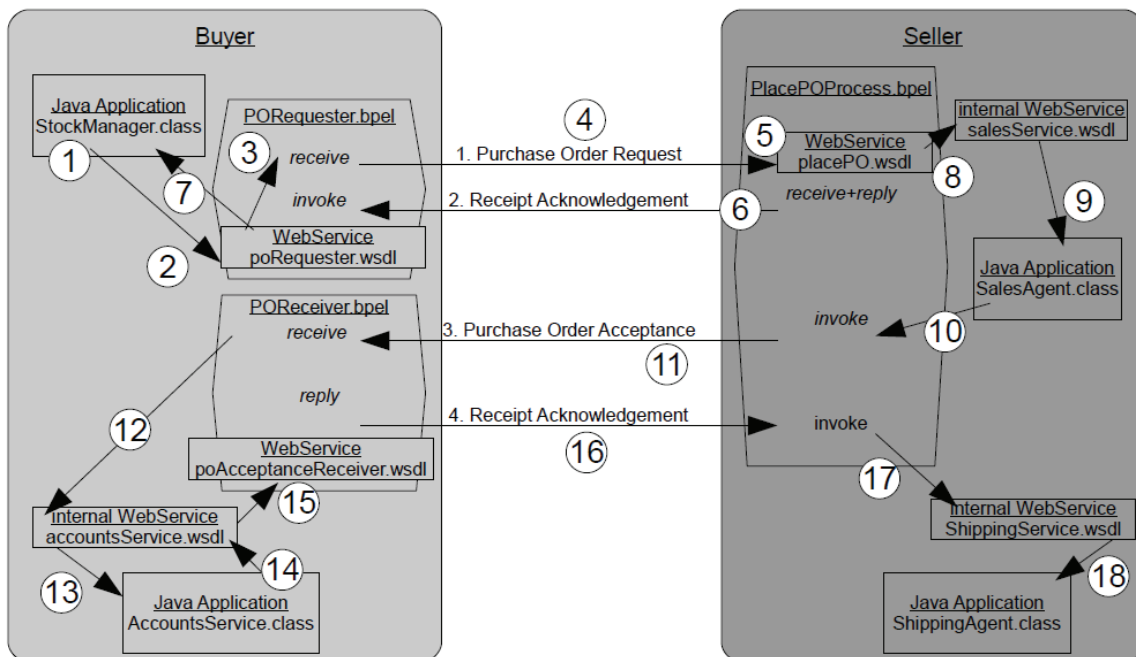


Abbildung 21: PIP 3A4 mit WebServices und BPEL4WS in Anlehnung an [Masud, 2003]

Die nachfolgende Aufzählung erläutert alle Schritte dieses Geschäftsprozesses, bei dem die Bestellung eines Wertpapiers bei einer Bank exemplarisch abgebildet ist.

1. Beim Kunden entsteht der Bedarf nach einem oder mehreren Wertpapieren, die unternehmensintern durch eine Back-End-Anwendung (StockManager.class) verwaltet werden.
2. Die Back-End-Anwendung (StockManager) ruft einen WebService zum Versand von Bestellungen auf (PORequester.bpel).
3. Der BPEL-Prozess (PORequester.bpel), in den der WebService eingebunden ist, reicht die Anfrage an die weiter, indem eine RosettaNet-Nachricht versendet wird.
4. Der Purchase Order Request wird ausgelöst und an die entsprechende Gegenstelle gesendet.

5. Die Gegenstelle liegt ebenfalls in Form eines Webservice (PlacePO.wsdl) vor und empfängt die Anfrage des Kunden.
6. Gemäß der PIP-Spezifikation und des BPEL-Prozesses erfolgt sofort eine Antwort an den Webservice (poRequester.wsdl) des Kunden.
7. Die BPEL-Definition (PORequester.bpel) leitet die Antwort der Bank an den StockManager weiter.
8. Auf Seite der Bank wird die Bestellung des Kunden an einen internen Webservice (salesService.wsdl) weitergeleitet.
9. Der interne Webservice der Bank ruft eine Back-End-Anwendung (SalesAgent.class) zur Prüfung der Bestellung auf. (Verfügbarkeit, Preis, ...)
10. Der SalesAgent ruft den Webservice (salesService.wsdl) wieder auf, um den Kunden zum Beispiel Verfügbarkeits- und Preisinformationen zu übermitteln.
11. Es erfolgt auf RosettaNet-Ebene eine Antwort an den Kunden (Purchase Order Acceptance).
12. Der Webservice des Kunden zum Empfang der Antworten von der Bank (poAcceptanceReceiver.wsdl) empfängt die Antwort. Gemäß der BPEL-Prozessdefinition (POReceiver.bpel) wird ein interner Webservice des Kunden aufgerufen, um die Bezahlung der Bestellung zu veranlassen.
13. Der interne Webservice (accountsService.wsdl) leitet die Antwort (Preisinformationen etc.) weiter zu einer Back-End-Anwendung (AccountsService.class). (Mit dieser Anwendung und ggf. anderen PIPs kann die elektronische Bezahlung veranlasst werden. Dies wird hier nicht weiter betrachtet.)
14. Der AccountsService bestätigt die Bezahlung an den Webservice (accountsService.wsdl).
15. Dieser leitet die Antwort an den öffentlichen Webservice (poAcceptanceReceiver.wsdl) weiter.
16. Gemäß der RosettaNet-PIP-Spezifikation erfolgt die Antwort an den Verkäufer (Bank) in Form einer Receipt Acknowledgement.
17. Die Bank erhält die Bestätigung des Kunden über die Ausführung der Zahlung und leitet die Bestätigung an einen internen Webservice zur Zahlungsabwicklung weiter (ShippingService.wsdl).
18. Der interne Webservice leitet die Bestätigung weiter zum entsprechenden Back-End-System (ShippingAgent.class), der die Lieferung des Wertpapierses veranlasst. (Auch dafür könnte wieder ein PIP verwendet werden, worauf hier jedoch nicht näher eingegangen wird).

Anhand dieses kleinen Szenarios konnte das Zusammenspiel von Webservices, BPEL und RosettaNet zumindest grundlegend aufgezeigt werden. In diesem Fall liefert RosettaNet die Choreografie des Nachrichtenaustausches zwischen Kunde und Bank, BPEL ermöglicht die Kommunikation mit den Back-End-Anwendungen, die Nachrichten werden gemäß den Konventionen der RosettaNet-Wörterbücher codiert und die Webservices stellen standardisierte Schnittstellen, sowohl für den unternehmensinternen, als auch unternehmensübergreifenden Bereich zur Verfügung. Im nächsten Abschnitt wird der PIP 3A4 praktisch implementiert, um die Funktionsweise von RosettaNet zu demonstrieren. Dies geschieht jedoch ohne Berücksichtigung des Einsatzes von Webservices oder BPEL4WS.

### **4.3 Implementierung des Fallbeispiels mit dem ORACLE WebLogic Server**

Einen Teilbereich des Vertriebsprozesses aus Abschnitt 4.1 bildet der Vertragsabschluss zwischen der Bank und dem Kunden, mit dem der Kunde eine verbindliche Bestellung über die von ihm gewünschten Produkte oder Dienstleistungen abgibt. Im vorherigen Abschnitt wurde das Fallbeispiel der Bestellung zuerst aus der Sicht von RosettaNet betrachtet für die ein UML Aktivitätsdiagramm im PIP-

Standard enthalten ist. Anschließend wurde der Bestellprozess daraufhin untersucht, welche internen, privaten Prozesse in der Bank ausgelöst werden und welche Schnittstellen zu den Back-End-Systemen vorhanden sind. Als letztes wurde der schematische Aufbau einer RosettaNet-basierte Kommunikation über den PIP 3A4 zur Abwicklung der Bestellung unter Zuhilfenahme von WebServices und BPEL4WS beschrieben.

In diesem Abschnitt wird der Bestellprozess mit Hilfe des ORACLE WebLogic 10.3 Application Servers praktisch implementiert, um die wesentlichen Bestandteile einer RosettaNet-Implementierung und die dazu notwendigen Kenntnisse zu identifizieren. Dieses Produkt von ORACLE wurde aufgrund seiner Nähe zum ORACLE Business Process Management Studio 10.3 gewählt, welches zur aktuellen ORACLE Produktfamilie gehört und mit dem im nachfolgenden Kapitel 4.4 der Referenzprozess „Vertrieb“ mithilfe der BPMN abgebildet wurde. Weiterhin stellt ORACLE sehr ausführliche Dokumentationen zur Implementierung von PIPs zur Verfügung und den ORACLE WebLogic Server selbst, nach der Registrierung, in einer kostenfreien Version. Weiterhin ist ORACLE ein bedeutender Bestandteil des RosettaNet-Konsortiums.

Den Kern dieses Abschnittes bildet die Beschreibung der systematischen Vorgehensweise bei der Implementierung einer RosettaNet-basierten Kommunikation zwischen zwei Unternehmen, am Beispiel des PIP 3A4. Die beschriebene Vorgehensweise ist teilweise ORACLE-spezifisch, insbesondere in Bezug auf die Definition der Begriffe für das Geschäftsprozessmanagement und die Softwareprodukte zu dessen Unterstützung, jedoch wird versucht die Beschreibung zu verallgemeinern, so dass sie auch für andere Softwareanbieter Verwendung findet. Die Beschreibung der Implementierung schließt sich an das Kapitel 3.4.4 zur Implementierung organisationsübergreifender Prozesse, sowie das Kapitel 3.5.4 zum RNIF an. Bei ORACLE wird die B2B-Integration grundlegend als „Trading Partner Integration“ bezeichnet, wofür der Softwareanbieter eine Reihe von Produkten zur Verfügung stellt. ORACLE definiert Trading Partners folgendermaßen:

*„A **trading partner** is understood to be an entity that has an agreement with another entity to participate in a specific business transaction, or service, by playing a predefined role associated with a distinct business purpose. Trading partner applications form the nodes in system-to-system interactions among business partners.“* [ORACLE, 2008c]

Das Ziel der „Trading Partner Integration“ bildet die Erstellung einer *Top-Down-Ansicht* für alle Geschäftstransaktionen und Teilbereiche der gesamten Wertschöpfungskette für die Geschäftspartner, die über eine zentrale, internetbasierte Plattform verfügbar ist. Dadurch können die Geschäftspartner Einsicht in den Prozessstatus der Wertschöpfungskette erhalten und entsprechend auf Probleme und Engstellen reagieren. Um eine solche zentrale Plattform anbieten zu können, ist entweder eine zentrale Ausführungsumgebung für alle Geschäftspartner notwendig, oder die Kopplung der dezentralen Ausführungsumgebungen der einzelnen Partner zu einer zentralen Plattform [ORACLE, 2008c]. Die wesentlichen Kernaufgaben bzw. Bestandteile dieser zentralen Integrationsplattform lassen sich nach [ORACLE, 2008c] wie folgt zusammenfassen :

- Möglichkeiten zur Integration von öffentlichen und privaten Prozessen
- Unterstützung von anerkannten Industriestandards und Protokollen wie RosettaNet, ebXML und WebServices
- Unterstützung des Trading Partner Managements und des Repository-Zugriffs
- Abruf von Laufzeitinformationen der aktiven Geschäftsprozesse für Reporting, Monitoring und Performancemessung
- Unterstützung von Mechanismen für die Sicherheit, Prüfung, Nachweisbarkeit, Nachrichtenpersistenz, Recovery, Transaktionsintegrität

Um diesen Integrations- und Implementierungsanforderungen zu erfüllen, benötigt jeder Geschäftspartner einen Server, der die Anfragen (Nachrichten) der anderen Unternehmen empfangen und entsprechend auswerten kann. ORACLE bietet für diesen Fall den WebLogic Server an. Der Geschäftsbereich, in dem zwei oder mehr Unternehmen zusammen arbeiten möchten, und dafür elektronische Geschäftsprozesse entwickeln wollen, wird als Domäne bzw. Integrationsdomäne bezeichnet. Für

jeden Geschäftsbereich, in dem ein Unternehmen Geschäftsprozesse mit einem anderen Unternehmen implementiert, gibt es in diesem Fall eine Integrationsdomäne, bei der das eigene Unternehmen stets als „Local Trading Partner“ und jedes andere Unternehmen der gleichen Integrationsdomäne als „Remote Trading Partner“ bezeichnet wird [ORACLE, 2008c]. Zwischen den Unternehmen einer Integrationsdomäne werden Geschäftsprozesse abgewickelt, die bei ORACLE als Konversationen (conversations) beschrieben werden:

*„When trading partners exchange business messages for a business purpose, they participate in a **conversation**. A conversation is a series of one or more business messages exchanged between trading partners. The nature of each conversation is determined by its business purpose [...]“ [ORACLE, 2008c]*

Die Geschäftspartner übernehmen in den Konversationen Rollen und sind entweder der Initiator des Geschäftsprozesses, oder ein Teilnehmer des gleichen. Wenn die Geschäftsprozesse auf Grundlage von RosettaNet implementiert werden sollen, ist zunächst die Registrierung auf der Internetseite von [www.rosettanet.org](http://www.rosettanet.org) notwendig, um Zugang zu allen benötigten Dokumenten zu erhalten [ORACLE, 2008b]. Dazu gehören:

- Dokumente zum RNIF
- RNIF Technical Advisories: Updates und zusätzliche Informationen zum RNIF
- RNIF Technical Recommendations: Implementierungshinweise und Empfehlungen zum RNIF
- RNIF Business Signals, Service Header & Preamble: enthalten Message-Guidelines und XML-DTDs für den Aufbau der RosettaNet-Nachrichten
- PIP Blueprints und alle PIP-Spezifikationen

Die unterste Ebene jeder RosettaNet-Implementierung bildet die RosettaNet-spezifische Protokollschicht, die den Nachrichtenaustausch über das Internet ermöglicht und Vorgaben für die Verpackung, den Transport und die Sicherheit der Nachrichten beinhaltet. Für die Implementierung des Nachrichtenaustausches werden bei den PIPs folgende Entwurfsmuster (PIP Design Pattern) unterschieden:

- Asynchronous single-action / two-action activity
- Synchronous single-action /two-action activity

Dabei werden die Entwurfsmuster für die synchrone Kommunikation aufgrund der verhältnismäßig hohen Latenz- und Prozesszeiten im Internet und beim überbetrieblichen Bereich nur sehr selten implementiert und durch den ORACLE WebLogic Server nicht unterstützt [ORACLE, 2008b].

Alle Geschäftsnachrichten werden ab RNIF Version 2.0 als „RosettaNet Business Messages“ (kurz RBM) bezeichnet. Jede RosettaNet Business Message, die von einem Geschäftspartner empfangen wird, leitet das Transport-Servlet zunächst an den RosettaNet-Decoder weiter. Dieser liest den protokollspezifischen Nachrichtenheader aus, identifiziert den sendenden Geschäftspartner und leitet die Nachricht an eine JMS Queue weiter. Bevor die Nachricht im Unternehmen weiter verarbeitet wird, erfolgt zunächst eine ausführliche Prüfung. Die PIP-Definitionen enthalten zu diesem Zweck detaillierte Validierungsregeln für die durch einen PIP austauschbaren Nachrichten. Die Validierung geschieht mithilfe von XML Schema Documents (XSD), die für alle Nachrichten aller PIPs verfügbar sind [ORACLE, 2008b]. Bei ORACLE wird die Umsetzung einer RosettaNet-Lösung in die vier Phasen *Planung*, *Entwicklung*, *Deploying* (Veröffentlichung) und *Management* unterteilt.

Die Phase der *Planung* beinhaltet eine Reihe von Tätigkeiten durch die das Unternehmen in die Lage gebracht wird, die RosettaNet-Lösung zu entwickeln, das heißt die Grundvoraussetzungen für die Implementierung zu schaffen. Dazu gehört neben der ausgiebigen Informationsbeschaffung zu Richtlinien, Empfehlungen und Best Practises bei RosettaNet-Lösungen auch die Suche nach einem geeigneten Softwareanbieter, auf dessen Grundlage die Implementierung geschehen kann. Den wichtigsten Punkt bei der Planung bildet jedoch nicht nur die spätere technische Umsetzung und die dafür notwendigen Mittel, sondern die klare Bestimmung der fachlichen Anforderungen, die durch den Einsatz von RosettaNet erfüllt werden sollen. Folgende Fragen müssen dabei beantwortet werden:

Welche Geschäftspartner sollen integriert werden? Welche Geschäftsprozesse sollen zu welchen Partnern entwickelt werden? Welche Informationen werden von den Partnern für die Implementierung benötigt? Für jeden einzelnen Geschäftsprozess ist zu klären: Welche PIPs bietet RosettaNet für die Unterstützung dieses Geschäftsprozesses an? Welche PIP Version wird verwendet oder ist bei den Partnern im Einsatz? Welche RNIF Version wird für den Nachrichtenaustausch verwendet? Welche Anforderungen ergeben sich für die Back-End-Systeme und privaten Geschäftsprozesse zur Unterstützung von RosettaNet? Müssen neue, interne Geschäftsprozesse entwickelt und integriert werden? [ORACLE, 2008b] Bei der Implementierung jedes PIPs sind folgende Punkte ebenfalls zu klären:

- das zu verwendende PIP Design Pattern (asynchronous one-action / two-action activity)
- die Anzahl und Art der auszutauschenden Geschäftsnachrichten
- das Schema (Aufbau) für jede Geschäftsnachricht
- der Erfolgspfad für den Nachrichtenaustausch
- die möglichen Fehlerfälle und deren Behandlung, sowohl durch das RNIF, als auch die internen Geschäftsprozesse und Back-End-Systeme (Wiederholungen, Timeouts, fehlende Nachrichten, Fehlerbenachrichtigungen, Logging, Recovery, etc.)

Die zweite Phase der *Entwicklung* wird auch als Design-Time bezeichnet und umfasst Design, Build und Test der RosettaNet-Lösung. Auch hier wird wieder eine Reihe von Schritten unterschieden, die im folgenden erklärt werden. Zuerst wird eine Domain auf dem Server des Anbieters erstellt, für den sich das Unternehmen entschieden hat (z. B. ORACLE WebLogic Server, Microsoft BizTalk Server), in der alle Geschäftsbeziehungen gebündelt werden, die mit RosettaNet implementiert werden sollen. Dies entspricht gleichzeitig einer fachlichen Domäne der überbetrieblichen Zusammenarbeit, die mit RosettaNet auf technischer Ebene verwirklicht wird. Anschließend werden die benötigten PIP-Distributionen, die während der Planungsphase identifiziert und beschafft wurden, in die Ausführungsumgebung des Servers importiert. Als nächstes erfolgt die Anpassung der PIPs an die lokalen Gegebenheiten mithilfe der verwendbaren Parameter wie IP-Nummern, Ports, Schnittstellen zu internen Prozessen und/oder Services. Für die Konfiguration der PIPs stellen die Softwareanbieter in der Regel entsprechende Dialogfelder, Tutorials oder Wizards zur Verfügung, jedoch muss die Anpassung der PIPs an die internen Geschäftsprozesse häufig auf Quellcodeebene geschehen. Anschließend werden die Sicherheitsbestimmungen für den unternehmensinternen Bereich (Wer darf im eigenen Unternehmen welchen unternehmensübergreifenden Geschäftsprozess auslösen?), als auch den unternehmensübergreifenden Bereich (Welcher Partner darf welche Geschäftsprozesse auslösen?) festgelegt. Am Ende erfolgt ein Test der Implementierung mit Testdaten [ORACLE, 2008b]. RosettaNet stellt zu diesem Zweck im Cluster 0 den PIP 0C2 zum Testen von asynchronen Verbindungen zur Verfügung.

Wenn die entwickelte Lösung erfolgreich getestet wurde, kann der Produktivbetrieb aufgenommen werden, und die Phase des *Deploying* beginnt. Dafür werden die Server der beteiligten Partner auf den produktiven Modus umgeschaltet und aneinander gekoppelt. Dafür werden die benötigten Informationen zum Trading Partner Management Repository hinzugefügt, wozu beispielsweise die Protokollbindungen für den Nachrichtenaustausch zählen. Weiterhin erfolgt der wechselseitige Import der Zertifikate der beteiligten Partner zur Signierung der Nachrichten. Die letzte Phase bildet das *Management* der RosettaNet-Lösung. Dazu gehört beispielsweise die Anpassung der Geschäftsprozesse an geänderte Bedingungen, sowie die Überwachung der Laufzeitperformance, Nachrichtenmenge und Ressourcenverwendung. Darauf wird an dieser Stelle nicht näher eingegangen.

Nachdem die Vorgehensweise und Bestandteile der Implementierung von RosettaNet im Allgemeinen beschrieben wurde, erfolgt als nächstes die konkrete Implementierung des Fallbeispiels anhand des Geschäftsprozesses zur Abwicklung von Bestellungen und der PIP 3A4. Da sich diese Beispielimplementierung auf einen einzelnen Rechner beschränken muss und der PIP später auch nur lokal ausgeführt wird, ergeben sich einige Einschränkungen. Die Vorlagen zur Implementierung der Bestellung sind zwar vollständig vorhanden, jedoch kann aufgrund der Prozesslogik der PIP nicht realitätsgetreu ausgeführt werden, weil für diesen Zweck zwei Server auf dem gleichen Computer gestartet werden müssten, die miteinander kommunizieren. Aufgrund der Vorgaben des RNIF kann eine realitätsge-

treue PIP-Kommunikation zwischen zwei Servern auf dem gleichen localhost jedoch nicht stattfinden [ORACLE, 2008d].

Daher wird der Bestellprozess soweit simuliert, dass die Bestellung vom Kunden an den Anbieter gesendet wird, dieser jedoch die Bestellung vollständig zurücksendet, anstatt dem Preis der Verfügbarkeit für diese Bestellung. Der Kunde reagiert darauf mit einer Fehlerbenachrichtigung (PIP 0A1: Notification of Failure) an den Anbieter. Die Angaben zur Menge der benötigten oder entstehenden Dateien während der Implementierung dient zur Darstellung des Implementierungsumfangs und der Komplexität einer minimalen RosettaNet-Implementierung. Die nachstehende Vorgehensweise zur Implementierung orientiert sich an einem Tutorial aus [ORACLE, 2008d]:

- 1) Installation des ORACLE WebLogic 10.3 Application Servers auf dem eigenen Computer
  - a) Der Download des Installationsprogramms war nach der Registrierung bei ORACLE über die ORACLE-Webseite möglich. (Umfang der Installation ca. 58.900 Dateien in 11.300 Verzeichnissen und 2,5 Gbyte)
- 2) Erstellung der Integrationsdomäne mithilfe des „WebLogic Configuration Wizards“
  - a) Angabe des Festplattenpfades, in dem alle zur Domäne gehörigen Konfigurationsdateien gespeichert werden, und wie die Domäne bezeichnet wird. (Umfang der Domäne ca. 12.300 Dateien in 3.700 Verzeichnissen und 250 Mbyte)
  - b) Erstellung eines Administratorenprofils zur Verwaltung der Integrationsdomäne
- 3) Start des Integrationssservers für die neue Domäne auf dem localhost.
  - a) Die Domänenkonfiguration ist jetzt über den localhost und Port 7001 möglich. Hier können weitere System- und Prozesskonfigurationen vorgenommen werden. Auch die Überwachung des Prozessstatus mit dem Process Instance Monitoring ist möglich. Siehe Bildschirmfotografie in Anhang I.
- 4) Start der Entwicklungs- und Ausführungsumgebung „ORACLE Workshop for WebLogic 10gR3“ (Bestandteil des vorherigen Installationspaketes; auf Eclipse basierendes Prozessentwicklungswerkzeug)
- 5) Erstellung eines neuen Projektes zur Implementierung eines öffentlichen Geschäftsprozesses, in dem der PIP 3A4 implementiert werden soll.
- 6) Import der benötigten PIP-Spezifikation für den PIP 3A4 in das neue Projekt. Diese liegt in Form von Java-Klassen und Paketen vor und wird von ORACLE zur Verfügung gestellt. (Umfang ca. 9.100 Dateien in 330 Verzeichnissen und 60 Mbyte)
  - a) Im Projektverzeichnis stehen dann alle Dateien zur Verfügung, die zur Ausführung des PIP benötigt werden (Siehe Anhang K). Diese sind auszugsweise folgende:
    - i) *PIP3A4.java*: Klasse zur Abwicklung des öffentlichen Geschäftsprozesses des Verkäufers, der die Bestellung über den PIP entgegennimmt
    - ii) *PIP3A4Private.java*: Klasse zur Abwicklung des privaten Geschäftsprozess des Verkäufers zur Bearbeitung der Bestellung (Kopplung zum Back-End-System)
    - iii) *PIP3A4Participant.java*: Klasse auf Seite des Käufers, mit der die Bestellung versendet wird
    - iv) *PIP3A4ParticipantPrivate.java*: Klasse zur Abwicklung des privaten Geschäftsprozesses des Käufers, der den PIP3A4 ursprünglich ausgelöst hat (Kopplung zum Back-End-System)
  - b) Für die vier genannten Klassen ist die visuelle Repräsentation im Anhang J enthalten.
- 7) Zur Veröffentlichung des PIP muss die Verbindung zur umschließenden Domain noch hergestellt werden. Dafür sind manuelle Anpassungen im Quellcode notwendig, wo die benötigten Konfigurationsdateien für die Ausführung des PIP innerhalb dieser Domain enthalten sind.

- 8) Dann werden die PIP-Java-Klassen kompiliert und an den Server gesendet. Der PIP wurde in diesem Moment auf dem Server veröffentlicht (Deployment). Ab dann steht sie innerhalb dieser Domain zur Verfügung und kann Anfragen verarbeiten.
- 9) Als nächstes kann dem PIP die erste Bestellung übergeben werden. Dafür stellt ORACLE eine Beispielbestellung zur Verfügung, die in Form einer XML-Datei vorliegt und dem Server (in diesem Fall „per Hand“ zur Demonstration) übergeben wird. Üblicherweise geschieht dies später automatisiert durch einen Webservice oder privaten Prozess der den Inhalt der Bestellung an den PIP überträgt.
  - a) Sowohl innerhalb Ausführungsumgebung, als auch der Administrationskonsole der Domain werden jetzt die einzelnen Prozessschritte der oben genannten Prozessklassen angezeigt. Im Anhang K sind die Ergebnisse des Prozessablaufes nacheinander abgebildet. Siehe dazu die Bildschirmfotografien in Anhang K. Der zeitliche Verlauf des Nachrichtenaustausches ist als Sequenzdiagramm in Abbildung 20 dargestellt.
  - b) Wie bereits im Vorfeld erklärt, kann der Empfänger nur mit einer Fehlermeldung (PIP 0A1: Notification of Failure) antworten, da die RosettaNet-Nachricht aus seiner Sicht von ihm selbst gesendet worden ist. Der Kunde kann die Bestellung zwar aufgeben, empfängt jedoch nur diese Fehlermeldung als Antwort. Wie dieser Prozess abläuft ist in Anhang K genau dokumentiert.
- 10) Sofern Server oder Prozessentwicklungsumgebung nicht mehr benötigt werden, können beide jetzt beendet werden.

Das eben beschriebene Implementierungsszenario zeigt die wesentlichen Schritte zur Umsetzung einer RosettaNet-Lösung auf. Die grundlegende Vorgehensweise ist unabhängig vom Anbieter der zur Implementierung verwendeten Software, auf deren Grundlage das Fallbeispiel entwickelt werden könnte. ORACLE bietet noch weitere Tutorials zur Entwicklung weiterer PIPs und deren Anpassung an die privaten Geschäftsprozesse. An dieser Stelle sei auf die im Literaturverzeichnis gegebenen Quellenangaben verwiesen.

Im nachfolgenden und letzten Abschnitt wird der Vertriebsprozess mit dem auf Basis der Business Process Modeling Notation modelliert, um die Zusammenhänge zwischen den einzelnen Prozessschritten und die sie unterstützenden Services hervorzuheben.

#### **4.4 Modellierung des Vertriebsprozesses mithilfe der Business Process Modeling Notation**

Ein wesentlicher Punkt dieser Arbeit bildet das Gebiet der Geschäftsprozessmodellierung. Im vorherigen Abschnitt wurde der Referenzprozess Vertrieb daraufhin untersucht, welche Prozessbestandteile durch RosettaNet unterstützt werden könnten. Dieser Abschnitt verfolgt das Ziel, den Referenzprozess Vertrieb mithilfe eines Geschäftsprozessmodellierungswerkzeuges in einer weitverbreiteten und standardisierten Geschäftsprozessmodellierungssprache abzubilden. Als Modellierungswerkzeug wurde das ORACLE Business Process Management Studio 10.3 gewählt, welches von ORACLE kostenfrei, aber mit eingeschränktem Funktionsumfang, im Internet zur Verfügung gestellt wird. Eine Bildschirmfotografie der Anwendung befindet sich auf der letzten Seite von Anhang H.

Zur Darstellung des Vertriebsprozesses wird die Business Process Modeling Notation (BPMN) in der Version 1.1 verwendet, die durch das Modellierungstool unterstützt wird. Dabei ist jedoch zu beachten, dass es sich bei dem Referenzprozess um eine allgemeine Beschreibung des Vertriebs auf einer sehr hohen, fachlichen Ebene handelt, deren Übertragung auf ein Geschäftsprozessmodell zahlreiche Konkretisierungen erfordert. Beispielsweise erfordert die Interaktion zwischen der Bank und dem Kunden eine Aktion auf Seite des Kunden zum Empfang oder dem Versand von Nachrichten und Informationen, die im Referenzprozess nicht oder nur grob beschrieben sind, beziehungsweise aus den Aktionen auf Seite der Bank abgeleitet werden wurden.

Dieses Kapitel verfolgt jedoch nicht nur das Ziel, den Referenzprozess in einer standardisierten Notation abzubilden, sondern die Zusammenhänge zwischen dem Geschäftsprozess „Vertrieb“ und den

Services wiederzugeben, die den Vertriebsprozess unterstützen können. Als Grundlage dafür wurde die Arbeit von [Eckert u. a., 2009] herangezogen, in der für alle Prozessschritte die verfügbaren Services aufgeführt sind. Die folgende Abbildung enthält einen Teilbereich des Geschäftsprozessmodells und wird im Anschluss erklärt.

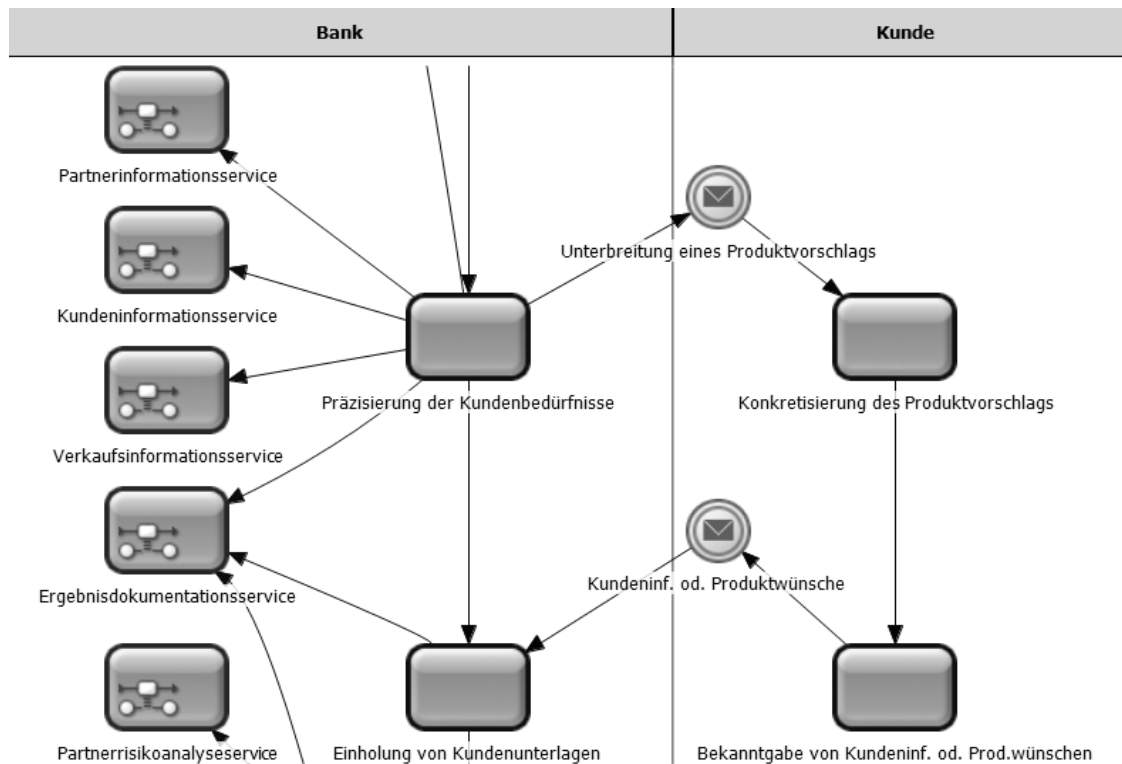


Abbildung 22: Modellausschnitt des Vertriebsprozesses auf Basis der BPMN Geschäftsprozessmodell erstellt mit dem ORACLE BPM Studio 10.3

Gemäß der BPMN ist das Geschäftsprozessmodell in Swimlanes gegliedert, welche die am Prozess beteiligten Partner und deren Aktivitäten enthalten. Auf der linken Seite ist die Bank dargestellt, auf der rechten Seite der Kunde. Blaue Rechtecke mit abgerundeten Ecken stellen einzelne Prozessschritte (Aktivitäten) und deren Interaktionen dar (schwarze Pfeile), rote Rechtecke mit abgerundeten Ecken und dem „Service“-Zeichen stellen Services dar, auf die während einer Aktivität zugegriffen werden kann (blaue Pfeile). Die Reihenfolge der Aktivitäten von oben nach unten entspricht der zeitlichen Reihenfolge. Dies trifft jedoch nicht auf die Services zu, deren Anordnung keine weiteren Informationen enthält. Zeigen mehrere blaue Pfeile zu einem Service, deutet dies seine Wiederverwendung in verschiedenen Kontexten an. Hinweise auf die Verwendung der Services bei anderen bankenspezifischen Prozessen (z. B. Anlegen), werden durch dieses Modell nicht abgebildet. Die Kommunikation zwischen den Partnern geschieht mithilfe von Nachrichten, in Form eines blauen Kreises mit Brief-Symbol. Im Anhang H befindet sich das vollständige Geschäftsprozessmodell, das von oben (Prozessbeginn) nach unten (Prozessende) zu lesen ist.

Neben der Möglichkeit, Geschäftsprozesse mit verschiedenen Notationen Modellieren zu können, bieten zahlreiche Modellierungswerkzeuge auch die Möglichkeit, verschiedene Arten von Services in die Geschäftsprozessmodelle zu integrieren. Dadurch können Services an die Verarbeitung von Aktivitäten und Aktionen gekoppelt werden. Beim ORACLE BPM Studio können folgende Services als externe Ressource in ein Prozessmodell eingebunden werden: CORBA Services, Java Messaging Services, JMX Services, Java-Klassenbibliotheken und deren Interfaces, Mail Incoming und Outcoming Services, Microsoft .NET und COM Services, SAP Services, WebServices und natürlich auch Services, die an den ORACLE Service Bus gebunden sind. An dieser Stelle ergibt sich die praktische Umsetzung der Kombination aus Service- und Prozessmodellierung, die im Hauptteil dieser Arbeit ausführlich beschrieben wurde.



## 5 Schluss

Den Schwerpunkt dieser Arbeit bildete die Untersuchung des Spannungsfeldes zwischen Service- und Prozessmodellierung, insbesondere für den unternehmensübergreifenden Bereich. Die einzelnen Teilbereiche einer serviceorientierten Geschäftsprozessmodellierung und -ausführung für den unternehmensinternen Bereich werden durch folgende Abbildung zusammengefasst.

serviceorientierte Geschäftsprozessmodellierung- und Ausführung  Bereich: Business Service Management	Geschäftsprozessmanagement	Modellierung von Geschäftsprozessen
		Ausführung von Geschäftsprozessen/Workflows
	Bestandteil serviceorientierter Architektur	Verwendung von Services zur Ausführung/Bearbeitung von Prozessschritten
		Anwendungssysteme/Back-End-Systeme, deren Funktionalität in Services gekapselt ist

Abbildung 23: Teilbereiche serviceorientierter Geschäftsprozessmodellierung

Wie die Studie von [Spath und Weisbecker, 2008, 7] des Fraunhofer Instituts zeigt, ist die Anzahl der auf dem deutschen Markt verfügbaren Geschäftsprozessmanagement- und Modellierungswerkzeuge angestiegen, woraus sich schlussfolgern lässt, dass dieser Bereich in den vergangenen Jahren stärker in das Bewusstsein vieler Unternehmen gerückt ist und auch weiterhin an Bedeutung zunimmt. Auf der anderen Seite deutet die steigende Zahl von Veröffentlichungen auf dem Gebiet der Servicemodellierung und Entwicklung serviceorientierter Architekturen auf den Bedeutungszuwachs dieses IT-Managementkonzeptes hin. Für den innerbetrieblichen Bereich stehen zahlreiche Entwicklungswerkzeuge für beide Gebiete zur Verfügung, deren Zusammenwachsen in Richtung einer serviceorientierten Geschäftsprozessmodellierung- und Ausführung durch viele Autoren befürwortet und durch die Entwicklung von entsprechenden Modellierungssprachen- und Notationen weiter vorangetrieben wird.

Im überbetrieblichen Bereich ergeben sich für dieses Konzept jedoch erhebliche Schwierigkeiten. Sollen unternehmensübergreifende Geschäftsprozesse für mehrere Unternehmen mit einer komplexen Prozesskette entwickelt werden, deren Verarbeitung bzw. Abarbeitung durch Services teil- oder vollautomatisch geschehen soll, ist die Heterogenität dieser Form der elektronischen Zusammenarbeit auf mehreren Schichten beziehungsweise Abstraktionsebenen zu überwinden. Ansätze wie RosettaNet, ebXML oder die Standards für BPMN, BPML und BPEL, sowie WebServices decken nur einen Teilbereich dieser Abstraktionsebenen ab und sind für sich allein betrachtet jedoch nicht ausreichend komplex, um all diese Aspekte zu umfassen.

Die zwei größten Problemfelder sind dabei die Verwendung von Prozessbeschreibungs- und Ausführungssprachen, die von allen beteiligten Geschäftspartnern gleichermaßen verstanden und angewendet werden, sowie die Entwicklung von Mechanismen zur Abrechnung von in Anspruch genommenen Services fremder Unternehmen, Transaktionssicherheit für die langen Prozessausführungszeiten, Vertrauen und Sicherheit für die Nachrichtenkommunikation, sowie der Prüfung von Rechten und Pflichten der Prozessteilnehmer, gemäß den festgelegten Geschäftsregeln. Die folgende Abbildung dient zur Abbildung der genannten Zusammenhänge.

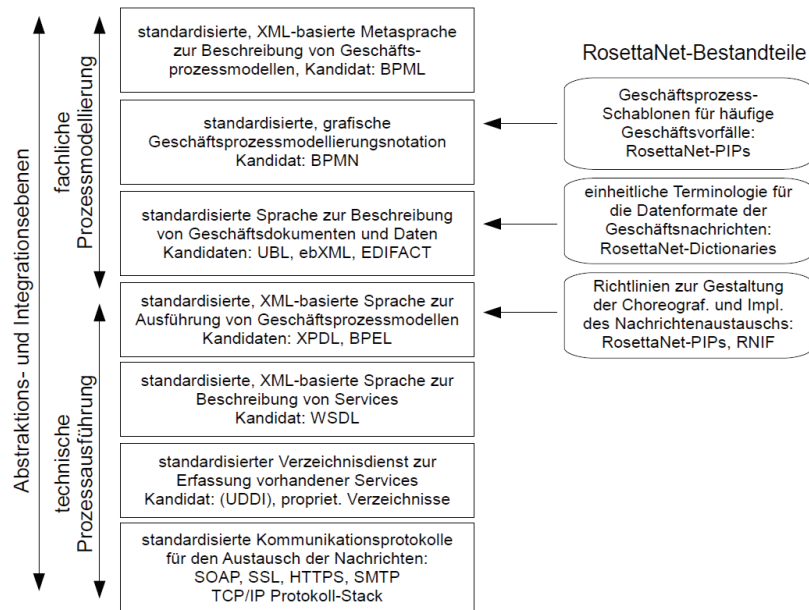


Abbildung 24: Abstraktions- und Integrationsebenen überbetrieblicher Prozessmodellierung

Für die Ausführung von übergreifenden Geschäftsprozessen müssen sich die beteiligten Unternehmen auf gemeinsame Schnittstellen einigen, nicht nur auf Ebene der Services, sondern auch den darüber liegenden Ebenen zur Modellierung, Speicherung und Ausführung der Geschäftsprozesse. Die Modellierung und Speicherung ist weniger kompliziert, als die spätere Ausführung, da standardisierte Datenformate für die Geschäftsprozessmodelle ein Import/Export in das Prozessmodellierungswerkzeug der Partner ermöglichen. Bei der Prozessausführung ergibt sich die Frage, ob der organisationsübergreifende Geschäftsprozess zentral durch einen Partner ausgeführt wird, oder ob es eine dezentrale Ausführung gibt. In dieser Arbeit konnten die Zusammenhänge der Service- und Prozessmodellierung zu Beginn grundlegend, später im Hauptteil für den unternehmensübergreifenden Bereich aufgezeigt werden. Das RosettaNet-Framework deckt bereits mehrere der dafür notwendigen Integrationsebenen ab, zeigt jedoch Schwächen in Bezug auf die Entwicklung komplexe Geschäftsprozesse mit mehreren Partnern beziehungsweise „über mehrere Partner hinweg“.

Die grundlegenden Problemfelder befinden sich jeweils zwischen den einzelnen Schichten der oben dargestellten Abstraktionsebenen (vertikale Integration), sowie der Kopplung der Unternehmen auf diesen Ebenen (horizontale Integration). In der nachfolgenden Aufzählung sind einige Schwerpunkte der horizontalen Integration genannt:

- Gemeinsames Geschäftsprozessmanagement der beteiligten Unternehmen auf fachlicher Ebene, mit gemeinsamer Modellierung, Optimierung, Anpassung und Überwachung der Geschäftsprozesse. (Siehe Kapitel 3.4.2)
- Verwendung einer gemeinsamen und standardisierten Geschäftsprozessmodellierungsnotation zur Verbesserung der gegenseitigen Akzeptanz; dabei Verwendung einer einheitlichen Sprache, z. B. englisch (siehe Kapitel 3.4.3 und 3.4.6)
- Verwendung einer gemeinsamen und standardisierten Sprache zur Beschreibung der Geschäftsnachrichten unter Zuhilfenahme von anerkannten Wörterbüchern für Geschäftsdaten
- Zentrale Prozessausführung durch einen Partner mit dem steuernden Workflow-Managementsystem, oder dezentrale Ausführung durch die Systeme der einzelnen Partner. In beiden Fällen ist die Verwendung der gleichen Prozessausführungssprache sinnvoll. (siehe Kapitel 3.4.4 und 3.4.5)
- Beschreibung der Services und Definition der Serviceschnittstellen in einer standardisierten Form, beispielsweise durch Verwendung von WSDL. Dies erleichtert die Veröffentlichung von Services und deren Verwendung durch die Geschäftspartner innerhalb einer Prozesskette.

- Erfassung der vorhandenen Services der beteiligten in einem zentralen System bei jedem Partner separat, oder einem unternehmensübergreifenden Serviceverzeichnis. Dadurch wird die Wiederverwendung und Auffindung der Services bei den Partnern erleichtert.
- Auf den niedrigsten, technischen Abstraktionsebenen sind bereits alle notwendigen Kommunikations- und Transportprotokolle vorhanden.

Für den Bereich der vertikalen Integration, ausgehend von den Anforderungen auf fachlicher Seite, und deren Umsetzung auf technischer Ebene, besteht die große Herausforderung darin, die auf der nächst höheren Ebene bereits erfassten Daten auf der darunter liegenden Ebene wiederzuverwenden, um den Implementierungsaufwand für elektronische Geschäftsprozesse zu senken. Da diese auf Daten aus den Anwendungssystemen der Unternehmen angewiesen sind, erscheinen Services als ein sinnvolles Instrument, die Daten innerhalb und außerhalb des Unternehmens über entsprechende Schnittstellen zur Verfügung zu stellen. Der bisher noch recht beschwerliche Weg vom fachlichen Prozessmodell zur fertigen Serviceimplementierung und Prozessausführung wurde im Kapitel 2.3.3 zu den Problemen bei der Integration von Geschäftsprozess- und Servicemodellen bereits beschrieben und fasst die wesentlichen Problemfelder und offenen Fragen zusammen. Exemplarisch wurden einige Teilbereiche dieser Integration im 4. Kapitel dieser Arbeit am Beispiel des Finanzdienstleistungssektors betrachtet. Zuerst wurden die Prozessschritte des Vertriebsprozesses daraufhin untersucht, welche Möglichkeiten das RosettaNet-Framework zur Unterstützung des elektronischen Nachrichtenaustauschs zwischen der Bank und den Geschäftskunden bietet, beziehungsweise welche durch RosettaNet standardisierten Geschäftsprozessvorfälle beim Vertriebsprozess ebenfalls auftreten. Im Ergebnis werden alle Prozessschritte durch RosettaNet unterstützt, die eine Interaktion mit dem Kunden erfordern. Als zweites wurden die Möglichkeiten untersucht, die sich bei der Unterstützung eines Teilprozesses des Vertriebs durch RosettaNet, WebServices und BPEL ergeben, in dem ein Interaktionsszenario zwischen dem Kunden und der Bank beschrieben wurde. Im letzten Kapitel 4.4 erfolgte die Modellierung des Vertriebsprozesses mit der Business Process Modeling Notation unter Berücksichtigung der zur Ausführung der Prozessschritte vorhandenen Services.

Zusammenfassend kann gesagt werden, dass verschiedene Ansätze zur Entwicklung organisationsübergreifender Geschäftsprozesse existieren, die einzelne Teilbereiche eines umfassenden B2B-Integrationsvorhabens abdecken. Jedoch gibt es bisher kein Rahmenwerk zur Entwicklung von Geschäftsprozessen mit einer so hohen Komplexität, welches alle oben genannten Integrationsebenen vollständig abdeckt und dadurch als Richtlinie für die Unternehmen zur Entwicklung ihrer eigenen, als auch der unternehmensübergreifenden, Geschäftsprozesse und einer prozessfähigen, serviceorientierten IT-Architektur dienen kann. Ein solches Rahmenwerk zu entwickeln, welches anerkannte Standards zur Modellierung, Speicherung, Übertragung und Ausführung von Geschäftsprozessen, unter Verwendung einheitlicher Wörterbücher, Geschäftsdokumenten- und Datenformate, sowie Nachrichten-Choreografiemuster, Serviceschnittstellenbeschreibungen und Serviceverzeichnissen enthält, bildet einen interessanten Forschungs- und Entwicklungsansatz für die Wirtschaftsinformatik. Mehrere der genannten Standards der OMG, OASIS, UN/CEFACT und anderer großer Unternehmen der IT-Branche könnten in dieses Rahmenwerk integriert werden, um die weltweite Akzeptanz des Rahmenwerks sicherzustellen.

Standardisierung auf der einen Seite und bessere vertikale Integration der Abstraktionsschichten zur Geschäftsprozessmodellierung und -ausführung auf der anderen Seite bilden den Grundpfeiler zur horizontalen Integration von Unternehmen auf fachlicher und technischer Ebene und damit zur Verbesserung des Business-IT-Alignments, als auch der unternehmensübergreifenden Integration. Der Einsatz von RosettaNet, Services oder einer Kombination aus beidem, wie im Abschnitt 3.5.7 beschrieben, kann dazu einen entscheidenden Beitrag leisten.

## Literaturverzeichnis

- Aalst, Wil van d. (Hrsg.); Bussler, Christoph (Hrsg.): Workshop Business Process Management BPM 2005, Springer-Verlag, Berlin, Heidelberg, 2005.
- Aalst, Wil van d. (Hrsg.); Dessel, Jörg (Hrsg.); Oberweis, Andreas (Hrsg.): Business Process Management: Models, Techniques, and Empirical Studies BPM 2000, Springer-Verlag, Berlin, Heidelberg, 2000.
- Abecker, Andreas; Hinckelmann, Knut; Maus, Heiko; Müller, Heinz-Jürgen: Geschäftsprozessorientiertes Wissensmanagement, Springer-Verlag, Berlin, Heidelberg, 2002.
- Abramowicz, Witold (Hrsg.); Maciaszek, Leszek (Hrsg.): Business Process and Service Computing: 1st International Working Conference on Business Process and Services, September 2007 Leipzig, Köllen Druck+Verlag, Bonn, 2007.
- Alonso, Gustavo; Casati, Fabio; Kuno, Harumi; Machiraju, Vijay: Web Services: Concepts, Architectures and Applications, Springer-Verlag, Berlin, Heidelberg, 2004.
- Alt, Rainer; Bernet, Beat; Zerndt, Thomas: Transformation von Banken: Praxis des In- und Outsourcing auf dem Weg zur Bank 2015, Springer-Verlag, Berlin, Heidelberg, 2009.
- Andonoff, Eric; Bouzguenda, Lotfi; Hanachi, Chihab: Specifying Web Workflow Services for Finding Partners in the Context of Loose Inter-organizational Workflow, in [van der Aalst und Bussler, 2005, S. 120ff.], Springer-Verlag, Berlin, Heidelberg, 2005.
- Bertram, Martin: Das Unternehmensmodell als Basis der Wiederverwendung bei der Geschäftsprozessmodellierung, in [Becker, 1996, S. 81ff.], Wiener Verlag, Wien, 1996.
- BPTG: Business Process Technology Group: BPMN Corner. Poster zur Notationsübersicht zu BPMN 1.1, Stand: 26.6.2008, [http://bpt.hpi.unipotsdam.de/pub/Public/BPMNCorner/BPMN1\\_1\\_Poster\\_DE.pdf](http://bpt.hpi.unipotsdam.de/pub/Public/BPMNCorner/BPMN1_1_Poster_DE.pdf), 2008.
- Brocke, Jan vom: Serviceorientierte Architekturen - SOA: Management und Controlling von Geschäftsprozessen, Franz Vahlen Verlag, München, 2008.
- Bussler, Christopher: The Role of B2B Protocols in Inter-Enterprise Process Execution, in: [Casati, 2001, S. 16ff.], Springer-Verlag, Berlin, Heidelberg, 2001.
- Bussler, Christoph: B2B Integration: Concepts and Architecture, Springer-Verlag, Berlin Heidelberg, 2003.
- Cannon, Edward: EDI Guide: A Step by Step Approach. Van Nostrand Reinhold, New York, 1993.
- Casati, Fabio (Hrsg.); Georgakopolous, Dimitrios (Hrsg.); Shan, Ming-Chen (Hrsg.): Technologies for E-Services: Second International Workshop, TES 2001, Springer-Verlag, Berlin, Heidelberg, 2001.
- DPAG: Sendungsverfolgung für Privatkunden der Deutschen Post AG. Einzelabfrage des Sendungsstatus am 29.11.2008, <https://www.deutschepost.de/sendungsstatus/bzl/sendung/>, 2008.
- Dressler, Sören: Business Process Outsourcing und Offshoring: Die moderne Ausgestaltung des Back-Office und mehr Effizienz im Unternehmen, Betriebswirtschaftlicher Verlag Dr. Th. Gabler, Wiesbaden, 2007.
- Dubray, Jean-Jacques: Where is BPEL 2.0 going? BPEL 2.0 Example for Order Management, [http://www.ebpml.org/bpel\\_2\\_0.htm](http://www.ebpml.org/bpel_2_0.htm), 2004.
- Dustar, Schaharam; Gall, Harald; Hauswirth, Manfred: Softwarearchitekturen für verteilte Systeme, Springer-Verlag, Berlin, Heidelberg, 2003.
- Dustdar, Scharaham: Towards Autonomic Processes and Services, in: [Abramowicz, 2007, S. 13ff.], Köllen Druck+Verlag, Bonn, 2007.

- Dustdar, Schahram; Tran, Huy; Zdun, Uwe: View-based and Model-driven Approach for Reducing the Development Complexity in Process-driven SOA, in: [Abramowicz, 2007, S. 105ff.], Köllen Druck+Verlag, Bonn, 2007.
- ebXML: Business Process and Business Information Analysis Overview v1.0, <http://www.ebxml.org/specs/bpOVER.pdf>, 2001.
- EBXMLa: Electronic Business using eXtensible Markup Language. About ebXML, <http://www.ebxml.org/geninfo.htm>, 2006.
- EBXMLb: Electronic Business using eXtensible Markup Language – ebXML Specifications. ebXML Deliverables, <http://www.ebxml.org/specs/index.htm>, 2006.
- Eckert, Clemens; Kohlmann, Falk; Mansfeld, Ken: Referenzprozesse und Servicelandkarten Vertrieb und Interbanken, Präsentation zum sourcing competence center 3, Elm, 4.2.2009.
- Emerald: Emerald Group Publishing Limited - RosettaNet Technical Report. RosettaNet Message Transfer, abgerufen am 21.12.2008, [http://www.emeraldinsight.com/\\_g/0291050508003.png](http://www.emeraldinsight.com/_g/0291050508003.png), 2008.
- Engels, Gregor; Hess, Andreas; Humm, Bernhard; Juwig, Oliver; Lohmann, Marc; Richter, Jan-Peter; Voÿ, Markus; Willkomm, Johannes: Quasar Enterprise: Anwendungslandschaften serviceorientiert gestalten, dpunkt Verlag, Heidelberg, 2008.
- Frei, Stefan; Reitbauer, Stefan: Sourcing-Modell im Zahlungsverkehr, Universitätsdokument, Universität St. Gallen, 2006.
- About RosettaNet Solutions: RosettaNet Solution System, <http://www.fujitsu.com/downloads/XML/rosettanet.pdf>, 2002.
- Gadatsch, Andreas: Grundkurs Geschäftsprozessmanagement: Methoden und Werkzeuge für die IT-Praxis, Friedrich Vieweg & Sohn Verlag, Wiesbaden, 2005.
- Galler, Jürgen: Vom Geschäftsprozessmodell zum Workflow-Modell, Betriebswirtschaftlicher Verlag Dr. Th. Gabler, Wiesbaden, 1997.
- Gavrilova, M. (Hrsg.): International Conference on Computational Science and Its Applications, ICCSA, 2006, Springer-Verlag, Berlin, Heidelberg, 2006.
- Gehring, H.: Betriebliche Anwendungssysteme: Prozessorientierte Gestaltung von Informationssystemen, Fernuniversität Hagen, Hagen, 1998.
- Greiner, Ulrike: Quality-Oriented Execution and Optimization of Cooperative Processes: Model and Algorithms, Druck der Universität Leipzig, Leipzig, 2006.
- Grover, Varun (Hrsg.); Markus, Lynee (Hrsg.): Business Process Transformation, M.E. Sharp Inc., New York, 2008.
- Grässle, Patrick; Baumann, Henriette; Baumann, Philippe: UML Projektorientiert: Geschäftsprozessmodellierung, IT-System-Spezifikation und Systemintegration mit der UML, Galileo Press, Bonn, 2000.
- Gruber, Jacek; Bzozowski, Tomasz: A Native Approach to Service Oriented Modeling and Implementation of Business Processes, in: [Abramowicz, 2007, S. 54ff.], Köllen Druck+Verlag, Bonn, 2007.
- Hagen, Richter C.; Stucky, Wolfried: Business-Process- und Workflow-Management: Prozessverarbeitung durch Prozess-Management, B.G. Teubner Verlag, Wiesbaden, 2004.
- Hangos, Katalin; Cameron, Ian: Process Modelling and Model Analysis, Academic Press, London, 2001.
- Hoffmann, Matthias; Reitbauer, Stefan: Vernetzungsmodelle für Banken, in [Alt u. a., 2009], Springer-Verlag, Berlin, Heidelberg, 2009.

## Literaturverzeichnis

- Hoyer, Volker; Bucherer, Eva; Schnabel, Florian: Collaborative e-Business Process Modelling: Transforming Private EPC to Public BPNM Business Process Models, S. 185-196, Springer-Verlag, Berlin, Heidelberg, 2007.
- Huvar, Martin; Falter, Timm; Fiedler, Thomas; Zubev, Alexander: Anwendungsentwicklung mit Enterprise SOA, Galileo Press, Bonn, 2008.
- Karle, Thomas; Oberweis, Andreas: Unterstützung von Kollaborationen im Rahmen der Softwareentwicklung auf Basis Service-orientierter Architekturen, in [Nüttgens, 2006, S. 77ff.], Köllen Druck+Verlag, Bonn, 2006.
- Khalaf, Ranai: From RosettaNet PIPs to BPEL Processes: A Three Level Approach for Business Protocols, in [van der Aalst und Bussler, 2005, S. 364ff.]. Springer-Verlag, Berlin, Heidelberg, 2005.
- Knolmayer, Gerhard; Endl, Rainer; Pfarherr, Marcel: Modeling Process and Workflow by Business Rules, in [van der Aalst u. a., 2000, S. 16ff.], Springer-Verlag, Berlin, Heidelberg, 2000.
- Korherr, Birgit: Business Process Modeling: Languages, Goals, and Variabiltities, VDM Verlag Dr. Müller, Saarbrücken, 2008.
- Krafzig, Dirk; Banke, Karl; Slama, Dirk: Enterprise SOA: Best Practises für Serviceorientierte Architekturen - Einführung, Umsetzung, Praxis, Redline GmbH, Heidelberg, 2007.
- Krcmar, Helmut; Böhmann, Tilo; Loser, Kai-Uwe: Modellierung von Prozessschnittstellen modularer Servicearchitekturen, in [Krcmar, 2005b, S. 167] . Physica Verlag, Heidelberg, 2005.
- Krcmar, Helmut (Hrsg.); Herrmann, Thomas (Hrsg.); Kleinbeck, Uwe (Hrsg.): Konzepte für das Service Engineering: Modularisierung, Prozessgestaltung und Produktivitätsmanagement. Physica Verlag, Heidelberg, 2005.
- Margolis, Ben; Sharpe, Joseph: SOA for the Business Developer: Concepts, BPEL, and SCA, MC Press, Big Sandy, Texas, 2007.
- Masak, Dieter: SOA? Serviceorientierung in Business und Software. Springer-Verlag, Berlin Heidelberg, 2007.
- Masud, Suhayl: Use RosettaNet-based Web Services, <http://www.ibm.com/developerworks/webservices/library/ws-rose1>, auch mit der Pfadendung /ws-rose2, /ws-rose3, /ws-rose4, 15.3.2003.
- Melano, Nuno; Pidd, Michael: Business Processes: Four Perspectives, in [Grover, 2008, S. 41ff.], M.E. Sharp Inc., New York, 2008.
- Meyer, Urs B.; Creux, Simone E.; Marin, Andrea K.: Grafische Methoden der Prozessanalyse: Für Design und Optimierung von Produktionssystemen, Hanser-Verlag, München, Wien, 2005.
- Microsoft: Microsoft Corporation - Rosetta-Net PIPs. Clusters and Segments, abgerufen am 20.12.2008, <http://www.microsoft.com/technet/prodtechnol/biztalk/2006/library/rosettanet/2f7e8db3-9ccb-403a-9fe7-58fbba3c4cb1.msp?mfr=true>, 2008.
- Nagl, Manfred; Westfechtel, Bernhard: Modelle, Werkzeuge und Infrastrukturen zur Unterstützung von Entwicklungsprozessen, Wiley-Vch Verlag, Weinheim, 2003.
- Nissen, Volker (Hrsg.); Petsch, Mathias (Hrsg.); Schorcht, Hagen (Hrsg.): Service-orientierte Architekturen: Chancen und Herausforderungen bei der Flexibilisierung und Integration von Unternehmensprozessen, Deutscher Universitätsverlag, Wiesbaden, 2007.

- OASIS: Organization for the Advancement of Structured Information Standards: Universal Business Language v2.0. Standard, 12 December 2006, 4.4. Ordering Process, <http://docs.oasis-open.org/ubl/os-UBL-2.0.zip>, 2006.
- OASIS: Organization for the Advancement of Structured Information Standards: Web Services Business Process Execution Language Version 2.0, OASIS Standard, <http://docs.oasis-open.org/wsbpel/2.0/OS/wsbpel-v2.0-OS.html>, Stand: 11.4.2007, 2007.
- OASIS: Organization for the Advancement of Structured Information Standards: Collaborative Processes and Agreements. Realizing Advantages of ebBP and CPP/A, <http://www.oasis-open.org/committees/download.php/26034/collaborating-processes-agreements-intro-110207.pdf>, Stand: 11.2.2007, 2007.
- OASIS: Organization for the Advancement of Structured Information Standards: Reference Architecture for Service Oriented Architecture 1.0. Public Review Draft 1, Authoritative PDF, <http://docs.oasis-open.org/soa-rm/soa-ra/v1.0/soa-ra-pr-01.pdf>, entnommen 23.04.2008, 2008.
- OASIS: Organization for the Advancement of Structured Information Standards: Universal Business Language TC. Frequently asked Questions, <http://www.oasisopen.org/committees/ubl/faq.php>, 2009.
- Oesterreich, Bernd; Weiss, Christian; Schröder, Claudia; Weilkiens, Tim; Lenhard, Alexander: Objektorientierte Geschäftsprozessmodellierung mit der UML, dpunkt Verlag, Heidelberg, 2004.
- OMG: Object Management Group: Business Process Modeling Notation (BPMN) 2.0. Request For Proposal, Stand: 5.6.2007, <http://www.bpmn.org/Documents/BPMN>
- OMG: Object Management Group: Business Process Modeling Notation, V1.1. OMG Available Specification, Stand: 17.1.2008, <http://www.omg.org/docs/formal/08-01-17.pdf>, 2008.
- Opincaru, Cristian: Service Oriented Security Architecture, in [Reichert, 2007, S. 61], Köllen Druck+Verlag, Bonn, 2007.
- ORACLE: Introduction to Rosetta-Net. What is RosettaNet?, abgerufen am 20.12.2008, <http://www.oracle.com/technology/pub/articles/dev2arch/2004/12/RosettaNet.html>, 2008.
- ORACLE: Introducing RosettaNet Solutions, abgerufen am 30.3.2009, [http://download.oracle.com/docs/cd/E13160\\_01/wli/docs10gr3/tpintro/rosettanel.html](http://download.oracle.com/docs/cd/E13160_01/wli/docs10gr3/tpintro/rosettanel.html), 2008.
- ORACLE: About Trading Partner Integration, abgerufen am 30.3.2009, [http://download.oracle.com/docs/cd/E13160\\_01/wli/docs10gr3/tpintro/intro.html](http://download.oracle.com/docs/cd/E13160_01/wli/docs10gr3/tpintro/intro.html), 2008.
- ORACLE: Tutorial: Building RosettaNet Solutions, abgerufen am 01.04.2009, [http://download.oracle.com/docs/cd/E13160\\_01/wli/docs10gr3/tptutorial/rosettanel.html](http://download.oracle.com/docs/cd/E13160_01/wli/docs10gr3/tptutorial/rosettanel.html), 2008.
- Overhage, Sven; Turowski, Klaus: Serviceorientierte Architekturen - Konzepte und methodische Herausforderungen, in [Nissen, 2007, S. 3.], Deutscher Universitätsverlag, Wiesbaden, 2007.
- Peter, Daniel; Wyss, Markus: Prozessmodellierung: Konzeptionelle Perspektive, aus einer Vorlesung der Hochschule für Wirtschaft Luzern. Universitätsvorlesung, [http://hswbscw1.hsw.fhz.ch/pub/bscw.cgi/d1458512/Handout\\_20LS04.pdf](http://hswbscw1.hsw.fhz.ch/pub/bscw.cgi/d1458512/Handout_20LS04.pdf), 2005.
- Picot, Arnold; Rohrbach, Peter: Organisatorische Aspekte von Workflow-Management-Systemen. Zeitschrift Information Management, 01/1995, S. 28-35, Saarbrücken, Hamburg, 1995.
- Reichert, Manfred (Hrsg.); Strecker, Stefan (Hrsg.); Turowski, Klaus (Hrsg.): Enterprise Modeling and Information Systems Architecture: Concepts and Applications, Beiträge des Workshops der GI-Fachgruppe, EMISA 2007, Köllen Druck+Verlag, Bonn, 2007.
- Rosenkranz, Friedrich: Geschäftsprozesse: Modell- und computergestützte Planung, Springer-Verlag, Berlin, Heidelberg, 2006.

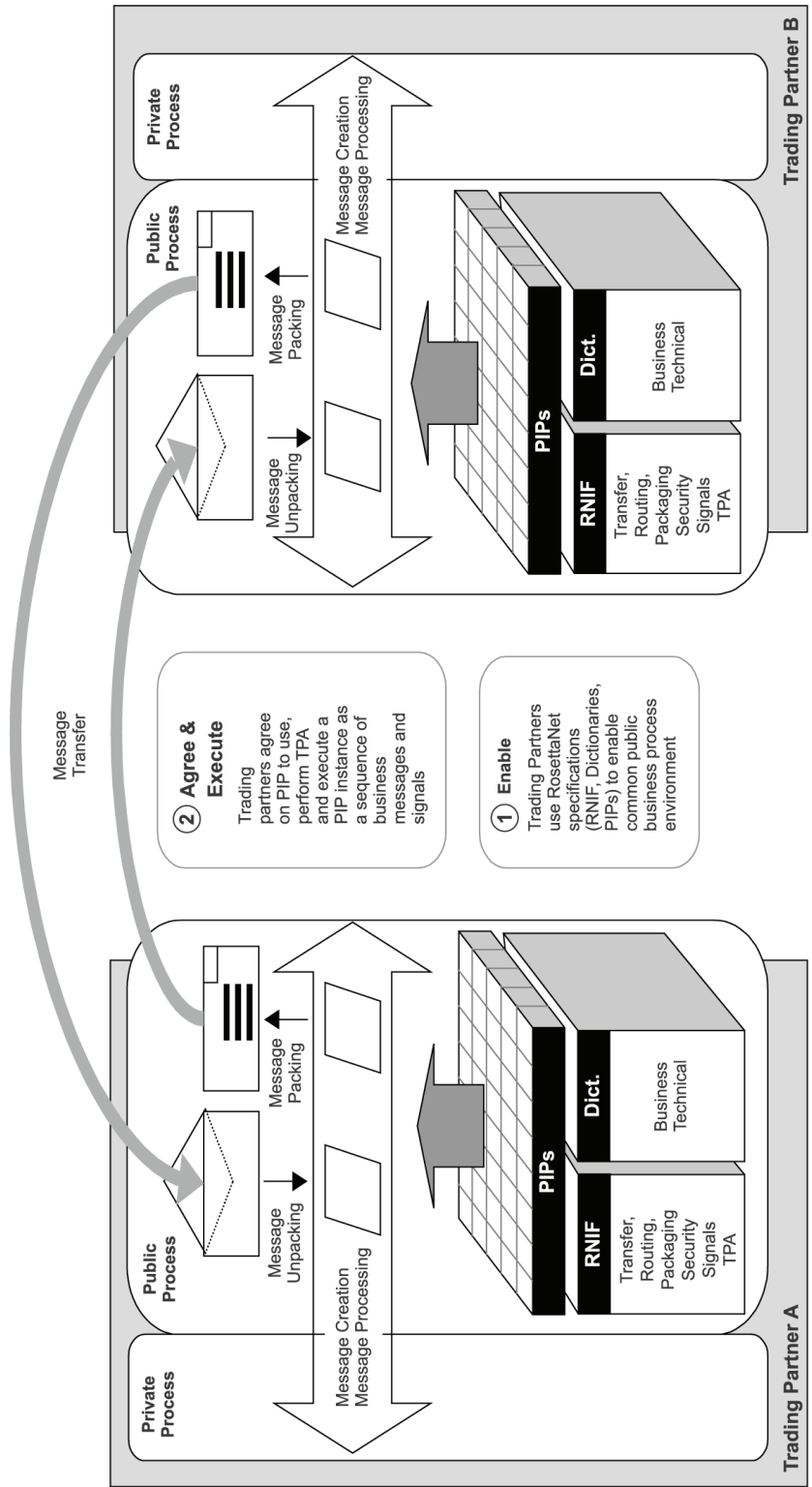
- RosettaNet: RosettaNet and Web Services: An Execution-Level View of RosettaNet and an Emerging Model for Application-Based B2B Commerce. RosettaNet Staff Members, [http://rosettanet.org/cms/export/sites/default/RosettaNet/Downloads/whitePapers/RosettaNet\\_Web\\_2003](http://rosettanet.org/cms/export/sites/default/RosettaNet/Downloads/whitePapers/RosettaNet_Web_2003).
- RosettaNet: About RosettaNet. Information, abgerufen am 20.12.2008, <http://www.rosettanet.org/cms/sites/RosettaNet/About/index.html>, 2008.
- RosettaNet: RosettaNet Overview Clusters, Segments, and PIPs, Version 02.06.00. Version 02.06.00, Stand: 09.01.2009, [http://members.rosettanet.org/dnn\\_rose/DocumentLibrary/tabid/2979/DMXModule/624/Command/2009](http://members.rosettanet.org/dnn_rose/DocumentLibrary/tabid/2979/DMXModule/624/Command/2009).
- Roukolainen, Toni; Kutvonen, Lea: Interoperability in Service-Based Communities, in [van der Aalst und Bussler, 2005, S. 317ff.], Springer-Verlag, Berlin, Heidelberg, 2005.
- Scheer, August-Wilhelm; Grieble, Oliver; Hans, Stephanie; Zang, Sven: Geschäftsprozessmanagement - The 2th Wave, in der Fachzeitschrift für Information Management and Consulting, S. 9-16. imc AG Verlag, 2002.
- Scheer, August-Wilhelm; Jost, Wolfram: Reorganisation von Geschäftsprozessen, in [Becker, 1996, S. 40ff.], Wiener Verlag, Wien, 1996.
- Schreiter, Torben; Laures, Guido: A Business Process-centered Approach for Modeling Enterprise Architectures, in [Nüttgens, 2006, S. 147ff.], Köllen Druck+Verlag, Bonn, 2006.
- Spath, Dieter; Weisbecker, Anette (Hrsg.): Business Process Management Tools 2008: Eine evaluierende Marktstudie zu aktuellen Werkzeugen, Fraunhofer IAO, Stuttgart, 2008.
- Staud, Josef: Geschäftsprozessanalyse: Ereignisgesteuerte Prozessketten und objektorientierte Geschäftsprozessmodellierung für betriebswirtschaftliche Standardsoftware, Springer-Verlag, Berlin, Heidelberg, 2006.
- Steinbuch, Pitter: Prozessorganisation - Business Reengineering – Beispiel R/3, Friedrich-Kiehl-Verlag, Ludwigshafen, 1998.
- Störrle, Harald; Glock, Wolfgang: Geschäftsprozessmodellierung für Service-orientierte Architekturen, in [Nissen, 2007, S. 77ff.], Deutscher Universitätsverlag, Wiesbaden, 2007.
- Tao, Aries; Yang, Jian: Supporting Differentiated Services with Configurable Business Processes, in: [Abramowicz, 2007, S. 194ff.], Köllen Druck+Verlag, Bonn, 2007.
- Thalbauer, Simon; Küng, Josef; Regner, Peter; Wiesinger, Thomas: Eine Integrationsplattform zur Verknüpfung von Geschäftsprozessen mit ITServices, in [Nüttgens, 2006, S. 63ff.], Köllen Druck+Verlag, Bonn, 2006.
- World Wide Web Consortium: Web Service Architecture - Service Model. Definition of Services, Stand: Februar 2004, <http://www.w3.org/2004/02/wsa/ServiceModel.owl>, 2004.
- WFMC: Workflow Management Coalition: Terminology and Glossary. Workflow Management Coalition Specification WFMC-TC-1011, <http://www.aiai.ed.ac.uk/project/wfmc/ARCHIVE/DOCS/glossary/glossary.html>, Juni, 1996.
- Yan, Zhixian; Mazzara, Manuel; Cimpian, Emilia; Urbanec, Alexander: Business Process Modeling: Classification and Perspectives, in: [Abramowicz, 2007, S. 222ff.], Köllen Druck+Verlag, Bonn, 2007.
- Yendluri, Prasad: RosettaNet Implementation Framework 2.0. RNIF Core 2.0, <http://www.rosettanet.org/cms/export/sites/default/RosettaNet/Downloads/whitePapers/RNIF2x1x1x1.pdf>, 2000.
- Yi, Yong G.; Park, Chiwoo; Kim, Minsoo: A Study on the Applications of BPM Systems for Implementation of RosettaNet based E-Logistics, in [Gavrilova, 2006, S. 697ff.], Springer-Verlag, Berlin, Heidelberg, 2006.



- Zhao, Xiaohui; Chengfei, Liu; Yang, Yun: An Organisational Perspective on Collaborative Business Processes, in [van der Aalst und Bussler, 2005, S. 17ff.], Springer-Verlag, Berlin, Heidelberg, 2005.
- Ziemann, Jörg; Matheis, Thomas; Freiheit, Jörn: Modeling of Cross-Organizational Business Processes: Current Methods and Standards, in [Reichert, 2007, S. 87ff.], Köllen Druck+Verlag, Bonn, 2007.

# Anhänge

## Anhang A: RosettaNet Message Transfer (Framework Overview)



Source: RosettaNet Technical Report (n.d.)

Abbildung 25: RosettaNet Message Transfer [Emerald, 2008]

Anhang B: BPMN 1.1 Notationselemente und Beispieldiagramm Vertrieb

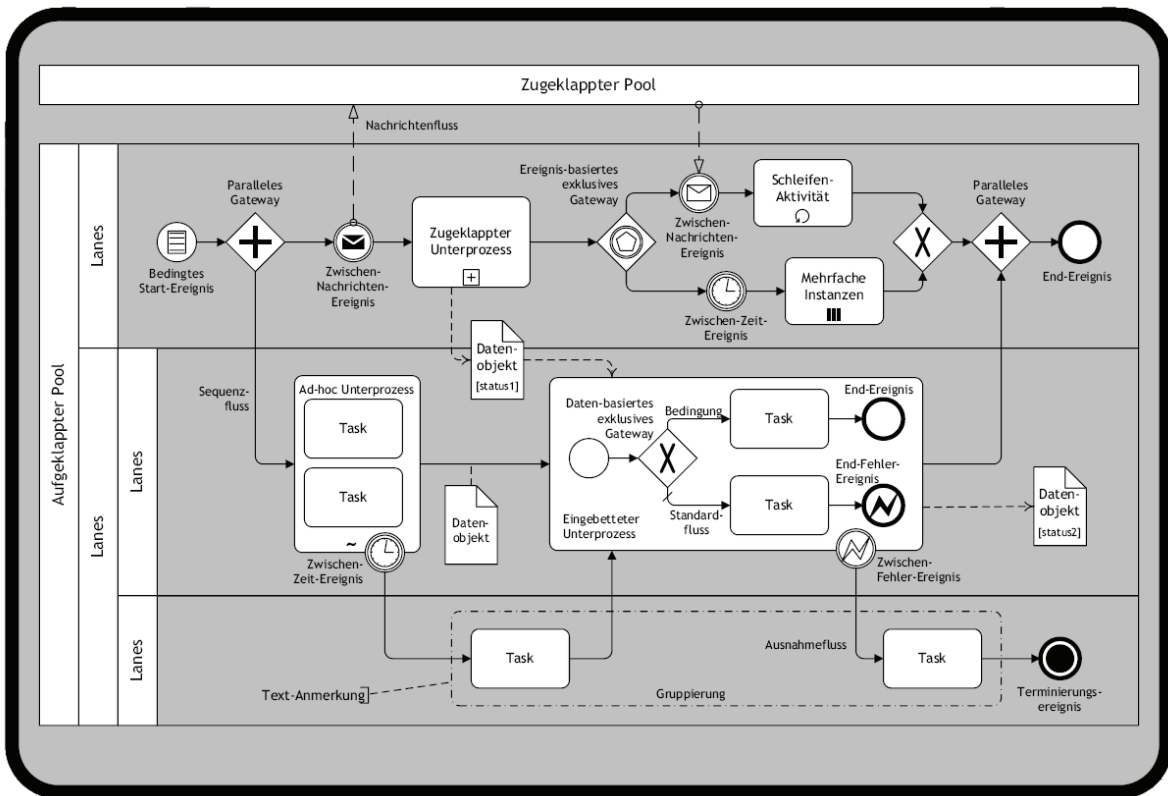


Abbildung 26: Notationselemente der BPMN 1.1 im Überblick [BPTG, 2008]

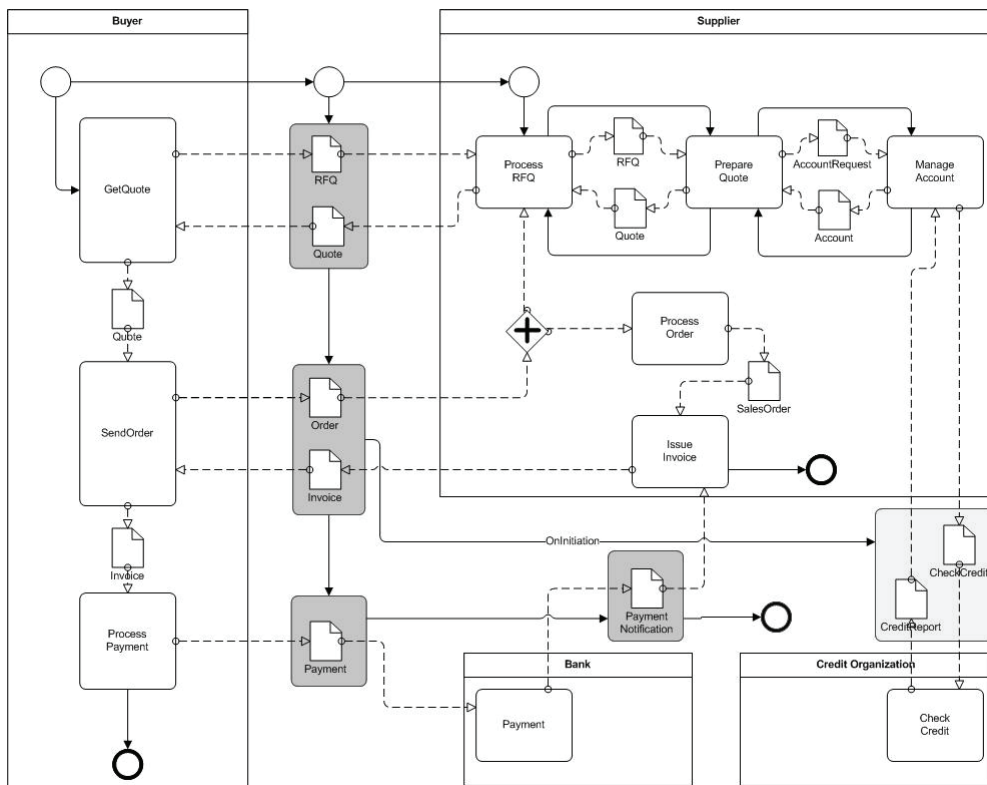


Abbildung 27: BPMN-Beispiel für den Vertrieb [Dubray, 2004]

## Anhänge

### Anhang C: RosettaNet Clusters and Segments

<b>Clusters</b>	<b>Segments</b>
0: RosettaNet Support	0A: Administrative 0C: Testing
1: Partner Product & Service Review	1A: Partner Review 1B: Product & Service Review
2: Product Information	2A: Preparation for Distribution 2B: Product Change Notification 2C: Product Design Information 2D: Collaborative Design & Engineering
3: Order Management	3A: Quote & Order Entry 3B: Transportation & Distribution 3C: Returns & Finance 3D: Product Configuration
4: Inventory Management	4A: Collaborative Forecasting 4B: Inventory Allocation 4C: Inventory Reporting 4D: Inventory Replenishment 4E: Sales Reporting 4F: Price Protection
5: Marketing Information Management	5A: Lead Opportunity Management 5B: Marketing Campaign Management
6: Service and Support	6A: Provide and Administer Warranties, Service Packages, and Contract Services 6B: Provide and Administer Asset Management (Merged with 6A) 6C: Technical Support and Service Management
7: Manufacturing	7A: Design Transfer 7B: Manage Manufacturing WO & WIP 7C: Distribute Manufacturing Information

Abbildung 28: RosettaNet Clusters und Segmente (englische Version)  
[Microsoft, 2008]

Anhang D: Bestellprozess bei der Universal Business Language

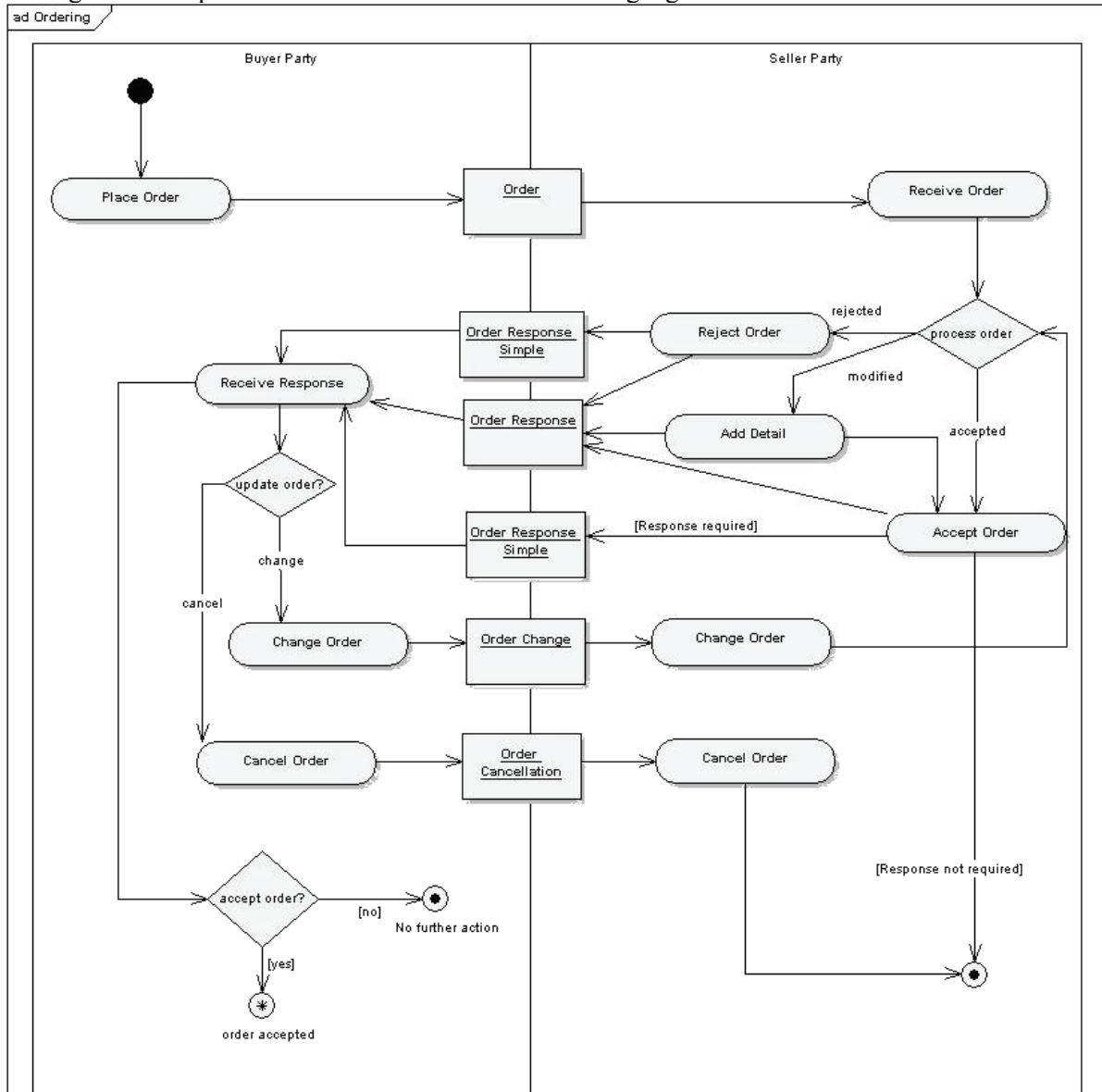


Abbildung 29: Ordering Process als Teil der Universal Business Language [OASIS, 2006]

## Anhänge

### Anhang E: Beschreibung des Referenzprozess Vertrieb

Prozesse	Definition
Kampagnenmanagement	Vertriebsunterstützung, aber kein Supportprozess gem. Bankmodell; Prozessauslösung hier durch aktives Vorgehen der Bank
Identifikation Zielgruppe	Identifikation einer Zielgruppe (Kunden und/oder Nichtkunden) nach bankspezifischen Kriterien (berufl. Situation, Vermögensverhältnisse, Alter etc.)
Kampagnenentwicklung & -planung	Entwicklung & -planung einer auf die identifizierte Zielgruppe abgestimmten Kampagne (Serienbrief, TV-Werbung, Plakate etc.)
Kampagnendurchführung	Durchführung der entwickelten und geplanten Kampagne
Leadmanagement	Vertriebsunterstützung, aber kein Supportprozess gem. Bankmodell
Kampagnendatenanalyse und –qualifikation	Analyse und Qualifikation der von der Kampagne zurücklaufenden Daten (Antworten, Nachfragen etc.)
Selektion Zielkunden	Selektion von einzelnen Zielkunden (Kunden und/ oder Nichtkunden; Bestandteil der Zielgruppe) auf Basis der Analyse/ Qualifikation
Zuordnung Kundenberater	Zuordnung der selektierten Zielkunden auf Kundenberater gem. Aufgabenbereich/ Zuständigkeit
gezielte Ansprache Zielkunde	Gezielte, personalisierte Ansprache der selektierten Zielkunden auf gewünschtem Kanal; bei Bedarf Vereinbarung eines Beratungstermins
Aktive Beratung	Kernbereich des Vertriebs; Prozessauslösung hier durch Kundenanfrage oder Input aus Betreuung
Informationsbeschaffung	Beschaffung von Informationen über Kunden, die ohne dessen Mitwirken zugänglich sind
Erstellung/Anpassung Kundenprofil	Erstellung (Neukunden)/Anpassung (Bestandskunden) des bankspezifischen Kundenprofils (demographische Daten, Überblick Vermögen & Verbindlichkeiten, Planung für Zukunft etc.)
Präzisierung Kundenbedürfnisse	Präzisierung der geäußerten Kundenbedürfnisse durch Aufdeckung von latenten Bedürfnissen und cross-selling-Ansätzen auf Basis des Kundenprofils und weiterer Fragen
Einholung Kundenunterlagen	Einholung von Kundenunterlagen, die das Kundenprofil bestätigen/ ergänzen/ausbauen (Übersicht Vermögen & Verbindlichkeiten, Einkommensteuerbescheid, Vertragskopien etc.)
Support Kompetenzzentrum	Fachliche Unterstützung von Kompetenzzentren der Bank bei speziellen Fachgebieten (Vermögensanlage im Ausland, Immobilienkauf etc.)

Prozesse	Definition
Ermittlung Kundengeschäftspotential und –risiko	Ermittlung des Kundengeschäftspotentials und -risikos auf Basis der vorangegangenen Schritte. Im Kreditgeschäft insb. Prüfung der Kreditfähigkeit (Rating) des Antragsstellers, wobei die Kreditportfoliopolitik und regulatorische Auflagen (z. B. Bankgesetz, Geldwäschereigesetz, Vereinbarung über die Standesregeln zur Sorgfaltspflicht der Banken (VSB), EBK-Rundschreiben) Berücksichtigung finden und die hinterlegten Sicherheiten bewertet werden.
Ablehnung durch Bank	Ablehnung von weiterer Beratung für gesamten Kunden oder einzelne Produkte bei Feststellung von zu geringem Kundengeschäftspotential und/oder zu hohem Kundengeschäftsrisiko
Empfehlung Produkt/ Lösung	Empfehlung eines Produkts/einer Produktklasse/eines Lösungspakets, das identifizierte Bedürfnisse bestmöglich deckt
Dokumentation	Dokumentation aller relevanten Tätigkeiten (bspw. zur Sicherstellung regulatorischer Vorgaben); ggf. Gegenzeichnung durch Kunden
Offerte	Kernbereich des Vertriebs
Angebotsgestaltung	Konkrete Ausgestaltung eines Produkt(paket)s mit den dazugehörigen Konditionen
Offerteneinreichung	Unterbreitung des Angebots an Kunden
Nachverhandlung und Anpassung	Nachverhandlung und ggf. Anpassung der vorgelegten Offerte (z. B. Veränderung des Produkt(paket)s und/ oder der Konditionen)
Annahme/ Ablehnung durch Kunde	Produkt- bzw. Lösungsannahme oder -ablehnung durch Kunden
Nachreichung Kundenunterlagen	Nachreichung noch fehlender Kundenunterlagen
Dokumentation	Dokumentation aller relevanten Tätigkeiten (bspw. zur Sicherstellung regulatorischer Vorgaben); ggf. Gegenzeichnung durch Kunden
Abschluss	Kernbereich des Vertriebs
Erstellung Vertrag	Vertragserstellung auf Basis der eingereichten bzw. nachverhandelten Offerte
interne Prüfung	Prüfung des zwischen Kundenberater und Kunde vereinbarten Vertrags (Vollständigkeit der Unterlagen, Einhaltung der regulatorischen Vorgaben, interne Zielvorgaben, Kompetenzen etc.)
Unterzeichnung	Unterzeichnung des Vertrags durch beide Seiten
Initialisierung Leistungserbringung	interne Weiterleitung der Vertragsunterlagen an entsprechende Fachabteilung und Auslösung des Folgeprozesses (Zahlen, Anlegen oder Finanzieren)
Prozesse	Definition

## Anhänge

Dokumentation	Dokumentation aller relevanten Tätigkeiten (bspw. zur Sicherstellung regulatorischer Vorgaben); ggf. Gegenzeichnung durch Kunden
Betreuung	bedarfsunabhängig
Sicherung Kundenzufriedenheit	Planung, Durchführung und Überwachung aller Maßnahmen zur Sicherung der Zufriedenheit des Kunden und Stabilisierung der Kundenbeziehungen; Bearbeitung von Kundenbeschwerden (produktabhängig und -unabhängig)
fortlaufende Beratung	Informierung und ggf. erneute aktive Beratung des Kunden bei veränderten Marktbedingungen, neuen Produkten, auslaufenden Verträgen etc.
Pflege Kundenkontakt & -profil	Produktübergreifende Pflege des Kundenkontakts (direkt durch Geschenke, Weihnachtskarten etc.) und -profils (intern bei Umzug, Geburt Kinder etc.)
Ausbau Kundenbeziehung	Ausbau der Kundenbeziehung durch Vertiefung bzw. Erweiterung der in Anspruch genommenen Produktkategorien
Beendigung Kundenbeziehung	Beendigung der Kundenbeziehung durch Konto-/Depot-/Kundenstammauflösung
Dokumentation	Dokumentation aller relevanten Tätigkeiten (bspw. zur Sicherstellung regulatorischer Vorgaben); ggf. Gegenzeichnung durch Kunden

Tabelle 10: Beschreibung des Referenzprozess Vertrieb  
[Eckert u. a., 2009]



## Anhang F: PIP-Spezifikationen für den Dienstleistungsbereich

Diese Tabelle enthält nicht alle durch RosettaNet spezifizierte PIPs, da einige nicht offiziell freigegeben sind.

PIP	Definition
Cluster 0: RosettaNet Support	Dieser Cluster enthält PIPs zur Prüfung der technischen Funktionalität der Verbindungen.
0A1: Notification of Failure	Benachrichtigung bei Fehlern, z. B. fehlenden Dokumenten oder Informationen.
0C2: Asynchronous Test Request / Confirmation	Prüfung der asynchronen Verbindung zum Geschäftspartner, indem eine Testnachricht gesendet und deren fehlerfreier Versand gemeldet wird oder ggf. ein Fehler.
0C4: Synchronous Test Query / Response	Prüfung der synchronen Verbindung zum Geschäftspartner, indem eine Testnachricht gesendet und deren fehlerfreier Versand gemeldet wird, oder ggf. ein Fehler.
Cluster 1: Partner Product and Service Review	Dieser Cluster unterstützt die Sammlung, Aktualisierung und Verteilung von Informationen zwischen den Partnern, insbesondere Produktinformationen.
Segment 1A: Partner Review	Dieses Segment unterstützt den Austausch von Informationen zwischen Geschäftspartnern, wie zum Beispiel Liefer- und Rechnungsadresse, Kundeninformationen, Kontaktdaten usw.
1A1: Request Account Setup	Sobald die Partnerschaft beschlossen ist, können relevante Informationen (Ort, Kontaktdaten usw.) elektronisch ausgetauscht werden. Dafür ist die Einrichtung eines Accounts (nicht im Sinne von Bankkonto) notwendig. Dies ist sowohl für Privatkunden, als auch Geschäftskunden erforderlich.
1A2: Maintain Account	Pflege/Wartung des Accounts. Tritt ein, wenn die im Account gespeicherten Informationen aktualisiert werden müssen, beispielsweise eine Lieferadresse oder Rechnungsadresse. Setzt 1A1 voraus.
Cluster 2: Product Introduction	Dieser Cluster unterstützt die Verteilung und periodische Aktualisierung von Produktinformationen- und details.
Segment 2A: Preparation of Distribution	Dieses Segment unterstützt die Verteilung von Produktinformationen.
2A1: Distribute Product Catalog Information	Veröffentlichung von Produktinformationen in einem Katalog, sowie deren Aktualisierung, Ergänzung und Veröffentlichung. Siehe auch 2B1. Für den Dienstleistungssektor verwendbar, wenn die verfügbaren Produkte (Konten, Fonds, Sparbücher, Kredite) in einem Katalog erfasst werden sollen, der vom Kunden abrufbar ist. (Dies ist jedoch nur für weitgehend standardisierte Produkte mit geringen Konfigurationsmöglichkeiten sinnvoll.)
2A3: Query Marketing Information	Abruf von Produktinformationen vor dem Kauf. Der Anbieter hat die Möglichkeit ein Angebot zusammenzustellen und dies dem Kunden zurück zu senden. Siehe auch 2B2.
2A4: Query Sales Promotion & Rebate Information	Abruf von Informationen über Preisnachlässe oder besondere Verkaufsbedingungen für ein spezielles Produkt.. Der Anbieter erstellt für ein spezifisches Produkt ein Preisangebot. Siehe auch: 2B3.
2A6: Query Product Life-cycle Information	Abfrage des Zustands eines Produktes zu einem gegebenen Zeitpunkt. Ermöglicht es dem Kunden, Informationen über ein gekauftes oder verfügbares Produkt einzuholen. Bezieht sich ursprünglich auf den Zustand eines Produktionsgutes während der Fertigung, also den Fertigungsfortschritt.

PIP	Definition
2A7: Query Product Discontinuation Information	Abfrage von Verfügbarkeitsinformationen eines speziellen Produktes, beispielsweise wann und wie lange ein bestimmtes Produkt noch verfügbar ist.
2A13: Distribute Material Composition Information	Ermöglicht es dem Anbieter, Informationen über die Zusammensetzung eines Produktes an den Kunden zu übermitteln. Dies bezieht sich ursprünglich auf physische Materialien und Komponenten, die zu einem komplexen Gegenstand zusammengesetzt werden. Dieses Konzept lässt sich beispielsweise auf die Zusammensetzung eines Aktien-Fonds aus verschiedenen Aktienpaketen übertragen, wobei der Kunde über die Zusammensetzung des Fonds benachrichtigt wird. Das Gegenstück bildet 2A15.
2A15: Request Material Composition Information	Ermöglicht es dem Kunden die Zusammensetzung eines Produktes beim Verkäufer zu erfragen. Siehe 2A13.
Segment 2B: Product Change Notifications	Dieses Segment unterstützt die Aktualisierung von Produktinformationen.
2B1: Change Basic Product Information	Ermöglicht es dem Anbieter die Änderung von Produktinformationen an den Kunden zu melden.
2B2: Change Marketing Information	Ermöglicht es dem Anbieter die Änderung von Verkaufsinformationen an den Kunden zu melden.
2B3: Change Sales Promotion & Rebate Informationen	Ermöglicht es dem Anbieter die Änderung von Preisinformationen und Rabatten an den Kunden zu melden.
2B5: Change Product Lifecycle Information	Ermöglicht es dem Anbieter die Änderung am Zustand eines Produktes an den Kunden zu melden. Diese PIP könnte dazu verwendet werden, einen Kunden über die Änderung des Kurses seiner Aktien zu informieren, um ihn gegebenenfalls zum Kauf oder Verkauf anzuregen.
2B7: Notify of Product Change	Mit diesem PIP kann der Anbieter den Kunden auf Änderungen von Produkten oder deren Ende (Vertragsende, Ende der Gültigkeitsdauer) aufmerksam machen.
2B8: Notify of Product Change Query	Mit diesem PIP kann der Kunde den Anbieter nach den Details einer Produktänderung befragen, zu denen der Verkäufer Auskunft gibt.
2B9: Notify of Product Change Acceptance	Der Kunde benachrichtigt den Anbieter über seine Zustimmung oder Ablehnung zu den Änderungen der Produkteigenschaften.
Cluster 3: Order Management	Dieser Cluster unterstützt die Verwaltung von Kundenbestellungen.
Segment 3A: Quote and Order Entry	Dieses Segment ermöglicht den Partnern den Austausch von Preis- und Verfügbarkeitsinformationen, Bestellungen, Bestellanfragen und Warenkörben.
3A1: Request Quote	Ermöglicht es dem Kunden eine Preisanfrage für ein oder mehrere Produkte an den Anbieter zu senden.
3A2: Request Price and Availability	Ermöglicht es dem Kunden eine Preis- und Verfügbarkeitsanfrage für ein oder mehrere Produkte an den Anbieter zu senden. Diese PIP ist weniger formal als 3A1 und dient eher dazu Preisinformationen von mehreren Anbietern einzuholen, als einen tatsächlichen Kauf bei einem speziellen Anbieter einzuleiten.
3A3: Request Shopping Cart Transfer	Mit diesem PIP ist der Kunde in der Lage, einen Warenkorb zum Anbieter zu senden. Der Warenkorb enthält dabei Produkte mit einem Preis und einer Menge. Der Anbieter kann die Produkte über einen Katalog zur Verfügung stellen. Beim Vertrieb im Bankensektor könnte dies ein Katalog mit den angebotenen Aktien eines Brokers sein, bei dem sich der Kunde die Produkte auswählen kann.

PIP	Definition
3A4: Request Purchase Order	Wird üblicherweise nach 3A1, 3A2 und 3A3 eingeleitet und dient dem Kunden zum Versand einer Bestellung an den Anbieter. Der Anbieter prüft anschließend die Bestellung und sendet dem Kunden entweder eine Bestätigung oder Ablehnung.
3A5: Query Order Status	Ermöglicht es dem Kunden, den Status der Bestellung einzusehen, um diese gegebenenfalls zu Ändern.
3A6: Distribute Order Status	Ermöglicht es dem Anbieter, den Kunden über den Bestellungsstatus zu informieren. Im Bankensektor könnte diese PIP bei erfolgreicher Bereitstellung eines neuen Online-Kontos oder Zugangs verwendet werden, in dem dem Kunden alle nötigen Dokumente und Zugangsdaten (PIN, TAN-Listen) zugesendet werden.
3A7: Notify of Purchase Order Update	Ermöglicht es dem Anbieter, den Kunden über Änderungen der Bestellung, zum Beispiel mangelnde Verfügbarkeit, zu benachrichtigen. Siehe auch 3A8.
3A8: Request Purchase Order Change	Ermöglicht es dem Kunden, auf Änderungen der Bestellung (3A7) zu reagieren. Der Anbieter kann auf die Reaktion des Kunden reagieren und die aktualisierte Bestellung entweder ablehnen oder bestätigen.
3A9: Request Purchase Order Cancellation	Ermöglicht es dem Kunden, eine Bestellung zu stornieren. Der Anbieter kann auf die Stornierung des Kunden reagieren und diese entweder ablehnen oder bestätigen.
3A10: Notify of Quote Acknowledgement	Ermöglicht es dem Kunden, den Empfang einer Bestellung zu bestätigen.
3A14: Distribute Planned Order	Ermöglicht es dem Kunden, den Anbieter auf potentielle, zukünftige Käufe aufmerksam zu machen, um sein Interesse an weiteren Produkten zu bekunden.
Segment 3B: Transportation and Distribution	Dieses Segment findet für den Dienstleistungssektor nur bedingt Anwendung, weil alle vorhandenen PIPs auf den physischen Warentransport der Produkte ausgerichtet sind. Das englische Wort „Distribution“ bezieht sich in diesem Fall nur auf die physische Warenverteilung und die damit zusammenhängenden, logistischen Prozesse. Gegebenenfalls ließen sich die PIPs auf die Übertragung von Dokumenten (z. B. Rechnungen, papiergebundene Kontoauszüge, geänderte AGB's, Werbeprospekte und der gleichen) anwenden, jedoch wird dies hier nicht weiter betrachtet.
Segment 3C: Returns and Finance	Dieses Segment unterstützt den Versand von Rechnungen, die Ausführung von Zahlungen, den Ausgleich von Krediten, sowie die Rücksendung von Produkten und deren finanzielle Auswirkungen.
3C1: Return Product	Ermöglicht dem Kunden, ein Produkt zurück zu geben. Zu diesem Zweck erstellt der Kunde einen Widerruf, den er an den Anbieter sendet. Der Anbieter reagiert auf den Widerruf und sendet dem Kunden alle Informationen zur Rücksendung der Waren. Für den Vertrieb im Bankensektor wäre die Verwendung dieser PIP zur Auflösung von Verträgen über Konten, Sparbücher oder Fonds möglich.
3C3: Notify of Invoice	Ermöglicht es dem Anbieter, dem Kunden eine Rechnung zuzusenden. Voraussetzung bildet der PIP 3A4 bei dem der Verkäufer die Ware an den Kunden ausliefert.
3C4: Notify of Invoice Reject	Ermöglicht es dem Kunden, die Rechnung des Anbieter zurückzuweisen, z. B. wenn ein Fehler vorliegt oder eine andere Finanzierungsva-

## Anhänge

	riante erwünscht ist. Vorr. bildet 3A4 und 3C3.
<b>PIP</b>	<b>Definition</b>
3C5: Notify of Billing Statement	Ermöglicht es dem Anbieter, den Kunden über den aktuellen Rechnungsstatus zu unterrichten, beispielsweise periodisch jede Woche, jeden Monat, jedes Quartal. Für den Bankensektor könnte dies zum Beispiel für den Versand der Kontoauszüge an die Kunden verwendet werden.
3C7: Notify of Self-Billing Invoice	Ermöglicht es dem Kunden, einen Gutschein/Gutschrift beim Anbieter einzulösen.
Cluster 4: Inventory Management	Das gesamte Cluster zum Bestandsmanagement und Bestandsverwaltung lässt sich nur unzureichend auf den Dienstleistungssektor, insbesondere den Bankensektor übertragen. Gemäß den hier definierten PIP-Spezifikationen bezieht sich der Bestand immer auf physisch verfügbare Güter und Verbrauchsgegenstände in einem Lager, die innerhalb einer Produktions- bzw. Wertschöpfungskette verwendet werden. Sehr entfernt ließe sich der Bestand an Aktien bei einem Broker oder einer Aktiengesellschaft hierfür anwenden, jedoch wird dies nicht weiter betrachtet.
Cluster 5: Marketing Information Management	Dieses Cluster enthält PIPs zum Austausch von Produktinformationen und Angeboten.
5C1: Distribute Product List	Ermöglicht es dem Anbieter, eine Liste aller verfügbaren Produkte zu einem Kunden zu senden. Dies entspricht einem Werbeangebot mit Produktvorschlägen, die für den Kunden interessant sein könnten.
Cluster 5: Service and Support	Dieses Cluster enthält PIPs für den Support nach dem Verkauf, das Angebot von Service-Leistungen, sowie das Anlagenmanagement.
6A1: Notify of Service Contract Request	Ermöglicht es dem Kunden, eine Änderung der Servicevereinbarungen beim Anbieter zu beantragen. Siehe auch 6A2.
6A2: Notify of Service Contract Reply	Ermöglicht es dem Anbieter auf den Antrag zur Änderung der Servicevereinbarungen des Kunden zu reagieren.
Cluster 7: Manufacturing	Dieses Cluster enthält PIPs zur Steuerung und Kontrolle der Produktionsanlagen und findet für den Dienstleistungssektor keine Anwendung.
Cluster 8	nicht öffentlich verfügbar
Cluster 9	nicht öffentlich verfügbar

Tabelle 11: Übersetzung der RosettaNet-Cluster  
Übersetzung aus [RosettaNet, 2009]

## Anhang G: Kombination der PIP-Spezifikationen mit dem Vertriebsprozess

Prozesse	Definition und unterstützende PIPs
Kontakt	
Kontaktplanung	Interner Prozess, keine PIP-Unterstützung
Kontaktvorbereitung	Interner Prozess, keine PIP-Unterstützung
Kontaktaufnahme	Zuordnung der selektierten Zielkunden auf Kundenberater: 1A2: Maintain Account (Aktualisierung der vorh. Kundeninformationen: Zuweisung des Beraters) Gezielte, personalisierte Ansprache der Zielkunden auf gewünschtem Kanal 2B1: Change Basic Product Information 2B2: Change Marketing Information 2B3: Change Sales Promotion & Rebate Information 2B5: Change Product Lifecycle Information 2B7: Notify of Product Change 2B8: Notify of Product Change Query (gehört zu 2B7) 5C1: Distribute Product List (Werbeangebot mit Produktvorschlägen an den/die Zielkunden)
Beratung	
Informationsbeschaffung	laut Referenzprozess rein interner Prozess, daher: keine PIP-Unterstützung
Erstellung/Anpassung Kundenprofil	Erstellung (Neukunden)/Anpassung (Bestandskunden) des Kundenprofils: 1A1: Request Account Setup 1A2: Maintain Account
Präzisierung Kundenbedürfnisse	Präzisierung der geäußerten Kundenbedürfnisse: 2A3: Query Marketing Information 2A4: Query Sales Promotion & Rebate Information 2A6: Query Product Lifecycle Information 2A7: Query Product Discontinuation Information 2A13: Distribute Material Composition Information (Beschreibung der Zusammensetzung/Konfiguration eines Produktes) 2A15: Request Material Composition Information (Anfrage der Zusammensetzung/Konfiguration eines Produktes)
Einholung Kundenunterlagen	Einholung von Kundenunterlagen, die das Kundenprofil bestätigen/ergänzen/ausbauen 1A2: Maintain Account (Aktualisierung/Erweiterung des Kundenprofils)
Ermittlung Kundengeschäftspotential und –risiko	Interner Prozess, keine PIP-Unterstützung
Ablehnung durch Bank	Ablehnung von weiterer Beratung für gesamten Kunden oder einzelne Produkte 2B7: Notify of Product Change (Kunde wird darüber benachrichtigt, dass ein Produkt nicht mehr zur Verfügung steht)
Empfehlung Produkt/ Lösung	Empfehlung eines Produkts/einer Produktklasse/eines Lösungspaketes 2A1: Distribute Product Catalog Information (Versand des aktuellen Produktkatalogs/Produktvorschlags an die Kunden) 5C1: Distribute Product List

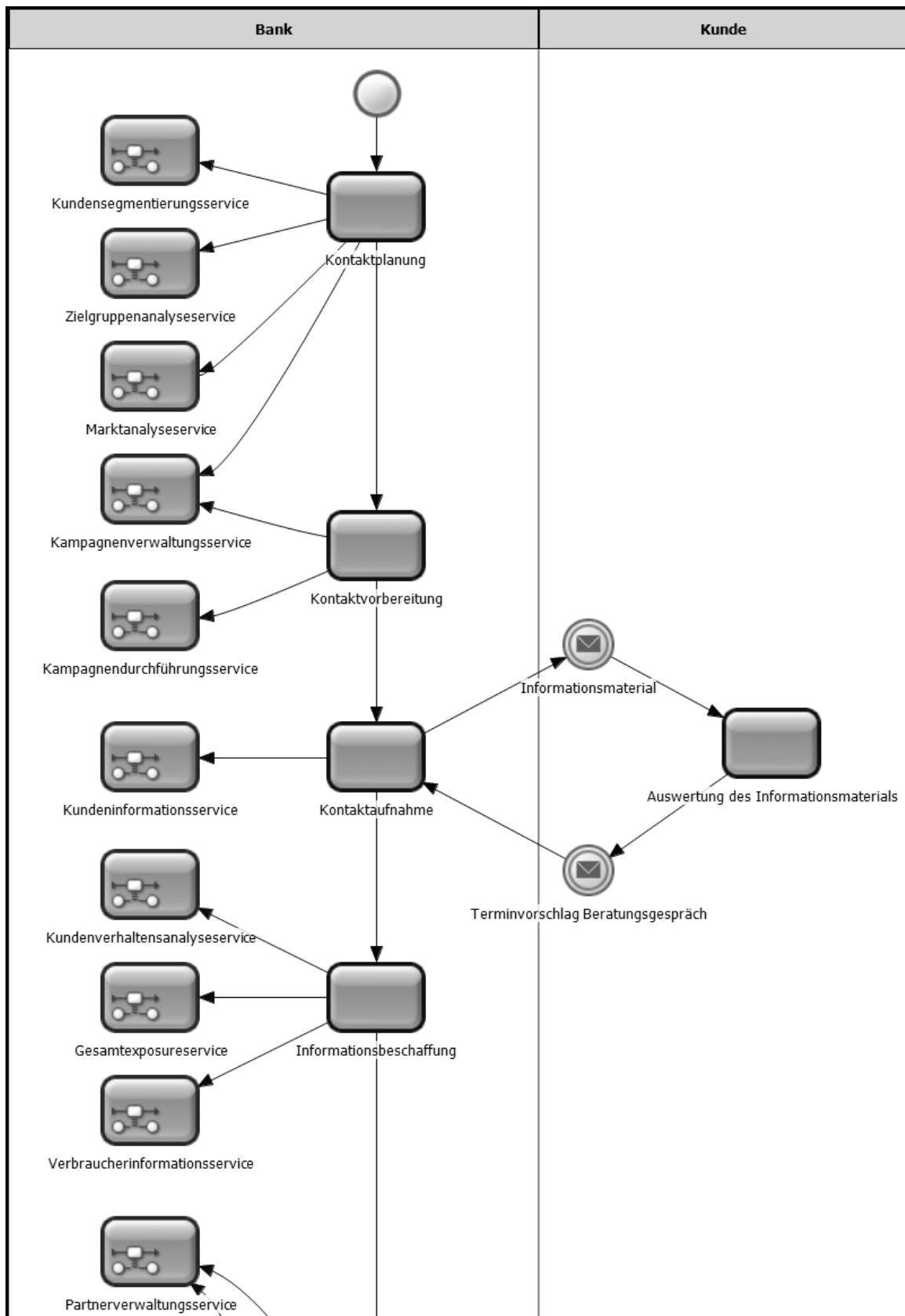
## Anhänge

	(Werbeangebot mit Produktvorschlägen an die Kunden)
Prozesse	Definition und unterstützende PIPs
Offerte	
Offertenstellung	Unterbreitung des Angebots an Kunden 2B1: Change Basic Product Information 2B2: Change Marketing Information 2B3: Change Sales Promotion & Rebate Information 2B4: Change Product Lifecycle Information
Nachverhandlung und Anpassung	Nachreichung noch fehlender Kundenunterlagen 1A2: Maintain Account (Ergänzung fehlender Kundeninformationen) Nachverhandlung und ggf. Anpassung der vorgelegten Offerte 2B8: Notify of Product Change Query (Benachrichtigung bei Vorschlägen) 2B9: Notify of Product Change Acceptance (Akzeptanz der Vorschläge) 3A3: Request Shopping Cart Transfer (Ausgestaltung des Warenkorb durch den Kunden)
Annahme/ Ablehnung durch Kunde	Produkt- bzw. Lösungsannahme oder -ablehnung durch Kunden implizit wieder 2B8, 2B9 wie bei der Nachverhandlung
Abschluss	
Erstellung Vertrag	Interner Prozess, keine PIP-Unterstützung
interne Prüfung	Interner Prozess, keine PIP-Unterstützung
Unterzeichnung	Unterzeichnung des Vertrags durch beide Seiten 1A1: Request Account Setup (Einrichtung des Kundenprofils, sofern nicht bereits eher geschehen) 3A4: Request Purchase Order (Versand der Bestellung) 3A5: Query Order Status (Abfrage des Bestellungsstatus durch Kunden)
Initialisierung Leistungserbringung	Auslösung der Folgeprozesse 3A6: Distribute Order Status (Bestätigung des Bestellungsstatus durch Anbieter) 3C3: Notify of Invoice (Versand der Rechnung an den Kunden) 3C4: Notify of Invoice Reject (bei fehlerhafter Rechnung) 3C7: Notify of Self-Billing Invoice (Bekanntgabe des Rechnungsstatus an den Kunden) 3A7: Notify of Purchase Order Update (Änderung der Bestellung durch Anbieter) 3A8: Request Purchase Order Change (Änderung der Bestellung durch den Kunden) 3A9: Request Purchase Order Cancellation (Stornierung) 3A10: Notify of Quote Acknowledgement (Empfangsbestätigung) 3C5: Notify of Billing Statement (Bekanntgabe des Rechnungsstatus)
Pflege (Betreuung)	
Sicherung Kundenzufriedenheit	Produktübergreifende Pflege des Kundenkontakts und -profils 1A2: Maintain Account (Aktualisierung/Erweiterung des Kundenprofils) Maßnahmen zur Sicherung der Zufriedenheit des Kunden 3C1: Return Product (Rückgabe des Produktes durch den Kunden) Informierung und ggf. erneute aktive Beratung des 2A13: Distribute Material Composition Information

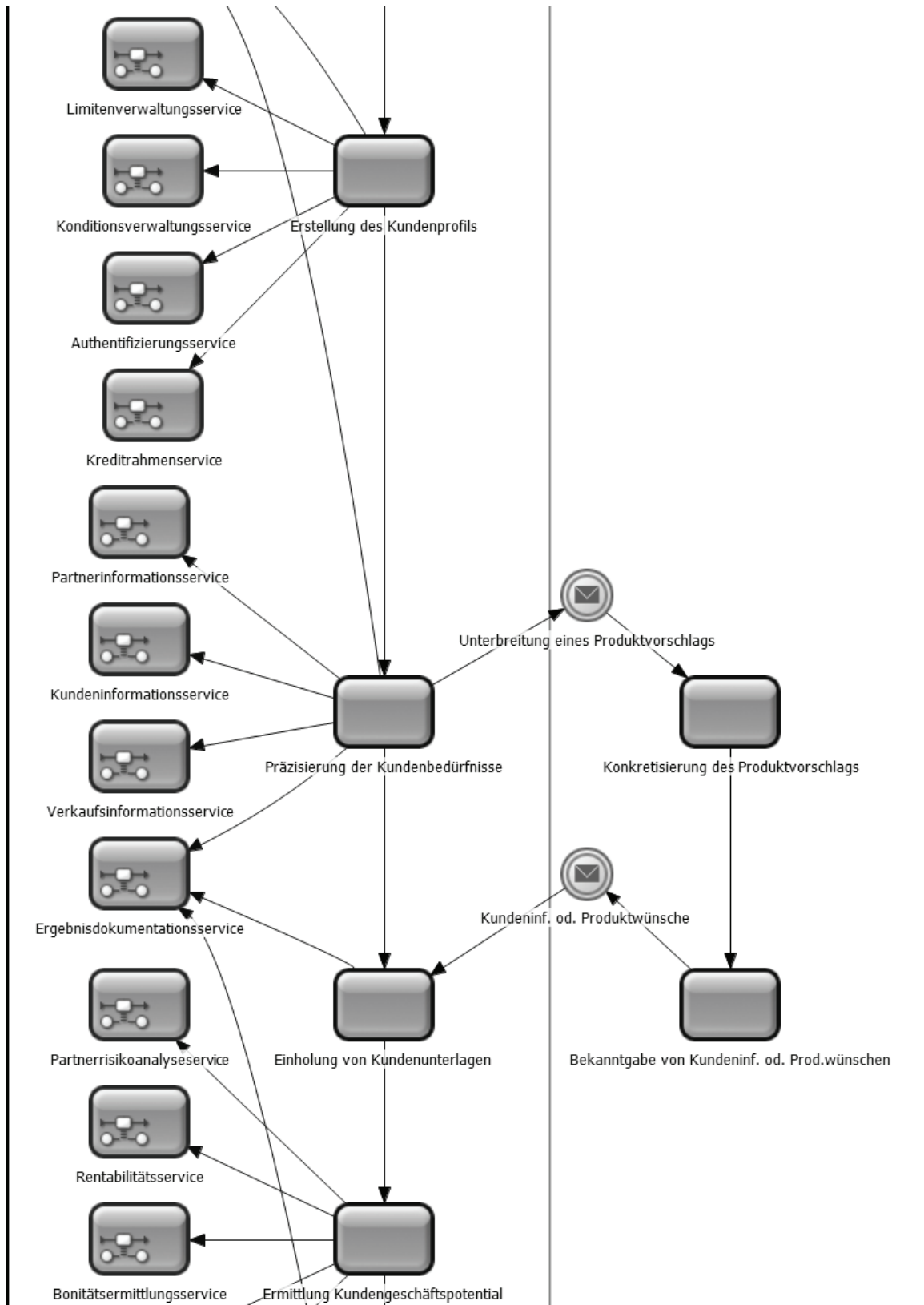
	(Benachrichtigung des Kunden bei Änderung des Leistungsangebots für bereits erworbene Produkte)
Prozesse	Definition und unterstützende PIPs
Ermittlung Zusatzpotential	Interner Prozess, keine PIP-Unterstützung
kontinuierliche Dokumentation	Interner Prozess, keine PIP-Unterstützung
Beendigung Kundenbeziehung	Beendigung der Kundenbeziehung 1A2: Maintain Account (Löschung des Kundenprofils)

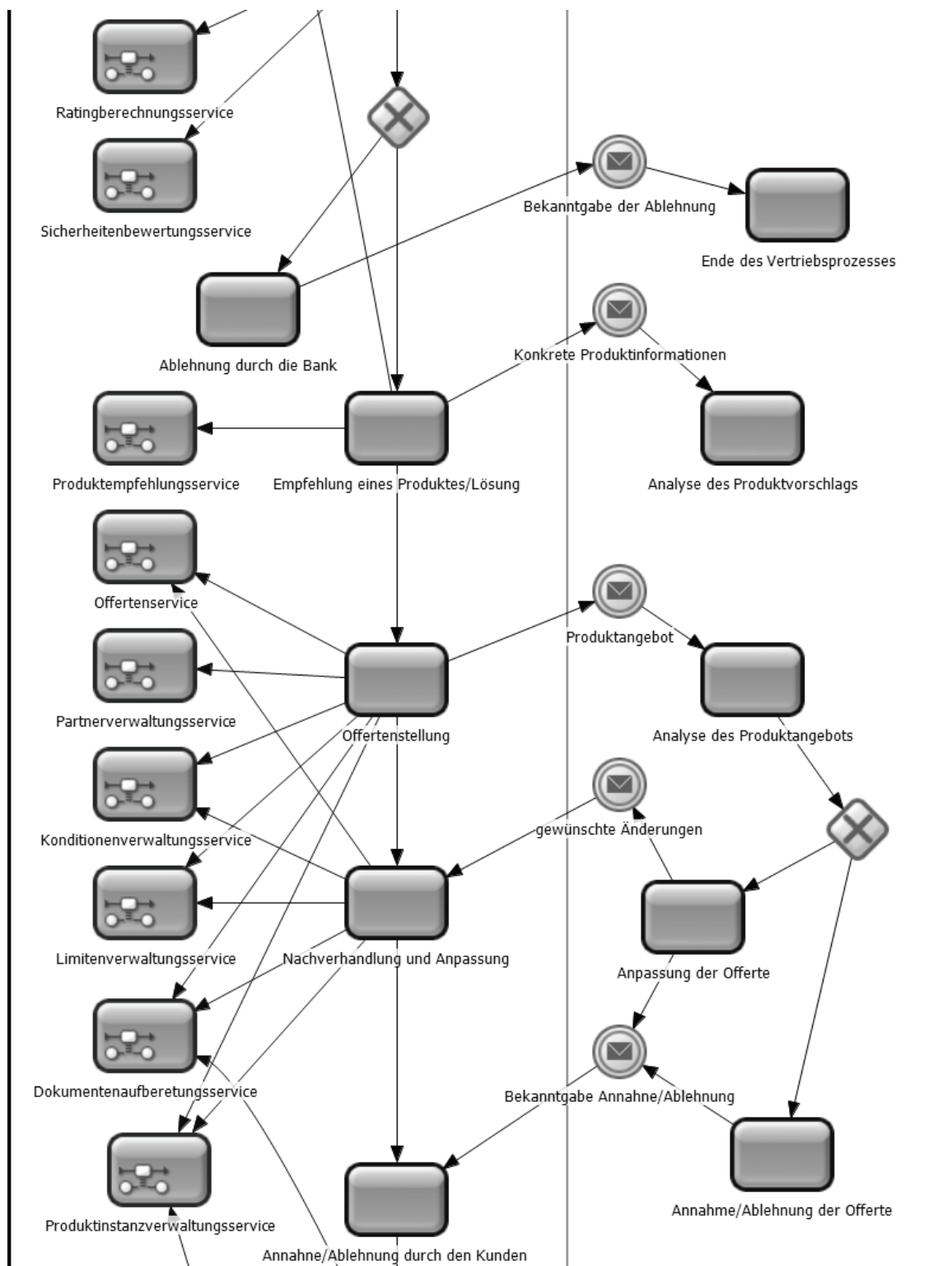
Tabelle 12: Kombination des Referenzprozess Vertrieb mit den PIP-Spezifikationen (ausführlich)

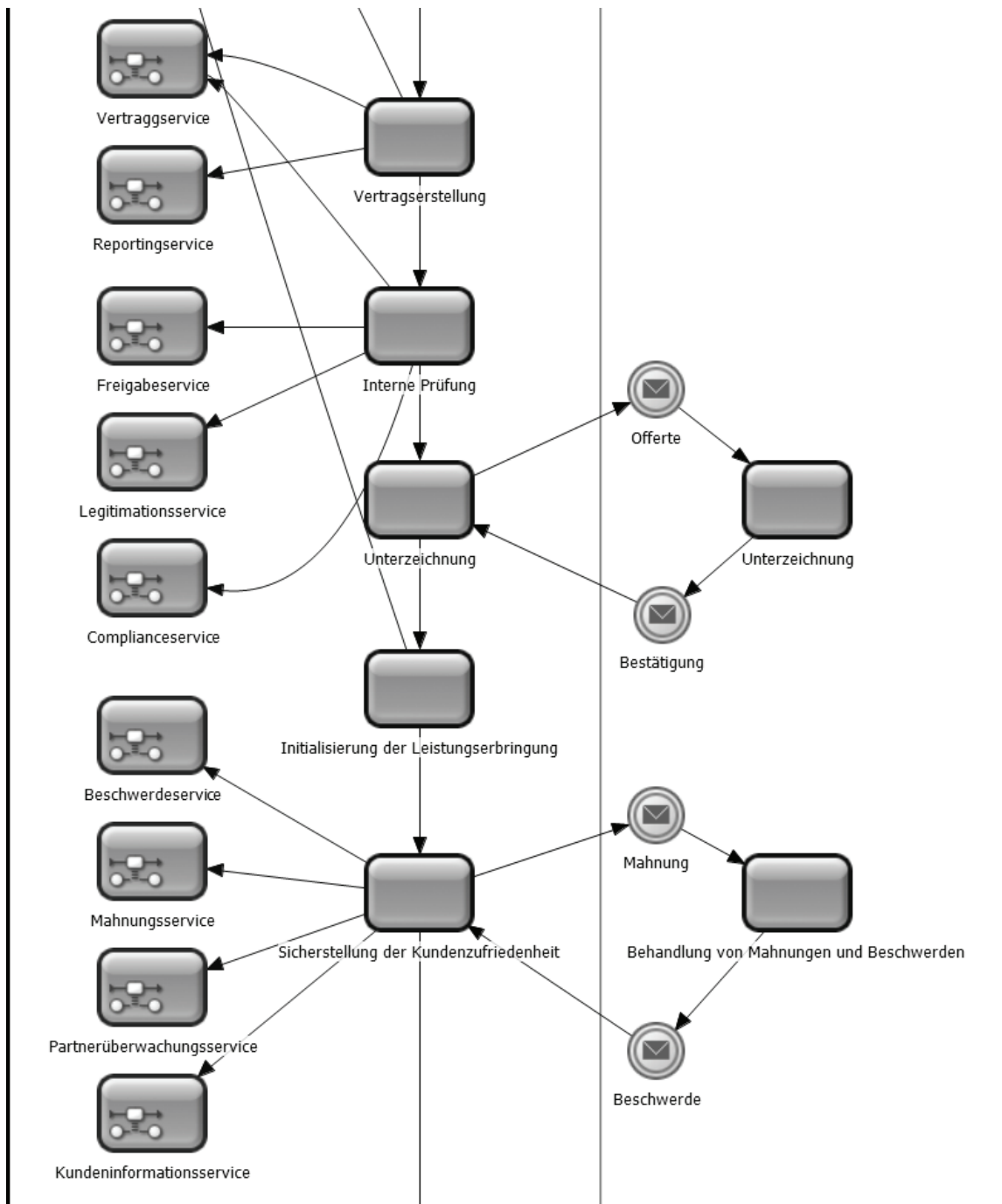
Anhang H: Referenzprozess Vertrieb als Business Process Diagram



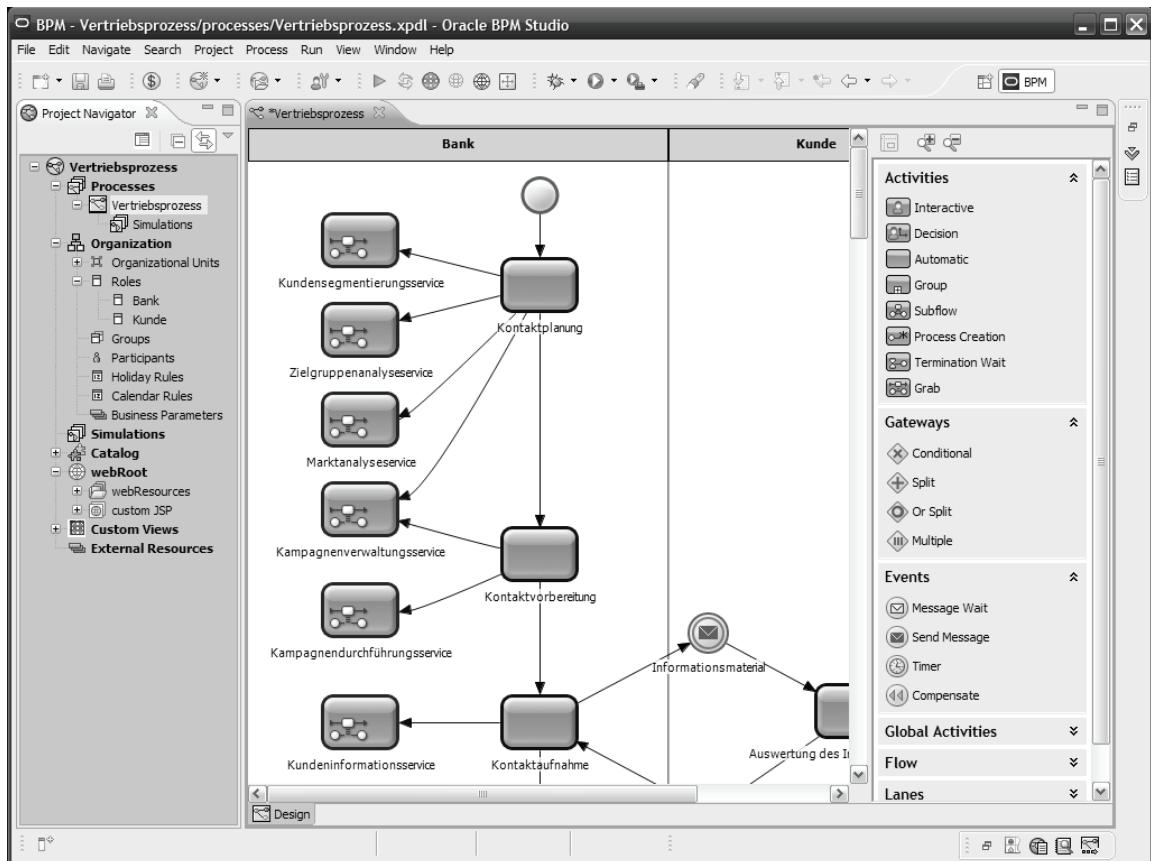
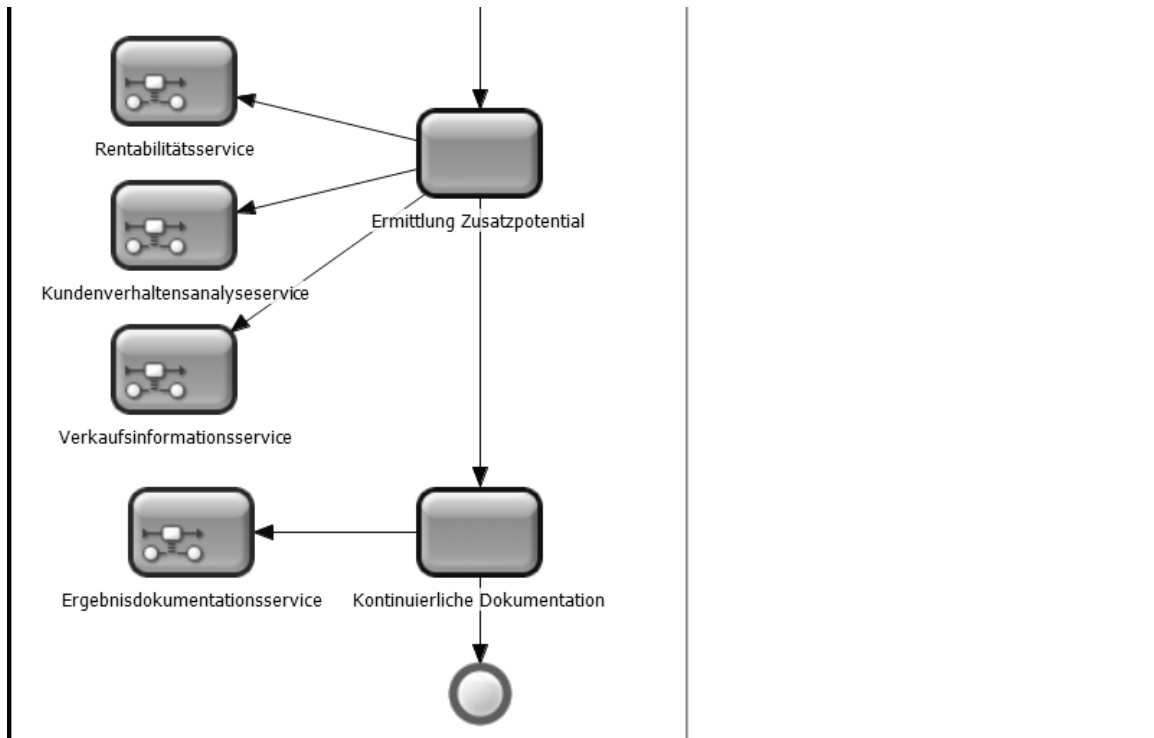








# Anhänge



Anhang I: Bildschirmfotografie der ORACLE WebLogic Integration Console

The screenshot shows the Oracle WebLogic Integration Administration Console interface. The browser address bar indicates the URL: `http://localhost:7001/wlconsole/wlconsole.portal?_nfb=true&ProcMonViewAllInstancePortlet_actionOverride=`. The console title is "ORACLE WebLogic Integration Administration Console".

The main navigation bar includes: "Process Instance Monitoring", "Welcome, weblogic", "Connected to : tptutorial2", "Home", "WLS Console", "LOGOUT", "Help", and "AskOracle".

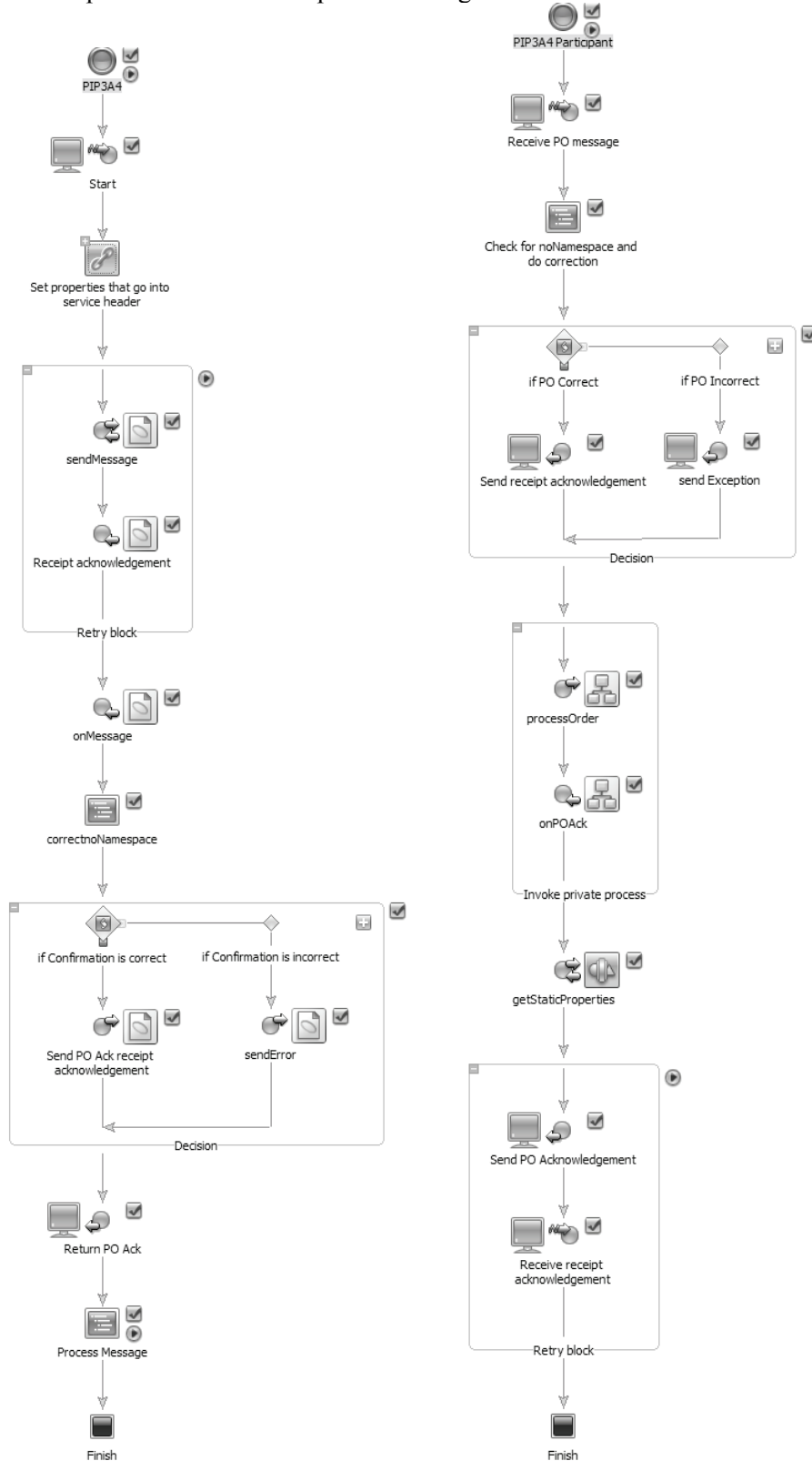
The left sidebar contains the following menu items: "Process Instances" (with sub-items "View Statistics", "View All", "Advanced Search", "System Health"), "System Configuration", "Process Instance Monitoring" (selected), "Process Configuration", "Message Broker", "Event Generators", and "Trading Partner".

The main content area displays "Process Instance Details" for a specific instance. Below this, there is a section for "Child Instances" which contains a table with 2 columns: "ID" and "Display Name". The table shows two completed instances.

ID	Display Name	Status	Start Time	Elapsed Time
192.168.178.32-60b2c40c.120614dcb2c.-7fe8	PIP3A4ParticipantPrivate	Completed	01.04.09 12:55	0.5 secs
192.168.178.32-60b2c40c.120614dcb2c.-7fdb	PrivateAdministratorAlert	Completed	01.04.09 12:55	0.1 secs

Anhänge

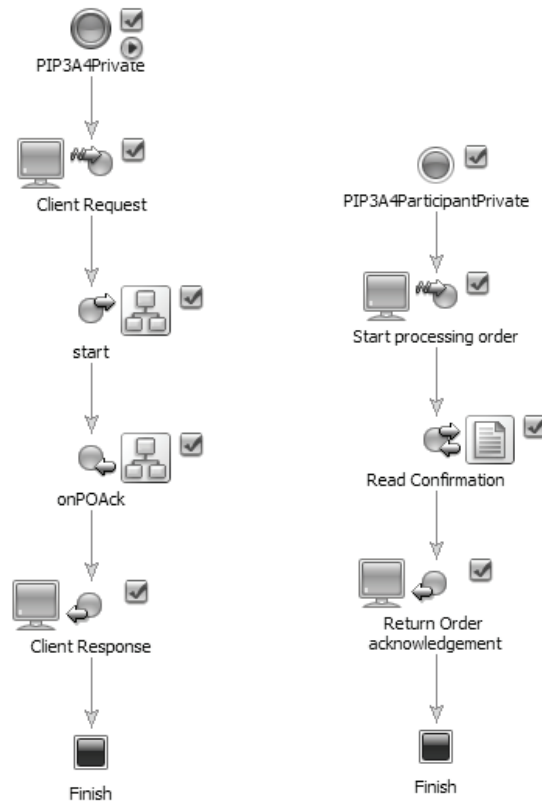
Anhang J: Geschäftsprozessmodell zur Implementierung der PIP 3A4



Öffentlicher Teil des Geschäftsprozesses

PIP 3A4 auf der Seite der Bank  
(PIP3A4.java)

PIP 3A4 auf der Seite des Kunden  
(PIP3A4Participant.java)



Privater Teil des Geschäftsprozesses, Kopplung der PIP 3A4 an den internen Prozess

Seite der Bank

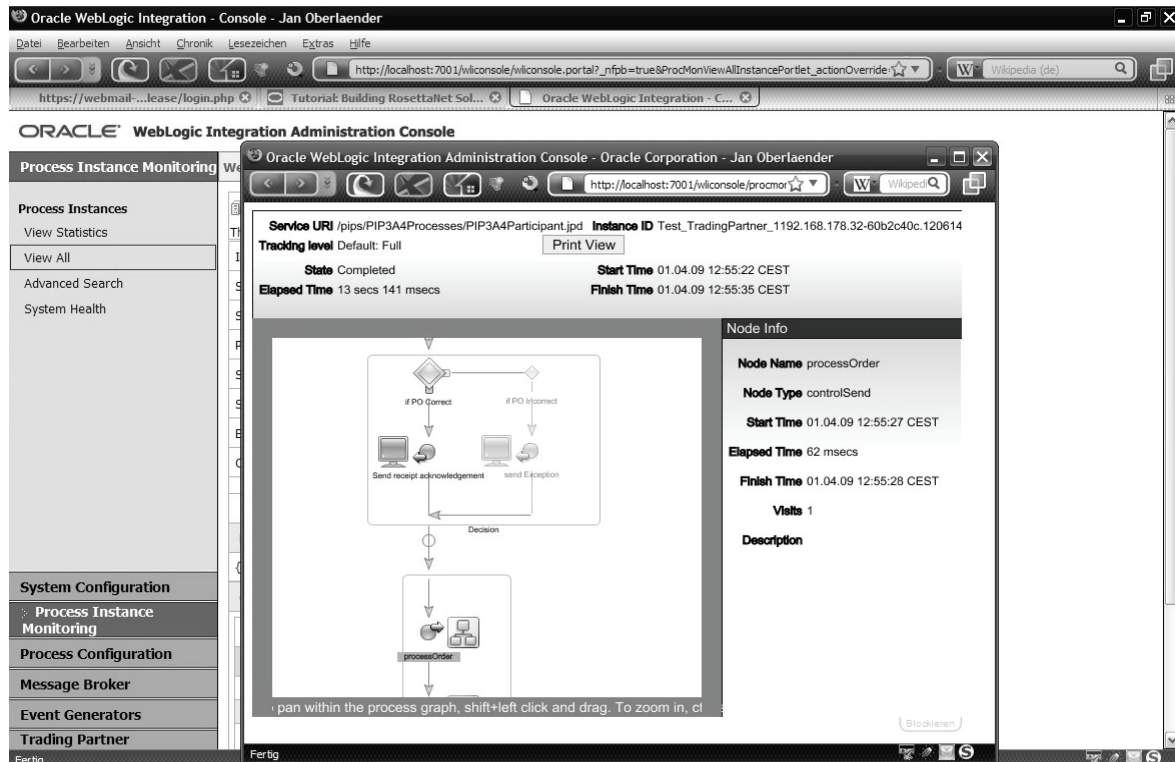
(PIP3A4Private.java)

Seite des Kunden

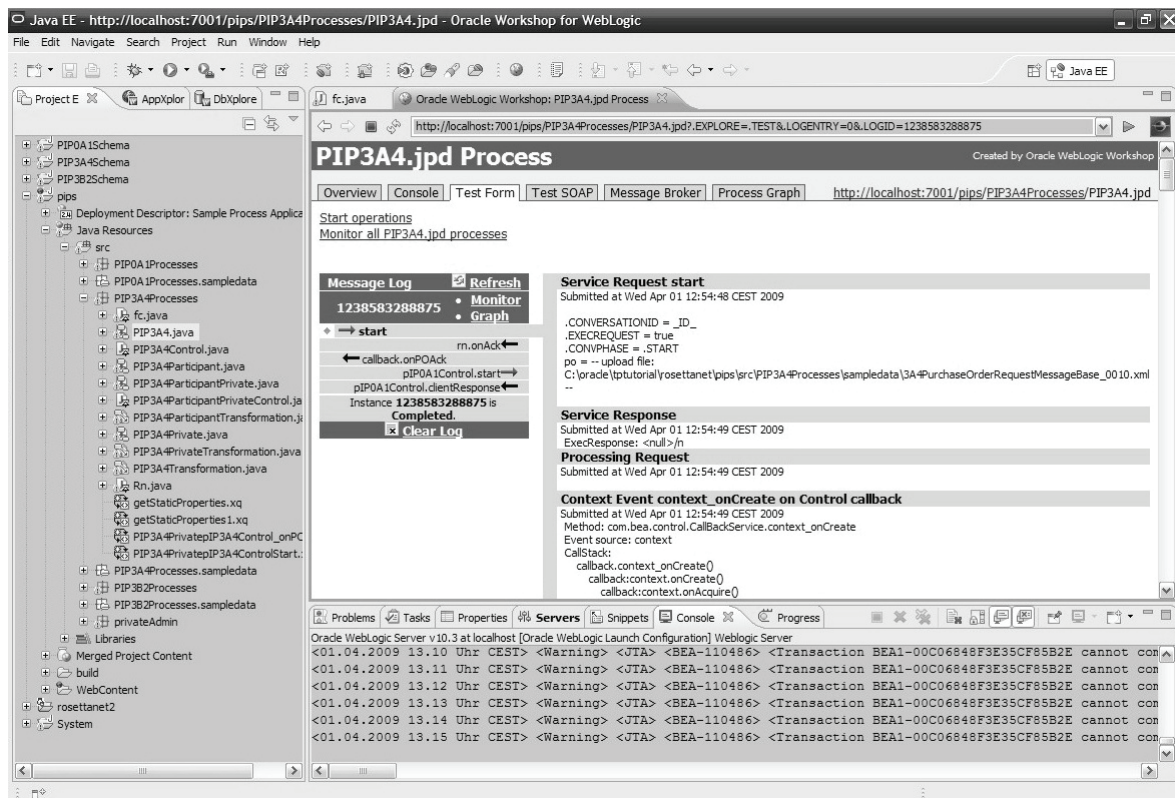
(PIP3A4ParticipantPrivate.java)

# Anhänge

## Anhang K: Bildschirmfotos der Ausführung der RosettaNet-PIP 3A4



Darstellung des Prozessausführungsstatus über den Webbrowser



Darstellung des Prozessausführungsstatus über die Ausführungsumgebung



## 1. Prozess-Start (Kunde übermittelt die Bestellung an den Anbieter)

Message Log	Refresh	Service Request start
1238583288875	<ul style="list-style-type: none"> <li>Monitor</li> <li>Graph</li> </ul>	Submitted at Wed Apr 01 12:54:48 CEST 2009
<ul style="list-style-type: none"> <li>→ start</li> <li>← callback.onPOAck</li> <li>pIPOA1Control.start</li> <li>pIPOA1Control.clientResponse</li> </ul>	<ul style="list-style-type: none"> <li>rn.onAck</li> </ul>	<pre>.CONVERSATIONID = _ID_ .EXECREQUEST = true .CONVPHASE = .START po = -- upload file: C:\prade\tputorial\rosettanet\pips\src\PIP3A4Processes\sampladata\3A4PurchaseOrderRequestMessageBase_0010.xml --</pre>
Instance 1238583288875 is Completed.	Clear Log	<b>Service Response</b> Submitted at Wed Apr 01 12:54:49 CEST 2009 ExecResponse: <null>/n
		<b>Processing Request</b> Submitted at Wed Apr 01 12:54:49 CEST 2009
		<b>Context Event context_onCreate on Control callback</b> Submitted at Wed Apr 01 12:54:49 CEST 2009 Method: com.bea.control.CallBackService.context_onCreate Event source: context

## 2. onAcknowledgement (Anbieter bestätigt den Eingang der Bestellung)

Message Log	Refresh	External Service Callback rn.onAck
1238583288875	<ul style="list-style-type: none"> <li>Monitor</li> <li>Graph</li> </ul>	Submitted at Wed Apr 01 12:55:28 CEST 2009
<ul style="list-style-type: none"> <li>→ start</li> <li>← callback.onPOAck</li> <li>pIPOA1Control.start</li> <li>pIPOA1Control.clientResponse</li> </ul>	<ul style="list-style-type: none"> <li>rn.onAck</li> </ul>	Executable Request:Callback.onAck <b>Returned from rn_onAck</b> Submitted at Wed Apr 01 12:55:28 CEST 2009
Instance 1238583288875 is Completed.	Clear Log	

## 3. PurchaseOrderAcknowledgement (Anbieter bestätigt die Bestellung, in dem er diese fälschlicherweise komplett zurück sendet; im Normalfall würde jetzt eine „vernünftige“ Antwort versendet werden)

Message Log	Refresh	Client Callback
1238583288875	<ul style="list-style-type: none"> <li>Monitor</li> <li>Graph</li> </ul>	Submitted at Wed Apr 01 12:55:34 CEST 2009
<ul style="list-style-type: none"> <li>→ start</li> <li>← callback.onPOAck</li> <li>pIPOA1Control.start</li> <li>pIPOA1Control.clientResponse</li> </ul>	<ul style="list-style-type: none"> <li>rn.onAck</li> </ul>	<pre>onPOAck(&lt;!-- Generated by Base Message Builder --&gt; &lt;DOCTYPE Pip3A4PurchaseOrderConfirmation SYSTEM "3A4_MS_V02_02_PurchaseOrderConfirmation.dtd"&gt; &lt;Pip3A4PurchaseOrderConfirmation&gt; &lt;fromRole&gt; &lt;PartnerRoleDescription&gt; &lt;ContactInformation&gt; &lt;contactName&gt; &lt;FreeFormText&gt;text&lt;/FreeFormText&gt; &lt;/contactName&gt; &lt;EmailAddress&gt;jdoe@doe.com&lt;/EmailAddress&gt; &lt;facsimileNumber&gt; &lt;CommunicationsNumber&gt;000.000.0000&lt;/CommunicationsNumber&gt; &lt;/facsimileNumber&gt;</pre>
Instance 1238583288875 is Completed.	Clear Log	

## 4. PIP 0A1 Start (Der Kunde hat seine eigene Bestellung zurück erhalten und reagiert mit einer Fehlermeldung an den Anbieter auf diesen Fehler)

Message Log	Refresh	External Service Request
1238583288875	<ul style="list-style-type: none"> <li>Monitor</li> <li>Graph</li> </ul>	Submitted at Wed Apr 01 12:55:34 CEST 2009
<ul style="list-style-type: none"> <li>→ start</li> <li>← callback.onPOAck</li> <li>pIPOA1Control.start</li> <li>pIPOA1Control.clientResponse</li> </ul>	<ul style="list-style-type: none"> <li>rn.onAck</li> </ul>	<pre>start(&lt;ros:RosettaNetContext xmlns:ros="http://www.bea.com/wli/control/rosettanetContext"&gt; &lt;ros:from&gt;000000001&lt;/ros:from&gt; &lt;ros:to&gt;000000002&lt;/ros:to&gt; &lt;ros:pip&gt;3A4&lt;/ros:pip&gt; &lt;ros:pip-version&gt;V02.02&lt;/ros:pip-version&gt; &lt;ros:from-role&gt;Buyer&lt;/ros:from-role&gt; &lt;ros:to-role&gt;Seller&lt;/ros:to-role&gt; &lt;ros:failure-report-administrator xsi:nil="true" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"/&gt; &lt;ros:global-usage-code&gt;Test&lt;/ros:global-usage-code&gt; &lt;ros:debug-mode&gt;&gt;false&lt;/ros:debug-mode&gt; &lt;ros:message-tracking-id&gt;000000001 000000002 1238583292437 03317615364570328413&lt;/ros:message-tracking-id&gt; &lt;ros:protocol-name&gt;RosettaNet&lt;/ros:protocol-name&gt; &lt;ros:protocol-version&gt;2.0&lt;/ros:protocol-version&gt; &lt;ros:conversation-id&gt;Test_TradingPartner_1192.168.178.32-60b2c40c.120614dcb2c.-7fed&lt;/ros:conversation-id&gt; &lt;ros:process-instance-id&gt;1238583288875&lt;/ros:process-instance-id&gt; &lt;ros:process-uri&gt;/pips/PIP3A4Processes/PIP3A4.jpd&lt;/ros:process-uri&gt; &lt;ros:business-action&gt;Purchase Order Request Action&lt;/ros:business-action&gt; &lt;ros:document-date-timestamp&gt;20020311T080030.005Z&lt;/ros:document-date-timestamp&gt; &lt;ros:proprietary-identifier&gt;pDocumentID&lt;/ros:proprietary-identifier&gt; &lt;/ros:RosettaNetContext&gt;</pre>
Instance 1238583288875 is Completed.	Clear Log	<b>External Service Response</b> Submitted at Wed Apr 01 12:55:34 CEST 2009 null

## Anhänge

### 5. PIP0A1 Client Response (Der Anbieter bestätigt dem Kunden den Erhalt der Fehlermeldung; die Konversation endet an dieser Stelle)

Message Log	Refresh
1238583288875	• Monitor • Graph
→ start	
← callback.onPOAck	m.onAck←
pIP0A1Control.start→	
◀ pIP0A1Control.clientResponse←	
Instance 1238583288875 is Completed.	
Clear Log	

**External Service Callback pIP0A1Control.clientResponse**  
Submitted at Wed Apr 01 12:55:36 CEST 2009  
Executable Request:Callback.clientResponse

**Service Response**  
Submitted at Wed Apr 01 12:55:36 CEST 2009  
ExecResponse: <null>/n

**Processing Request**  
Submitted at Wed Apr 01 12:55:36 CEST 2009

**Control Event pIP0A1Control\_clientResponse**  
Submitted at Wed Apr 01 12:55:36 CEST 2009  
Method: PIP3A4Processes.PIP3A4.pIP0A1Control\_clientResponse  
Event source: pIP0A1Control  
CallStack:  
pIP0A1Control\_clientResponse()

**Process Completed**  
Submitted at Wed Apr 01 12:55:36 CEST 2009

**Enqueued call to pIP0A1Control\_clientResponse**  
Submitted at Wed Apr 01 12:55:36 CEST 2009

**Request Complete**  
Submitted at Wed Apr 01 12:55:36 CEST 2009



Die Forschungsberichte des Instituts für Wirtschaftsinformatik (IWi) der Universität Leipzig erscheinen in unregelmäßigen Abständen.

Ein Heft kostet 15 Euro, Erscheinungsort ist immer Leipzig.

Bisher in dieser Reihe veröffentlichte Forschungsberichte:

Heft 1: Hrach, C.; Alt, R.: Einsatz von Business Intelligence-Technologien in Call Centern, 2008.

Heft 2: Schmelich, V.; Alt, R.: Functional Analysis of Open Source ERP Systems – An Exploratory Analysis, 2008.

Heft 3: Alt, R.; Eisenecker, U.; Franczyk, B.: 9. Interuniversitäres Doktorandenseminar Wirtschaftsinformatik der Universitäten Halle-Wittenberg, Jena und Leipzig, 2008.

Heft 4: Müller, R.: Konzeption und prototypische Implementierung eines Generators zur Softwarevisualisierung in 3D, 2009.



Impressum:

Prof. Dr. Rainer Alt,  
Prof. Dr. Ulrich Eisenecker,  
Prof. Dr. Bogdan Franczyk

Visualisierung:

Andreas M. Gärtner u. Oliver A. Christ, Hamburg

Grimmaische Straße 12  
D-04109 Leipzig  
[www.iwi.uni-leipzig.de](http://www.iwi.uni-leipzig.de)

**ISSN : 1865-3189**