

**Carmen Hackländer**

**ebXML. Kritische Darstellung eines  
Standards zur Unterstützung von  
e-Business-Prozessen**

**Diplomarbeit**

# BEI GRIN MACHT SICH IHR WISSEN BEZAHLT



- Wir veröffentlichen Ihre Hausarbeit, Bachelor- und Masterarbeit
- Ihr eigenes eBook und Buch - weltweit in allen wichtigen Shops
- Verdienen Sie an jedem Verkauf

Jetzt bei [www.GRIN.com](http://www.GRIN.com) hochladen  
und kostenlos publizieren



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

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

Dieses Werk sowie alle darin enthaltenen einzelnen Beiträge und Abbildungen sind urheberrechtlich geschützt. Jede Verwertung, die nicht ausdrücklich vom Urheberrechtsschutz zugelassen ist, bedarf der vorherigen Zustimmung des Verlanges. Das gilt insbesondere für Vervielfältigungen, Bearbeitungen, Übersetzungen, Mikroverfilmungen, Auswertungen durch Datenbanken und für die Einspeicherung und Verarbeitung in elektronische Systeme. Alle Rechte, auch die des auszugsweisen Nachdrucks, der fotomechanischen Wiedergabe (einschließlich Mikrokopie) sowie der Auswertung durch Datenbanken oder ähnliche Einrichtungen, vorbehalten.

### **Impressum:**

Copyright © 2002 GRIN Verlag  
ISBN: 9783656990703

### **Dieses Buch bei GRIN:**

<https://www.grin.com/document/185815>

**Carmen Hackländer**

**ebXML. Kritische Darstellung eines Standards zur  
Unterstützung von e-Business-Prozessen**

## **GRIN - Your knowledge has value**

Der GRIN Verlag publiziert seit 1998 wissenschaftliche Arbeiten von Studenten, Hochschullehrern und anderen Akademikern als eBook und gedrucktes Buch. Die Verlagswebsite [www.grin.com](http://www.grin.com) ist die ideale Plattform zur Veröffentlichung von Hausarbeiten, Abschlussarbeiten, wissenschaftlichen Aufsätzen, Dissertationen und Fachbüchern.

### **Besuchen Sie uns im Internet:**

<http://www.grin.com/>

<http://www.facebook.com/grincom>

[http://www.twitter.com/grin\\_com](http://www.twitter.com/grin_com)

**ebXML:**  
**Kritische Darstellung eines Standards**  
**zur Unterstützung von e-Business-Prozessen**

**Diplomarbeit**

**Institut für Wirtschaftsinformatik**  
**Lehrstuhl für Entwicklung betrieblicher Informationssysteme**  
**Schwerpunkt Wirtschaftsinformatik und Informationswirtschaft**

Fachbereich Wirtschaftswissenschaften  
Johann Wolfgang Goethe-Universität  
Frankfurt am Main

von

cand. rer. pol. Carmen Hackländer

Studienrichtung Betriebswirtschaftslehre

# Inhaltsverzeichnis

	Seite
<b>Inhaltsverzeichnis.....</b>	<b>I</b>
<b>Abbildungsverzeichnis.....</b>	<b>VII</b>
<b>Tabellenverzeichnis.....</b>	<b>X</b>
<b>Abkürzungsverzeichnis.....</b>	<b>XI</b>
<b>1 Einleitung .....</b>	<b>1</b>
1.1 Die ebXML Initiative.....	1
1.2 Zielsetzung und Aufbau der Arbeit.....	5
1.3 Formale und rechtliche Anmerkungen.....	6
<b>2 EbXML Systemübersicht und Grundlagen .....</b>	<b>8</b>
2.1 EbXML Anwendungsbeispiel.....	8
2.2 EbXML Komponenten.....	10
2.2.1 Business Process Specification Schema (BPSS) .....	10
2.2.2 Registry/Repository .....	11
2.2.3 Collaboration-Protocol Profile (CPP) und Agreement (CPA).....	11
2.2.4 Core Components .....	12
2.2.5 Messaging Services .....	12
2.3 Ansichten der ebXML Architektur .....	12
2.3.1 Business Operational View (BOV) .....	13
2.3.2 Functional Service View (FSV).....	15
2.4 Funktionale Phasen eines ebXML Szenarios.....	16
2.5 Kurze Zusammenfassung und Folgerungen.....	18

2.6	Grundlagen: Die Unified Modelling Methodology .....	19
2.6.1	Klassifizierung der Modellierungsaktivitäten in UMM .....	22
2.6.1.1	UMM Workflows .....	23
2.6.1.2	UMM Phasen .....	24
2.6.1.3	UMM Iterationen .....	25
2.6.2	Workflows und Worksheets in UMM und ebXML.....	25
2.6.2.1	Business Modelling Workflow .....	27
2.6.2.2	Requirements Workflow.....	28
2.6.2.3	Analysis Workflow .....	29
2.6.2.4	Design Workflow.....	30
2.6.3	Das UMM Metamodell.....	30
2.6.4	UMM Patterns .....	33
<b>3</b>	<b>Geschäftsprozess-Modellierung in ebXML .....</b>	<b>35</b>
3.1	Die Grundidee .....	35
3.2	Geschäftsprozesse .....	39
3.3	Das ebXML Business Process Specification Schema .....	41
3.3.1	BPSS Systemüberblick .....	44
3.3.2	Business Transactions.....	49
3.3.2.1	Business Signals .....	53
3.3.2.2	Dokumente einer Business Transaction.....	58
3.3.2.3	Grundlegende Semantik von Business Transactions.....	62
3.3.2.3.1	Erstellung rechtsverbindlicher Verträge .....	63
3.3.2.3.2	Non-Repudiation.....	64
3.3.2.3.3	Authorization Security.....	65
3.3.2.3.4	Document Security .....	65
3.3.2.3.5	Reliability .....	66
3.3.2.4	Semantik von Business Transactions zur Laufzeit .....	68
3.3.2.4.1	Time-outs .....	70
3.3.2.4.2	Exceptions.....	71
3.3.3	Binary Collaborations.....	71
3.3.4	Multiparty Collaborations.....	75
3.3.5	Choreography .....	77
3.3.6	Das Gesamtmodell in UML.....	82

3.3.7	Umsetzung der UML Version in XML .....	84
3.3.7.1	Production Rules.....	84
3.3.7.2	Packages und weitere zusätzliche Elemente der XML Version .....	86
3.3.7.2.1	ProcessSpecification .....	87
3.3.7.2.2	SubstitutionSet.....	88
3.3.7.2.3	Include .....	90
3.3.7.3	Namensvergabe und Referenzierung in ebXML .....	91
3.3.8	XML Version des ebXML BPSS .....	93
3.4	Das Business Service Interface (BSI) .....	94
3.5	Folgerungen des ebXML BPSS .....	97
<b>4</b>	<b>EbXML Registry und Repository.....</b>	<b>99</b>
4.1	Das Registry Information Model (RIM) .....	100
4.1.1	RIM Übersicht: High Level Public View .....	101
4.1.2	RIM Detailansicht.....	103
4.1.2.1	Interfaces RegistryObject und Versionable .....	104
4.1.2.2	Interface RegistryEntry und seine Unterklassen.....	105
4.1.2.3	Interface Association .....	106
4.1.2.4	Klassifizierung eines RegistryEntry .....	107
4.1.2.5	Abbildung von Prüfungsketten .....	109
4.1.2.6	RIM Sicherheitsaspekte .....	110
4.2	EbXML Registry Services .....	111
4.2.1	Systemübersicht des ebXML Registry .....	112
4.2.1.1	Architektur des Registry .....	113
4.2.1.2	Profile und Verträge im ebXML Registry .....	115
4.2.1.3	Interface Services.....	116
4.2.2	Der ObjectManagement Service.....	118
4.2.3	Der ObjectQueryManagement Service.....	120
4.2.3.1	Browse and Drill Down Query .....	121
4.2.3.2	Filtered Query .....	122
4.2.3.3	SQL Query.....	124
4.2.3.4	Content Retrieval .....	125
4.2.4	Sicherheit des Registry .....	126

4.3	Folgerungen des ebXML Registry/Repository .....	129
<b>5</b>	<b>Weitere Inhalte des ebXML Registry .....</b>	<b>131</b>
5.1	Collaboration-Protocol Profile and Agreement .....	131
5.1.1	Überblick über CPP und CPA .....	133
5.1.2	Struktur eines CPP .....	137
5.1.2.1	Element CollaborationProtocolProfile.....	139
5.1.2.2	Element PartyInfo .....	140
5.1.2.3	Element CollaborationRole .....	141
5.1.2.4	Element DeliveryChannel.....	143
5.1.3	Struktur eines CPA .....	144
5.2	Core Components.....	147
5.2.1	Konzept der Core Components.....	149
5.2.2	Gesamtzusammenhang der Core Component Dokumentationen .....	151
5.2.2.1	Context and Re-Usability of Core Components .....	153
5.2.2.2	Catalogue of Context Drivers .....	155
5.2.2.3	Document Assembly and Context Rules .....	155
5.2.2.4	Naming Convention for Core Components .....	156
5.2.2.5	Core Component Discovery and Analysis.....	157
5.2.2.6	Guide to the Core Component Dictionary .....	159
5.2.3	Gesamtzusammenhang: Business Process, CPP/CPA und Core Components .....	161
<b>6</b>	<b>EbXML Message Service .....</b>	<b>166</b>
6.1	Simple Object Access Protocol (SOAP).....	167
6.1.1	Struktur einer SOAP Nachricht .....	168
6.1.2	SOAP Messages with Attachments (SWA).....	171
6.1.3	Sicherheit und Zuverlässigkeit von SOAP .....	172
6.2	Überblick über den ebXML Message Service .....	173
6.3	Packaging.....	176
6.4	EbXML SOAP Extensions .....	179
6.4.1	Header Extensions .....	181
6.4.1.1	<MessageHeader> Element.....	181

6.4.1.2	<TraceHeaderList> Element.....	183
6.4.1.3	<Errorlist> Element .....	184
6.4.1.4	<Acknowledgement> Element .....	185
6.4.1.5	<Via> Element.....	185
6.4.2	SOAP Body Extensions.....	186
6.5	Weitere Dienste des Message Service Handler (MSH) .....	188
6.6	Zuverlässigkeit der Nachrichtenübertragung .....	189
6.6.1	EbXML Reliable Messaging Protocol.....	191
6.6.2	Fehlerbehandlung .....	195
6.7	Sicherheit der ebXML Nachrichtenübertragung.....	195
<b>7</b>	<b>Kritische Würdigung von ebXML .....</b>	<b>198</b>
7.1	Vorhandene Kritiken über ebXML.....	198
7.1.1	"Overview of the ebXML Architectures" .....	199
7.1.2	"ebXML and Interoperability" .....	201
7.1.3	"ebXML and SMEs".....	202
7.1.4	"Market Impact of the ebXML Infrastructure Specifications" .....	204
7.1.5	"Process Modelling for e-Business" .....	205
7.2	Reaktionen auf die Artikel .....	206
7.3	Praktische Schwierigkeiten und Mängel der ebXML Spezifikationen... 210	
7.3.1	EbXML BPSS.....	212
7.3.2	EbXML Registry/Repository.....	214
7.3.3	EbXML CPP/CPA.....	214
7.3.4	EbXML Core Components.....	215
7.3.5	EbXML Message Service .....	216
<b>8</b>	<b>Zusammenfassung und Ausblick .....</b>	<b>217</b>
	<b>Anhang.....</b>	<b>XV</b>
	Anhang A: Beispiel einer XML Business Process Specification .....	XV
	Anhang B: BPSS Document Type Definition (DTD) .....	XXI
	Anhang C: BPSS XML Schema .....	XXVI

Anhang D: EbXML Links ..... XXXVIII

Anhang E: EbXML Glossar .....XL

**Literaturverzeichnis..... XLVII**

## Abbildungsverzeichnis

	Seite
Abbildung 2-1: Überblick über den Dialog zweier Unternehmen zur Durchführung von e-Business mit ebXML .....	9
Abbildung 2-2: Zusammenhang zwischen BOV und FSV.....	13
Abbildung 2-3: ebXML Business Operational View .....	14
Abbildung 2-4: ebXML Functional Service View .....	16
Abbildung 2-5: Elemente und Vorgänge der funktionalen ebXML Phasen Implementierung, Discovery und Retrieval.....	18
Abbildung 2-6: UMM Phasen und Workflows .....	23
Abbildung 2-7: Übersicht der UMM Workflows und Artifacts .....	26
Abbildung 2-8: UMM Metamodell .....	32
Abbildung 2-9: UMM Metamodell und ebXML BPSS .....	33
Abbildung 3-1: Übergangsgraph eines Automaten .....	36
Abbildung 3-2: Synchronisation zweier Automaten .....	38
Abbildung 3-3: ebXML Business Process und Anwendungsfallbeispiel.....	40
Abbildung 3-4: Zusammenhang zwischen UMM und ebXML.....	42
Abbildung 3-5: Komponenten des ebXML BPSS und ihre Beziehung zu UMM, CPP/CPA und Core Components .....	45
Abbildung 3-6: Zusammenhang des BPSS mit anderen ebXML Komponenten ..	46
Abbildung 3-7: UML Klassendiagramm einer Business Transaction .....	50
Abbildung 3-8: UMM Business Transaction Pattern - Request/Response.....	53
Abbildung 3-9: Document Flows und Business Signals einer Transaktion und ihre Reihenfolge.....	53
Abbildung 3-10: UML Sequenzdiagramm eines Nachrichtentransfers.....	55
Abbildung 3-11: UML Klassendiagramm der Elemente einer Business Transaction-Nachricht .....	59
Abbildung 3-12: UML Klassendiagramm der Elemente einer Binary Collaboration .....	72
Abbildung 3-13: UML Klassendiagramm einer Multiparty Collaboration .....	76

## VIII

Abbildung 3-14: UML Klassendiagramm einer Choreographie .....	78
Abbildung 3-15: UML Klassendiagramm des ebXML BPSS Gesamtmodells.....	83
Abbildung 3-16: Hierarchisches Modell des Elements Package.....	86
Abbildung 3-17: Hierarchisches Modell der DTD .....	88
Abbildung 3-18: Das Business Service Interface .....	94
Abbildung 4-1: UML Klassendiagramm der Objekte des RIM .....	102
Abbildung 4-2: UML Klassendiagramm: Detailansicht des RIM.....	104
Abbildung 4-3: Beispiel einer Assoziation eines RegistryEntry .....	106
Abbildung 4-4: Beispiel eines Klassifikationsbaums .....	108
Abbildung 4-5: UML Klassendiagramm der Klassifikation eines RegistryEntry im RIM .....	108
Abbildung 4-6: UML Klassendiagramm des Sicherheitsmodells im RIM .....	110
Abbildung 4-7: Mögliche Topologien der Architektur eines ebXML Registry ..	114
Abbildung 4-8: UML Klassendiagramm der Interfaces des ebXML Registry....	116
Abbildung 4-9: UML Klassendiagramm der Hierarchie der Klasse RegistryResponse .....	118
Abbildung 4-10: UML Zustandsdiagramm des Lebenszyklus eines Repository Items .....	119
Abbildung 4-11: UML Sequenzdiagramm des Protokolls SubmitObject .....	120
Abbildung 4-12: UML Sequenzdiagramm des Protokolls GetRootClassificationNodes.....	122
Abbildung 4-13: UML Klassendiagramm der Basisstruktur eines Clause.....	123
Abbildung 4-14: UML Sequenzdiagramm einer typischen Sequenz von Queries und Retrieval .....	126
Abbildung 5-1: Übersicht der Definition eines CPP .....	133
Abbildung 5-2: Übersicht der Definition eines CPA.....	134
Abbildung 5-3: Übersicht des Zusammenspiels von CPP/CPA und ebXML Registry.....	135
Abbildung 5-4: Beispiele für DeliveryChannels .....	143
Abbildung 5-5: Core Components in einem Business Document .....	150
Abbildung 5-6: Zusammenhang der Core Component Technical Reports und Catalogues.....	152

Abbildung 5-7: Discovery und Analyse .....	158
Abbildung 5-8: Übersicht der Typen von Core Component-Einträgen im Katalog .....	159
Abbildung 5-9: Zusammenhang zwischen Business Process und Core Components .....	161
Abbildung 5-10: Bestandteile eines Business Document .....	162
Abbildung 5-11: Gesamtzusammenhang zwischen Core Components, CPP, CPA und Business Process .....	164
Abbildung 6-1: Struktur einer SOAP Nachricht .....	169
Abbildung 6-2: Struktur einer SWA Nachricht .....	171
Abbildung 6-3: Funktionale Module des ebXML Message Service .....	175
Abbildung 6-4: Struktur eines ebXML Message Package .....	177
Abbildung 6-5: Auszug aus einem XML Message Package .....	178
Abbildung 6-6: Erfolgreiche Nachrichtenübertragung mit ebXML Reliable Messaging Protocol .....	192
Abbildung 6-7: Verlust einer Nachricht während der Übertragung .....	193
Abbildung 6-8: Verlust einer Bestätigung während der Übertragung .....	193
Abbildung 6-9: Wiederholtes Senden nicht bestätigter Nachrichten .....	194

## Tabellenverzeichnis

	Seite
Tabelle 3-1: UMM Vorgabewerte der Requesting Business Activities .....	55
Tabelle 3-2: UMM Vorgabewerte der Responding Business Activities .....	55
Tabelle 3-3: Umsetzung der UML Klassen aus BPSS in XML Elemente .....	82
Tabelle 3-4: Reihenfolge der Elemente in BPSS DTD und Schema.....	92
Tabelle 6-1: Varianten zuverlässiger Nachrichtenübertragung in ebMS Version 2.0.....	 192

## Abkürzungsverzeichnis

ANSI	American National Standards Institute
API	Application Program Interface
ASC X12	Accredited Standards Committee X12
ASP	Application Service Provider
B2B	Business-to-Business
BIO	Business Information Object
BNF	Backus-Naur Form
BOD	Business Object Document
BOM	Business Operational Map
BOV	Business Operational View
BPA	Business Process Activity
BPMI	Business Process Management Initiative
BPML	Business Process Markup Language
BPMS	Business Process Management System
BPSS	Business Process Specification Schema
BRV	Business Requirements View
BSI	Business Service Interface
BSV	Business Service View
BTV	Business Transaction View
CCT	Core Component Type
CII	Center for Information of Industry
CORBA	Common Object Request Broker Architecture
CPA	Collaboration-Protocol Agreement
CPP	Collaboration-Protocol Profile
CRM	Customer Relationship Management
DCOM	Distributed Component Object Model
DN	Distinguished Name
DTD	Document Type Definition
e-Business	electronic Business
EAI	Enterprise Application Integration

ebMS	ebXML Message Service
eBP	executable Business Process
ebXML	electronic business XML
EDI	Electronic Data Interchange
EDIFACT	Electronic Data Interchange For Administration, Commerce and Transport
ERP	Enterprise Resource Planning
FSV	Functional Service View
FTP	File Transfer Protocol
GIF	Graphics Interchange Format
GUI	Graphical User Interface
GUID	Globally Unique Identifier
HTTP	HyperText Transfer Protocol
ID	Identification
ID/IDREF	Identification/Identification Reference
IEC	International Electrotechnical Commission
IEEE	Institute of Electrical and Electronics Engineers
IETF	Internet Engineering Task Force
ISO	International Organization for Standardization
ISV	Independent Software Vendors
IT	Informationstechnologie
JAXM	Java API for XML Messaging
JMS	Java Message Service
MIME	Multipurpose Internet Mail Extension
MOM	Message Oriented Middleware
MSH	Message Header Service
OAG	Open Applications Group
OASIS	Organization for the Advancement of Structured Information Standards
OCL	Object Constraint Language
OLAP	OnLine Analytical Processing
OLTP	OnLine Transaction Processing
OMG	Object Management Group
PIP	Partner Interface Process

ROI	Return on Investment
RPC	Remote Procedure Call
RUP	Rational Unified Process
S/MIME	Secure MIME
SAML	Security Assertion Markup Language
SCM	Supply Chain Management
SGML	Standard Generalized Markup Language
SME	Small and Middle sized Enterprises
SMTP	Simple Mail Transfer Protocol
SOAP	Simple Object Access Protocol
SQL	Structured Query Language
SSL	Secure Socket Layer
SWA	SOAP with Attachments
TCP/IP	Transmission Control Protocol/Internet Protocol
TMWG	Techniques and Methodology Working Group
UDDI	Universal Description, Discovery and Integration
UID	Unique Identifier
UBL	Universal Business Language
UML	Unified Modelling Language
UMM	UN/CEFACT Modelling Methodology
UN/CEFACT	United Nations/Center für Trade Facilitation and Electronic Business
UN/EDIFACT	United Nations/Electronic Data Interchange For Administration, Commerce and Transport
URI	Uniform Resource Indicator
URL	Uniform Resource Locator
UTC	Coordinated Universal Time
UUID	Universally Unique Identifier
VAN	Value Added Network
VPN	Virtual Private Network
W3C	World Wide Web Consortium
WSFL	Web Services Flow Language
xBP	executable Business Process
xCBL	XML Common Business Library

XMI	XML Metadata Interchange
XML	eXtensible Markup Language

# 1 Einleitung

## 1.1 Die ebXML Initiative

Electronic Business XML (ebXML) wurde als Projekt zur Entwicklung von Spezifikationen für die direkte Kommunikation und Zusammenarbeit zwischen Partnern im e-Business im Mai 2001 nach 18 Monaten abgeschlossen. Es gilt als eine der ambitioniertesten und wichtigsten Anstrengungen in diesem Bereich, nicht nur für die beiden Initiatoren UN/CEFACT und OASIS.

UN/CEFACT, das "United Nations Center for Trade Facilitation and Electronic Business" [<http://www.unece.org/cefact>], ist als globale Organisation verantwortlich für Strategien und die technische Entwicklung von Handelserleichterungen, auch im e-Business. Insbesondere entwickelte UN/CEFACT internationale Standards wie UN/EDIFACT für Electronic Data Interchange (EDI).

OASIS, die "Organization for the Advancement of Structured Information Standards" [<http://www.oasis-open.org>], ist ein gemeinnütziges Konsortium, das Spezifikationen im Bereich der strukturierten Informationsverarbeitung und zur direkten Kommunikation zwischen Unternehmen erstellt. Letztere basieren auf öffentlichen Standards wie XML und SGML.

Seit mehr als 25 Jahren stellt EDI Unternehmen eine papierlose Geschäftsabwicklung, Kostenreduktion und verbesserte, effiziente Geschäftsprozesse in Aussicht, die auf dem elektronischem Austausch der für die Prozesse nötigen Informationen basieren. Idealerweise sollen Unternehmen jeglicher Größe diese elektronische Geschäftsabwicklung, das sogenannte e-Business, vollkommen spontan und ohne jegliche vorausgehenden Abstimmungen betreiben können. Jedoch erfüllte EDI diese Vision nicht: Nur große Firmen konnten sich eine Implementierung leisten. EDI-gestütztes e-Business spielte sich daher vor allem im Umfeld großer, dominanter Firmen ab, die den Handelspartnern ihre eigenen Standards aufzwingen.

In den vergangenen Jahren wurde die Extensible Markup Language (XML) schnell zum bevorzugten Datenformat für den Austausch strukturierter Informationen und ihrer Metadaten von e-Business-Anwendungen im Internet. Obwohl XML offenere, flexiblere Geschäftsmodelle als EDI ermöglicht, löste es

EDI entgegen vieler Erwartungen nicht ab. Zum einen waren mit einer erfolgreichen Implementierung schon zu viele Erfahrungen in der Abwicklung von Geschäftsprozessen mit EDI verbunden und große Summen investiert, zum anderen sorgte die rapide Zunahme an unterschiedlichen XML Spezifikationen für große Verwirrung und unnötigen Mehrfachaufwand bei den Anwendern. Trotz oder gerade wegen der Flexibilität von XML und seinem unbestrittenen Potential, webbasierte Transaktionen zu vereinfachen, waren nach wie vor für eine problemlose Kommunikation unterschiedlicher Systeme aufwendige Abstimmungen nötig.

Die heute in nahezu allen Unternehmen eingesetzten Enterprise Resource Planning (ERP)-Systeme bieten vielfältige Möglichkeiten für die automatisierte Kommunikation über Unternehmensgrenzen hinweg und unterstützen e-Commerce, Customer Relationship Management (CRM) und Supply Chain Management (SCM). Bisherige Middleware-Ansätze wie CORBA [<http://www.corba.org>] oder DCOM [<http://www.microsoft.com/com/tech/DCOM.asp>] erleichtern zwar die Integration zweier Systeme, behandeln aber nicht das Problem, Kommunikationssysteme zu errichten, die hunderte und tausende von Systemen miteinander verbinden. Denn eine besondere Herausforderung liegt darin, diese Systeme auch bei kleinen und mittelgroßen Unternehmen ohne großen Kostenaufwand in ein Netzwerk vieler, miteinander flexibel verbundener und zusammenarbeitender Unternehmen zu integrieren, das auf Interaktionen zwischen den Teilnehmern aufgebaut ist. [ChCh01 S.11, 207]

Deshalb sollte ein Standard für ein flexibles System im Internet geschaffen werden, in dem alle Unternehmen automatisch mit allen gewünschten Geschäftspartnern auch kurzfristige Geschäftsbeziehungen aufnehmen und direkt kommunizieren können, und zwar ohne aufwendige Abstimmungen und ohne große technologische Hindernisse. Der Standard, der die Infrastruktur für eine solche Interoperabilität errichten soll, ist ebXML.

### Die Vision von ebXML ist

"... to deliver a single set of internationally agreed-upon technical specifications that consist of common XML semantics and related document structures to facilitate global trade." [ebREQ S.6]

"... to create a single global electronic marketplace where enterprises of any size and in any geographical location can meet and conduct business with each other through the exchange of XML based messages. ebXML enables anyone, anywhere, to do business with anyone else over the internet." [ebWhite S.1]

### Die ebXML Spezifikationen sollen

"... together enable a modular, yet complete electronic business framework. If the Internet is the information highway for electronic business, then ebXML can be thought of as providing the on ramps, off ramps, and the rules of the road."  
[ebWhite S.1]

Die meisten bisher eingesetzten e-Business Infrastrukturen automatisierten im Wesentlichen statische, das heißt etablierte, langfristige Geschäftsbeziehungen zwischen bereits bekannten Partnern. Im Gegensatz hierzu soll mit ebXML eine Basis für dynamische Geschäftsbeziehungen geschaffen werden. Dabei bedeutet dynamisch nicht nur die Unterstützung der Suche nach neuen Geschäftspartnern, sondern auch Automatisierung der für eine direkte Zusammenarbeit nötigen Abstimmungen zwischen den Partnern und eine automatische Konfiguration der Kommunikationssysteme. Der Return on Investment (ROI) ergibt sich hier insbesondere aus der potentiellen Ausweitung der Umsätze durch zahlreiche zusätzliche Geschäftsabschlüsse und nicht nur aus Kostenreduzierungen, Zeiteinsparungen oder Vorteilen eines verbesserten Informationsaustauschs. [ChCh01 S.281]

Mit seinem standardisierten Nachrichtensystem begegnet ebXML den Problemen, denen ein Unternehmen gegenübersteht, wenn es mit hunderten von Geschäftspartnern integriert werden und mit ihnen zahlreiche unterschiedliche Geschäftsaktivitäten durchführen muss. Eine Firma, die beispielsweise fünf Kernprozesse automatisiert, wobei jeder dieser Prozesse im Durchschnitt fünf Geschäftsdokumente erfordert und diese Prozesse mit 100 ihrer Geschäftspartner durchführt, muss 2.500 unterschiedliche Netz-Endknoten erstellen und verwalten. Es ist möglich, dass diese Endknoten jederzeit geändert werden müssen, z.B. aufgrund eines neuen Dokumentenformats, geänderter Sicherheitsanforderungen,

URLs etc. Betrachtet man große Unternehmen beispielsweise aus der Luft- und Raumfahrtindustrie mit bis zu 30.000 Lieferanten oder große Automobilhersteller mit ca. 2.500 Informationssystemen, wird sehr leicht deutlich, welcher Aufwand nötig wäre, um an solchen "highly connected", eng verbundenen Systemen teilnehmen zu können. Insbesondere auch kleine Lieferanten müssten einen hohen Aufwand betreiben, um für jeden ihrer wichtigen Kunden spezifische Implementierungen vorzunehmen. [ChCh01 S.197]

EbXML ist ein offener technischer Rahmen, der XML einsetzt, auf widerspruchsfreie und einheitliche Art und Weise standardisiert und erweitert, um eine globale, systemunabhängige, interoperable Infrastruktur für Business-to-Business (B2B)-Transaktionen im Internet aufzubauen. Jedes Unternehmen soll durch die Modellierung seiner Geschäftspraktiken und Informationen in standardisierter Form als XML Dokument an ebXML teilnehmen können.

EbXML basiert auf internationalen Standards wie den XML Empfehlungen des World Wide Web Consortiums (W3C) [<http://www.w3.org>], anderen technischen Spezifikationen der Internet Engineering Task Force (IETF) [<http://www.ietf.org>], International Organization for Standardization (ISO) [<http://www.iso.ch>], Institute of Electrical and Electronics Engineers (IEEE) [<http://www.ieee.org>], International Electrotechnical Commission (IEC) [<http://www.iec.ch>], UN/CEFACT, OASIS und der Object Management Group (OMG) [<http://www.omg.org>] - und soll selbst ein internationaler Standard werden. [ebREQ S.6]

Das Gesamtprojekt war in mehrere Teams aufgeteilt, die verschiedene Aspekte bearbeiteten, z.B. Geschäftsprozess-Modellierung, Verwaltung und Speicherung, Kernkomponenten, Nachrichten-Service, Technischer Aufbau und Sicherheit. Es wurde von mehr als 75 Industrieunternehmen und mehr als 2.000 Partnergesellschaften aus mehr als 30 Ländern unterstützt. [ebWhite S.1]

Auch wenn bei der Abschlusskonferenz vom 7.-11. Mai 2001 in Wien alle ebXML Spezifikationen vorlagen, war klar, dass trotz dieser immensen Leistung die Entwicklung von ebXML in allen Bereichen weitere Arbeiten verlangen würde. Dieser noch andauernde Prozess erfolgte unter der Führung der beiden Initiatoren UN/CEFACT und OASIS sowie mit Unterstützung zahlreicher großer e-Business Organisationen wie z.B. RosettaNet [<http://www.rosettanet.org>], Open

Applications Group (OAG) [<http://www.openapplications.org>] und Covisint [<http://www.covisint.com>].

## 1.2 Zielsetzung und Aufbau der Arbeit

In dieser Arbeit wird versucht, die sehr komplexen ebXML Spezifikationen und Zusammenhänge verständlich zusammenzufassen und zu erläutern. Sie stützt sich vor allem auf die Unterlagen, die im Mai 2001 zur ebXML Abschlusskonferenz veröffentlicht wurden.

Anhand eines allgemeinen Einführungsbeispiels und einer Übersicht über ebXML werden in Kapitel zwei die ebXML Komponenten zunächst allgemein vorgestellt und in den darauffolgenden Kapiteln detailliert erläutert. Dazu wird im Vorfeld die grundlegende Modellierungsmethodik dargestellt, auf die ebXML als Rahmen für eine flexible, direkte Kommunikation und Zusammenarbeit im Internet aufsetzt.

Den Schwerpunkt der Arbeit bildet die Geschäftsprozess-Modellierung in ebXML, die ausführlich in Kapitel drei beschrieben wird. Dieser Teil wird deshalb sehr detailliert behandelt, da die Erkenntnisse der vorliegenden Arbeit als Grundlage für eine spätere weitere Untersuchung dienen sollen, ob Prozesse in ebXML mit Petri-Netzen abgebildet und somit ihre Zusammenhänge simuliert werden können.

Nach der Beschreibung der weiteren ebXML Komponenten, nämlich ebXML Registry und Repository in Kapitel vier, Collaboration-Protocol Profiles/Agreement und Core Components in Kapitel fünf und dem ebXML Message Service in Kapitel sechs, wird in Kapitel sieben ebXML als Gesamtheit einer ausführlichen kritischen Würdigung unterzogen.

Kapitel acht rundet die gesamte Arbeit mit einer Zusammenfassung aller erarbeiteten Ergebnisse ab und gibt einen Ausblick auf weitere Entwicklungen in diesem Bereich des e-Business.

In Anhang A findet sich ein Beispiel für eine ebXML Geschäftsprozess-Spezifikation als XML Dokument, in Anhang B die zugehörige XML DTD und in Anhang C das XML Schema. Anhang D enthält einige interessante ebXML Links. Das anschließende ebXML Glossar in Anhang E dient als Nachschlagewerk für ebXML Termini.

### 1.3 Formale und rechtliche Anmerkungen

Die vorliegende Arbeit beruht zum großen Teil auf den englischsprachigen Spezifikationen und Veröffentlichungen des ebXML Projekts. Sämtliche Literaturhinweise sind deshalb allein wegen der Übersetzung in die deutsche Sprache als sinngemäße Zitate zu verstehen. Aus Gründen der Einheitlichkeit wird diese Art der Quellenangabe auch für die wenigen deutschsprachigen Literaturquellen fortgeführt. Wörtliche Zitate sind in jedem Fall durch die veränderte Textformatierung und umschließende Anführungszeichen kenntlich gemacht.

Eine Quellenangabe am Ende eines beschreibenden Textabschnitts bezieht sich auf den gesamten Abschnitt, eventuell auch auf vorhergehende Abschnitte, sofern dort keine andere Quelle kenntlich gemacht ist. Abweichungen von dieser Formatierung sind beispielsweise zu Beginn eines größeren Textteils jeweils gesondert gekennzeichnet.

Einige der englischen Originalbegriffe wurden unverändert aus den ebXML Spezifikationen übernommen. Dies erfolgte teils wegen ihrer besonderen Bedeutung und spezifischen Definition in ebXML, teils wegen der prägnanten englischen Ausdrucksweise, die sich nur schwierig annähernd kurz, einfach und präzise in der deutschen Sprache wiedergeben lässt. Solche Begriffe sind durch *kursive Schreibweise* hervorgehoben. Alle kursiv gesetzten Ausdrücke sind im anhängenden ebXML Glossar gesammelt und zusätzlich zur Erklärung im laufenden Text nochmals kurz erläutert. Das Glossar dient somit als allgemeines Nachschlagewerk für die ebXML Terminologie. Gängige, meist englische Fachtermini und Abkürzungen aus dem Bereich der Informationstechnologie, wie z.B. Software, HTTP oder FTP, werden als bekannt vorausgesetzt und nicht eigens erläutert. Alle Akronyme erscheinen im Abkürzungsverzeichnis zu Beginn der Arbeit.

Angegebene Auszüge von XML Dateien und daraus direkt in den laufenden Text übernommene `<Elemente>` oder `attribute` sind durch eine veränderte Schriftart hervorgehoben.

Die Spezifikationen des ebXML Projekts sind im Internet öffentlich unter <http://www.ebxml.org/specs> zugänglich. Jede dieser Spezifikationen unterliegt folgender Copyright-Bestimmung:

Copyright © UN/CEFACT and OASIS, 2001. All Rights Reserved.

This document and translations of it may be copied and furnished to others, and derivative works that comment on or otherwise explain it or assist in its implementation may be prepared, copied, published and distributed, in whole or in part, without restriction of any kind, provided that the above copyright notice and this paragraph are included on all such copies and derivative works. However, this document itself may not be modified in any way, such as by removing the copyright notice or references to ebXML, UN/CEFACT, or OASIS, except as required to translate it into languages other than English.

The limited permissions granted above are perpetual and will not be revoked by ebXML or its successors or assigns. This document and the information contained herein is provided on an "AS IS" basis and ebXML DISCLAIMS ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTY THAT THE USE OF THE INFORMATION HEREIN WILL NOT INFRINGE ANY RIGHTS OR ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE

## **2 EbXML Systemübersicht und Grundlagen**

Das folgende Kapitel liefert anhand eines Anwendungsbeispiels einen ersten Überblick, wie ein Geschäftsablauf in ebXML funktionieren kann und mit Hilfe welcher ebXML Komponenten diese Funktionalität gewährleistet wird. Danach wird die allgemeine Architektur eines ebXML Systems vorgestellt, die in zwei unterschiedliche Sichtweisen getrennt ist, um eine höhere Flexibilität von Systemen im e-Business zu ermöglichen. Auf der Grundlage eines dieser Views werden die für die Durchführung eines ebXML Szenarios benötigten Phasen erläutert.

Als Grundlage für das Vorgehensmodell einer ebXML Implementierung und der Modellierung von ebXML Geschäftsabläufen schließt die Vorstellung der Unified Modelling Language (UML) dieses Kapitel ab.

### **2.1 EbXML Anwendungsbeispiel**

Die folgende Abbildung 2-1 beschreibt stark vereinfacht ein Anwendungsszenario zwischen zwei Geschäftspartnern, die zunächst ihre Systeme konfigurieren, dann eine einfache Geschäftstransaktion und den dazugehörigen Datenaustausch durchführen. Anhand dieses Szenarios sollen die einzelnen Prozesse und Schritte aufgezeigt werden, die für Konfiguration und Einsatz von ebXML Anwendungen und der jeweiligen Systemkomponenten nötig sind. [ebTA S.7-9]

Dieses einfache Beispiel dient als Einstieg in ebXML und kann etwa um mehrere Partner mit gemeinsamen Geschäftsprozessen und Datenaustausch oder um ein Service-Portal für Geschäftsprozesse erweitert werden. [ebTA S.35-39]