XML-Standards im Business Networking – Grundlage für Business Collaboration Infrastructures

Fabian Erni, Florian Leser

Bericht Nr.: BE HSG/ CCBN/ 18
Lehrstuhl: Prof. Dr. H. Österle
Version: 1.0
Datum: 2. Dezember 2001

**Universität St. Gallen -
Hochschule für Wirtschafts-, Rechts- und Sozialwissenschaften (HSG)**

Institut für Wirtschaftsinformatik
Müller-Friedberg-Strasse 8
CH-9000 St. Gallen
Tel.: ++41 / 71 / 224 2420
Fax: ++41 / 71 / 224 2777

Prof. Dr. A. Back
Prof. Dr. W. Brenner
Prof. Dr. E. Fleisch
Prof. Dr. H. Österle
Prof. Dr. R. Winter (geschäftsführend)

Inhaltsverzeichnis

Abkürzungsverzeichnis vii

1 Einleitung 1

1.1. Ausgangslage 1

1.2. Zielsetzung und Aufbau 2

1.3. Abgrenzung 2

2 Einführung in die Thematik der Standards 3

2.1. Standards im Allgemeinen 3

2.2. Bedeutung von Standards für den elektronischen Geschäftsverkehr 5

2.3. EDI als Vorreiter 6

2.4. XML als Nachfolger von EDI 7

2.5. Standards im Geschäftsmodell des Informationszeitalters 8

3 Strukturierung von XML Standards 10

3.1. Sprachen, Repositories und Frameworks 11

3.2. Fokussierung auf Industrie- oder Dienstleistungssektor 14

3.3. Objekte des Business Networking 15

3.4. Kooperationsprozesse des Business Networking 16

3.4.1 Content and Community 17

3.4.2 Product Life Cycle 17

3.4.3 Commerce 17

3.4.4 Supply Chain 18

3.4.5 Maintenance and Repair 18

3.4.6 Finance 18

3.5. Die vier Dimensionen im Kontext 19

3.6. Aufbau und Struktur einer XML-Standards-Datenbank 19

4 Analyse ausgewählter B2B Standards 21

4.1. Electronic Business XML 21

4.1.1 Spezifikation von ebXML 22

4.1.2 Zukunftsaussichten für ebXML 25

4.2. RosettaNet 26

4.2.1 Standardisierungsinitiative von RosettaNet 26

4.2.2 Zukunftschancen von RosettaNet 27

4.3. Commerce eXtensible Markup Language 28

4.3.1 Spezifikation von cXML 28

4.3.2 Beurteilung und Zukunftsaussichten für cXML 30

4.4. Common Business Library 30

4.4.1 Spezifikation von xCBL 30

4.4.2 Zukunftsaussichten für xCBL 34

4.5. Simple Object Access Protocol 34

4.5.1 Spezifikation von SOAP 35

4.5.2 Beurteilung und Zukunftsaussichten für SOAP 37

4.6. BizTalk 38

4.6.1 Spezifikation von BizTalk Framework 39

4.6.2 Zukuftsaussichten für BizTalk 42

4.7. Business Transaction Protocol 42

4.7.1 Spezifikation von BTP 42

4.7.2 Zukunftschancen für BTP 45
4.8. Customer Profile Exchange 46
4.8.1 Spezifikation von CPExchange 46
4.8.2 Zukunftsaussichten für CPExchange 47
4.9. Universal Description, Discovery, and Integration 48
4.9.1 Aufbau von UDDI 48
4.9.2 Zukunftsaussicht für UDDI 49
4.10. Data Universal Numbering System 50
4.10.1 Spezifikation von DUNS 50
4.10.2 Zukunftsaussichten für DUNS 51
4.11. WebService Definition Language 51
4.11.1 Spezifikation der WSDL-Dokumente 51
4.11.2 Zukunftsaussichten von WSDL 53
4.12. BMEcat 53
4.12.1 Struktur eines BMEcat-Dokuments 53
4.12.2 Zukunftsaussichten für BMEcat 55
4.13. eCl@ss 55
4.13.1 Spezifizierung von eCl@ss 56
4.13.2 Zukunftsaussichten für eCl@ss 57
4.14. Zuordnung ins Standard-Modell 57
5 Beispiel: Web Services und Business Collaboration Infrastructure in der Telekommunikationsbranche 60
5.1. Kundenservice der Phonecom 61
5.2. Ausgangssituation: Prozess- und Systemarchitektur 61

5.3. Zielsetzung: Prozess- und Systemarchitektur 62

5.4. Exkurs: Bea WebLogic Integration 64

5.5. Weitere Projektschritte 65

5.6. Einsatzmöglichkeiten von XML-Standards 65

6 Fragen des Managements 67

6.1. Strategische Fragen 67

6.2. Fragen an Software-Anbieter 68

6.2.1 Fragen zur Unternehmung 68

6.2.2 Fragen zum Produkt 68

6.2.3 Fragen zu den Kosten 71

6.3. Fazit 71

7 Zusammenfassung und Ausblick 71

7.1. Zusammenfassung 72

7.2. Ausblick 73

Anhang A: B2B-Standards im Modell 75

Anhang B: Kontaktpersonen aus der IT-Branche 76

Literatur 77

Abkürzungsverzeichnis

ANSI	American National Standard Institute
ASP	Active Server Page
B2B	Business-to-Business
BCI	Business Collaboration Infrastructure
BFC	BizTalk Framework Compliant
BIPS	Bank Internet Payment System
BME	Bundesverband Materialwirtschaft, Einkauf und Logistik
CAD	Computer Added Design
CIML	Customer Information Markup Language
CPA	Collaborative Partner Agreement
CPExchange	Customer Profile Exchange
CPFR	Collaborative Planning, Forecasting and Replenishment
CRM	Customer Relationship Management
cXML	Commerce XML
DIN	Deutsches Institut für Normierung
DTD	Document Type Definition
DUNS	Data Universal Numbering System
EAI	Enterprise Application Integration
ebXML	Electronic Business XML
EC	Electronic Commerce
EDI	Electronic Data Interchange
EDIFACT	EDI for Administration, Commerce and Transport
FTC	Federal Trade Commission
GUI	Graphical User Interface
HRML	Human Resource Markup Language
HTML	HyperText Markup Language
HTTP	HyperText Transfer Protocol

ID	Identifikation (-snummer)
ISO	International Organization for Standardization
J2EE CA	Java 2 Enterprise Edition Connector Architecture
KMU	Klein- und Mittelunternehmen
MIME	Multipurpose Internet Message Extensions
OASIS	Organisation for the Advancement of Structured Information Standards
OFX	Open Financial Exchange
P3P	Platform for Privacy Preferences Project
PDML	Product Data Markup Language
PIP	Partner Interface Processes
PLC	Product Life Cycle
RFI	Request For Information
RFS	Ready For Service
RNIF	RosettaNet Implementation Framework
RPC	Remote Procedure Call
SGML	Standard Generalized Markup Language
SOAP	Simple Object Access Protocol
TCP/IP	Transmission Control Protocol / Internet Protocol
UBR	UDDI Business Registry
UDDI	Universal Discovery, Description and Integration
UN/CEFACT	United Nations Centres for the Facilitation of Procedures and Practices for Administration, Commerce and Transport
URL	Unified Resource Locator
VAN	Value Added Network
W3C	World Wide Web Consortium
Web	World Wide Web
WLI	WebLogic Integration
xCBL	Common Business Library (auf XML basiert)
XML	eXtensible Markup Language
XSL	eXtensible Style Sheet Language

1 Einleitung

„It seems that hardly a week goes by without the formation of another vendor consortium or partnership tasked with standardizing some aspect of our business or technical environment. But there's been a marked increase in standardization efforts over the past few years, largely centered around XML."

(Leigh Dodds, Ingenta, 2000)

1.1. Ausgangslage

Im Zeitalter der Globalisierung spielen Standards für die betriebliche Verarbeitung und Integration von Informationen eine immer wichtigere Rolle. Seitdem das World Wide Web Consortium (W3C) vor etwas mehr als drei Jahren die Spezifikation der eXtensible Markup Language (XML) veröffentlicht hat, haben weitere Organisationen und Unternehmen dutzende von möglichen Standards entwickelt, welche auf der Metasprache XML basieren. Welche sich aber als wirkliche Standards auszeichnen werden, wird sich erst in den nächsten Jahren abzeichnen. Einige von ihnen haben sich bereits in mehreren Projekten etabliert und dadurch einen grösseren Bekanntheitsgrad erreicht. Ihnen wird gewiss eine grössere Durchsetzungswahrscheinlichkeit als anderen zugesagt. Seit diesem Boom der Standardisierungsvorhaben sind hunderte von Artikeln und Büchern erschienen, welche sich mit der Thematik von XML und den Weiterentwicklungen auseinandersetzen. Jedes Konsortium stellt zudem, in zum Teil umfassenden Whitepapers, ihren aktuellen Entwicklungsstand vor. Durch die immense Datenflut ist es kaum mehr möglich einen Überblick über die vielen Standards zu gewinnen.

Aus diesem Grund stellt sich das Management bei der Implementierung einer Lösung im Business-to-Business (B2B) Bereich die Frage, welche Standards überhaupt vorhanden sind, die das unternehmensspezifische Problem abbilden. Ist die Auswahl erstmals eingeschränkt, kann mit der Selektion des geeignetsten Standards begonnen werden. Aktuell haben jedoch beim Management die Entscheidungen zur Gewährleistung der internen Datenintegration eine höhere Prorität. Ist diese gesichert, kann die Auseinandersetzung mit den Standards für den Datenaustausch begonnen werden. Wie viele dieser Standards es in Zukunft geben wird und welche Bedeutung sie einnehmen werden, wird sich in den nächsten Jahren verdeutlichen.

© HSG / IWI / CCBN / 18

1.2. Zielsetzung und Aufbau

Der vorliegende Arbeitsbericht verfolgt das Ziel einen Überblick über die XML-basierten Standardisierungsvorhaben zu verschaffen. Zusätzlich wurde ein Modell mit vier Dimensionen entwickelt, in welche sich die Standards zuordnen lassen. Durch die Erfassung dieser in einer Datenbank wird es dem Management ermöglicht, den geeigneten Standard für ein unternehmensspezifisches Problem zu finden. Ferner wird mittels eines Praxisbeispiels aufgezeigt, dass sich das Management bei der Implementierung einer Business Collaboration Infrastructure mit vielen weiteren Fragen auseinandersetzen muss, als nur mit der Auswahl eines geeigneten Standards.

In Kapitel 2 wird zunächst die Entwicklung und die Notwendigkeit von Standards im Geschäftsverkehr des Informationszeitalters dem Leser näher gebracht. Im dritten Kapitel wird ein Modell vorgestellt, in welches sich die aktuellen Standardisierungsvorhaben einordnen lassen. Anhand dieses Modells lässt sich eine Datenbank erfassen, welche einem Management ermöglicht, den richtigen Standard zu finden. In Kapitel 4 werden aktuelle Standards vorgestellt und in das oben erwähnte Modell eingeordnet. Dabei handelt es sich um eine Auswahl von Standards, welche anhand ihrer Aktualität und Popularität ausgewählt wurden. Anschliessend wird in Kapitel 5 ein Praxisbeispiel aus der Telekommunikationsbranche erörtert und die Verwendung von XML-Nachrichten aufgezeigt. Dieses Beispiel veranschaulicht, dass die Auseinandersetzung mit den XML-Standards aktuell nur einen Teil der ganzen Problematik ausmachen. Die Schlussfolgerung daraus ist, dass sich das Management bei der Implementierung einer B2B-Lösung eine Reihe weiterer Fragen stellen muss, die sich nicht ausschliesslich auf die XML-Standards beziehen. Notwendige Aspekte für die Erstellung eines Fragenkatalogs sind in Kapitel 6 übersichtlich dargestellt. Abschliessend wird im siebten Kapitel ein Ausblick auf die Entwicklung der Standardisierungsvorhaben und die künftige Thematik von XML gegeben.

1.3. Abgrenzung

Um den Umfang dieses Arbeitsberichts in Grenzen zu halten, wurden folgende Abgrenzungen vorgenommen:

- Es wird vorausgesetzt, dass der Leser sich bereits mit der Thematik von XML auseinandergesetzt hat und somit über die erforderlichen Grundkenntnisse verfügt.

© HSG / IWI / CCBN / 18

- Da die Zahl der Standards, welche auf XML basieren, äusserst umfangreich ist, wurde in Kapitel 4 nur eine ausgewählte Anzahl der wichtigsten Standards umschrieben.
- Das Beispiel aus der betrieblichen Praxis der PHONECOM[1] veranschaulicht ausschliesslich einen Bestellprozess, welcher Teil des gesamten Customer Relationship Managements (CRM) ist.
- Die in Kapitel 6 aufgeführten Management-Fragen sind nur von allgemeinem Charakter. Bei der Implementierung einer B2B-Lösung muss der Fragekatalog unternehmensspezifisch ausformuliert werden.

Aufgrund der Aktualität dieses Themas basiert die Informationsbeschaffung einerseits auf der klassischen Literaturrecherche mit Integration des Internets und andererseits auf einer hohen Interaktion mit Vertretern aus der IT-Branche. Da durch die Übersetzung englischer Begriffe der Informationsgehalt nicht gewährleistet ist, wurden zur Verständlichkeit bewusst Anglizismen verwendet.

2 Einführung in die Thematik der Standards

> „*International Standards thus contribute to making life simpler, and to increase the reliability and effectiveness of the goods and services we use." [ISO 2001a]*

2.1. Standards im Allgemeinen

Standards sind dokumentierte Vereinbarungen, welche technische Spezifikationen oder exakte Kriterien enthalten. Sie dienen als Vorschriften, Richtlinien oder als Definitionen von Eigenschaften und sollen sicherstellen, dass Materialien, Produkte, Prozesse und Dienstleistungen ihren Zweck verfolgen oder erfüllen [vgl. ISO 2001a]. Standards sollen dazu beitragen unser Leben leichter zu machen. So wurde zum Beispiel von der Internationalen Standard Organisation (ISO) das Format von Kreditkarten und Telefonkarten festgelegt. Die Einheitlichkeit ermöglicht den weltweiten Gebrauch solcher Karten [vgl. ISO 2001a].

Standards können auf zwei verschiedene Arten zustande kommen. Zum einen entstehen sie sehr unformal, wenn eine Firma ein Produkt oder bestimmte Merkmale entwirft, die dann so weit verbreitet werden, dass eine Abweichung von dieser Norm Kompatibilitätsprobleme oder eine eingeschränkte Vermarktung zur Folge hat. Somit entstehen die sogenannten DeFacto-

[1] Name vom Autor geändert

Standards wie z. B. die Microsoft-Produktgruppe. Zum zweiten entstehen Standards über einen sehr formalen Prozess. Durch eine Arbeitsgruppe oder ein Komitee werden die verschiedenen bestehenden Methoden, Verfahren, technologischen Trends und Entwicklungen intensiv untersucht. Das Komitee schlägt dann den Standard einer anerkannten Organisation – wie ISO, OASIS, DIN oder ANSI – vor. Am Ende des Prozesses entscheidet die Organisation über die Annahme des Standards [vgl. Encarta 2000b]. Dabei ist es wichtig, dass die entwickelten Standards offen sind. Dies bedeutet, dass sie von allen interessierten Unternehmen implementiert werden können und keine Abhängigkeit bei der Technologie der Geräte besteht. Im Allgemeinen sind offene Standards die Voraussetzung, dass unterschiedliche Geräte über verschiedene Netzwerke miteinander funktionieren [vgl. Convergence 2001].

Standards haben schon im Mittelalter eine wichtige Rolle gespielt. So waren damals die Mengen- und Masseinheiten, die Währung oder die Adressierung von Strassen bereits vereinheitlicht. Im Zeitalter der Globalisierung spielen Standards jedoch eine immer wichtigere Rolle, damit eine internationale Kommunikation und Kompatibilität ermöglicht wird. Standards sollen schliesslich unser tägliches Leben vereinfachen. Aktuelle Themen sind zum Beispiel die Einführung des Euros in der EU oder die Lancierung der Internetzeit von Swatch. Dabei wird ein Tag in 1000 Swatch ‚.beats' zerlegt. Ein neuer Tag beginnt bei null, um Mitternacht bei zentraleuropäischer Winterzeit. Diese neue Zeiteinheit soll die Zeitzonen und die geografischen Grenzen aufheben. Somit können sich zwei Internet-Surfer aus New York und Rom zu einem bestimmten Zeitpunkt, zum Beispiel @625 Swatch .beats, in ihrem Internet-Forum treffen [vgl. Swatch 2001] ohne sich mit Zeitumrechnung zu beschäftigen. Um diesen zur Zeit DeFacto-Standard zu etablieren wird Swatch Partnerschaften mit relevanten Organisation und Unternehmungen, welche von der Internetzeit und ihren Funktionen profitieren können, eingehen [vgl. Swatch 2001].

Internationale Standards sind bereits in vielen Technologien der vielfältigsten Industrie- und Dienstleistungssektoren wie Informationsbearbeitung, Nachrichtentechniken, Textilfabrikation, Verpackung, Logistik, Energieherstellung und –nutzung, Schiffsbau oder Finanzdienstleistungen stark etabliert. Wir sind jedoch nur am Anfang dieser Globalisierung und viele weitere Standards werden in allen Sektoren der Industrie folgen [ISO 2001b].

© HSG / IWI / CCBN / 18

2.2. Bedeutung von Standards für den elektronischen Geschäftsverkehr

Standards beziehen sich auf technische Gegenstände, welche Teile eines marktlich koordinierten Austauschprozesses sind [vgl. Reimers 1995, S. 8]. Grundsätzlich lassen sich dabei zwei Gegenstände unterscheiden. Dies ist zum einen das Gut an sich und zum anderen der Geschäftsverkehr. Zum Geschäftsverkehr gehören Teilprozesse wie beispielsweise die Vertragsabwicklung, die Rechnungsstellung oder auch das Garantieversprechen.

Standards haben im täglichen Geschäftsverkehr einer Unternehmung eine grosse Bedeutung, vor allem dadurch, dass damit Transaktionskosten gesenkt werden können. Diese Senkung entsteht erstens dadurch, dass bei den beteiligten Akteuren eine hohe Vertrauenssicherheit über die Identität des Transaktionsgegenstandes entsteht und zweitens ein einseitiger Betrug verhindert wird [vgl. Reimers 1995, S. 57]. Ein Betrug kann beispielsweise durch die Änderung einer Währungseinheit vollzogen werden. Benutzen zwei Geschäftspartner einen Standard für den Bestellvorgang, so ist darin auch die Währungseinheit geregelt, und ein Betrug mit einem Währungswechsel ist ausgeschlossen.

Die in der Mikroökonomie verwendeten Skaleneffekte enthalten auch den sogenannten Erfahrungskurveneffekt. Dieser besagt, dass bei einer Verdoppelung des kumulierten Absatzes sich ein Kostenreduzierungspotential von zirka zwanzig bis dreissig Prozent ergeben [vgl. Felden 2000, S. 44]. Analog dazu kann auch durch die Standardisierung von Geschäftsprozessen, und den dadurch entstehenden Erfahrungen, eine Kostensenkung erreicht werden. Hingegen ist es bei Dienstleistungen schwieriger, diese zu Standardisieren, da sie meist kundenspezifisch sind.

Electronic Data Interchange (EDI) hat sich seit einigen Jahrzehnten als Standard für den elektronischen Geschäftsverkehr etabliert. EDI ermöglicht den elektronischen Austausch strukturierter Geschäftsdaten. Seit der Veröffentlichung von XML durch das W3C ist es nun auch möglich diese strukturierten Geschäftsdaten über das Internet zu versenden. Da XML eine Metasprache ist, sind einige Organisationen und Unternehmungen an der Entwicklung von spezifischen Dokumenten, welche zum Standard eines elektronischen Geschäftsverkehrs entwickelt werden. In den nächsten zwei Kapiteln wird kurz auf die Bedeutung dieser zwei Sprachen eingegangen.

© HSG / IWI / CCBN / 18

2.3. EDI als Vorreiter

Unter Electronic Data Interchange (EDI) wird der elektronische Austausch strukturierter Geschäftsdaten meist via proprietären Netzwerken verstanden. Unter strukturierten Geschäftsdaten werden alle Handelsdokumente subsumiert, welche heute bereits in Formularform zwischen Unternehmen per Papier weitergegeben oder ausgetauscht werden. Dies sind beispielsweise Rechnungen, Bestellungen, Offerten, Speditionsaufträge oder auch Kontoauszüge [vgl. Schmoll/Nommensen 1996, S. 13]. EDI wird bereits seit den späten 60er Jahren von Unternehmen genutzt und ermöglicht einen medienbruchlosen Informationsfluss zwischen verschiedenen Computersystemen [vgl. Weitzel et al. 2001, S. 6]. Dies bedeutet, dass nicht mehr der Mensch, sondern direkt die Maschine des Kommunikationspartners der Empfänger der EDI-Nachricht ist [vgl. Huemer 2001, S. 15].

Die Verwendung von EDI hat Einfluss auf drei Ebenen von Informationssystemen [Leser 2001, S. 6]. In den Geschäftsapplikationen werden Geschäftsobjekte wie beispielsweise eine Bestellung erzeugt, die dann an einen Handelspartner übermittelt werden sollen. Übersetzungsapplikationen konvertieren die Geschäftsnachrichten zu EDI-Nachrichten, welche dann mittels Kommunikationsapplikationen an den Handelspartner übermittelt werden. Zur Übermittlung solcher Nachrichten haben sich Verbände und Interessensgruppen zusammengeschlossen um eigene branchenspezifische Standards zu entwickeln [vgl. Schmoll/Nommensen 1996, S. 23]. Als branchenunabhängiger, internationaler Standard hat sich dabei EDIFACT durchgesetzt, aber auch der branchenübergreifende, vorwiegend in den USA verwendete, Standard ANSI X.12 hat eine grosse Akzeptanz erreicht[2].

Mit EDI ist es für Geschäftspartner möglich, direkt zwischen ihren Systemen zu kommunizieren und somit Prozesse zu automatisieren. Damit können die Kosten der einzelnen Dokumentverarbeitung reduziert werden, was wiederum eine geringere Durchlaufzeit und somit eine Just-in-Time Produktion ermöglicht. Typischerweise wird von branchenunabhängigen Kosteneinsparungen von fünf bis sechs Prozent des Umsatzes gesprochen [vgl. Weitzel et al. 2001, S. 7]. Obwohl EDI ein enormes Einsparungspotenzial aufweist, hat es nicht die erwartete Verbreitung gefunden. So sind es beispielsweise nur fünf Prozent der Unternehmen, bei welchen ein Einsatz von EDI vorteilhaft wäre, die tatsächlich EDI verwenden. Der Hauptgrund liegt darin, dass vornehmlich Grossunternehmen, die durch hohe Transaktionsvo-

[2] Weitere Informationen über EDI und dessen Standards sind in [Schmoll/Nommensen 1996, S. 22ff] oder [Ballnus 2000] zu finden.

lumina einen höheren Nutzen aus verbesserten Geschäftsprozessen realisieren können, EDI als Einsparung nutzten. Kleine und mittelständische Unternehmen (KMU) scheuen die erheblichen Setup und Betriebskosten [vgl. Weitzel et al. 2001, S. 7].

Weiter besitzt EDI Eigenschaften, die aus heutiger Sichtweise nicht mehr erforderlich sind. So werden Nachrichten kodiert um Speicherplatz zu sparen [vgl. Leser 2001, S. 5], was zu früheren Zeiten ein wichtiger Aspekt bei der Datenverwaltung war. In der heutigen Zeit kämpft jedoch niemand mehr damit, da zusätzlicher Speicherplatz günstig zu erwerben ist. Auch die Implementationszeit einer EDI-Lösung dauert gegenüber moderneren Lösungen viel länger. So muss bei der Einführung einer EDI-Lösung mit einer Projektzeit von mindestens neuen bis zwölf Monaten gerechnet werden. Vergleichsweise hat eine analoge Lösung mit XML hingegen eine Projektzeit von nur drei bis sechs Monaten [von Ammon 2001].

2.4. XML als Nachfolger von EDI

Dieses Unterkapitel ist keine Einführung in eXtensible Markup Language (XML), sondern soll nur die Vorteile des Datenaustausches mittels XML über das Internet veranschaulichen. Für ausführliche Informationen zu XML-Dokumenten und dessen Präsentation mittels der eXtensible Style Sheet Language (XSL) ist auf [Weitzel et al. 2001, Kapitel 2] oder [Fitzgerald 2001, Kapitel 2] verwiesen.

Die endgültige Version von XML wurde im Februar 1998 vom World Wide Web Consortium (W3C), welches ein internationales Industriekonsortium aus mehreren hundert Softwareherstellern ist, verabschiedet [vgl. Weitzel et al. 2001, S. 2, 10]. Dieser Standard ist eine Teilmenge der Standard Generalized Markup Language (SGML)[3] und definiert das Regelwerk zur Spezifikation von Markup-Sprachen für die Codierung bestimmter Dokumenten- beziehungsweise Nachrichtentypen. „Somit ist XML mehr als eine Sprache, wie beispielsweise HTML, sondern eine Metasprache“.[Huemer 2001, S. 17]

Da XML eine Metasprache ist, kann jeder eine neue Sprache in Form einer DTD oder als XML-Schema erzeugen. Die Einfachheit des XML-Regelwerks erlaubt es auch für technische Laien innerhalb kürzester Zeit ihre Nachrichtenformate zu entwickeln. Diese Flexibilität garantiert eine rasche Lösung um Daten auszutauschen. Bei der Nachrichtenstandardisierung von EDI-Dokumenten ist dies im Gegensatz ein langwieriger Prozess [vgl. Huemer 2001, S.

[3] Für ausführliche Information über SGML vgl. [McGrath 1998, S. 19ff, 361-392] oder [Möhr/Schmidt 1999].

18]. Da XML die Definitionen von Datenaustauschformaten erlaubt und es prinzipiell möglich ist DTDs zu erzeugen, die semantisch äquivalent zu einem EDI-Standard sind, liegt es auf der Hand, dass XML-basierte Sprachen für EDI genutzt werden können [vgl. Huemer 2001, S. 17]. Viele Unternehmen, welche bereits EDI verwendet haben, wechseln dabei ihre traditionellen Formate in neue XML-Sprachen [vgl. Weitzel et al. 2001, S. 11], um somit ihren Datenaustausch übers Internet abzuwickeln. Schliesslich kann durch die Datenübertragung über das Internet viel Geld gespart werden. Da bei traditionellem EDI die Nachrichten über ein Value Added Network (VAN) übermittelt wurden, entstanden dort hohe Kosten. So zahlte Mitte der 90er Jahre ein Unternehmen bei monatlich 25'000 EDI-Nachrichten zwischen 56 Cent bis zu einem US Dollar pro Nachricht an seinen VAN-Provider [vgl. Weitzel et al. 2001, S. 8]. Während bei VANs die Kosten pro Nachricht oder pro Zeichen berechnet wurden, gibt es eine solche Abrechnung bei einer Datenübermittlung über das Internet in der Regel nicht. Ein zusätzlicher Vorteil von XML-Nachrichten ist, dass sich nicht nur textbasierte Geschäftsdaten wie bei EDI übermitteln lassen. Auch Multimediadaten wie beispielsweise CAD-Daten oder Röntgenbilder lassen sich durch ungeparste Elemente und Notationen in eine XML-Nachricht einbinden [vgl. Huemer 2001, S. 18]. Einige Unternehmen verfolgen gar nicht das Ziel die Daten in Anwendungen zu integrieren, da es bei ihnen an automatisierbaren Prozessen mangelt. Sie verwenden XML lediglich als eine einfache Möglichkeit zur Datenanzeige im Browser [vgl. Weitzel et al. 2001, S. 9]. Mittels den XML Style Sheets kann eine solche Darstellung einfach erfolgen, während dies bei EDI-Daten kaum realisierbar ist.

Das Problem ist jedoch, dass die offene XML-Sprache lediglich die Datenrepräsentation festlegt, nicht aber die ganze Semantik des Datenaustauschs. Aus diesem Grunde sind zur Zeit einige Konsortien und Unternehmungen an der Entwicklung von neuen Standardsprachen, welche nicht nur die Syntax, sondern auch die Semantik definieren. Eine Auswahl solcher Standardisierungsvorhaben wird im vierten Kapitel näher vorgestellt.

2.5. Standards im Geschäftsmodell des Informationszeitalters

Das Geschäftsmodell des Informationszeitalters zeichnet sich dadurch aus, dass es nicht mehr produkt-, sonder kundenorientiert ist. Es muss den gesamten Prozess des Kunden bei seiner Problemlösung abdecken. Dieser Kundenprozess bestimmt die Produkte und Dienstleistungen, die ein Unternehmen anbieten muss. Alle diese Leistungen werden in einem Kundenprozessportal zusammengefasst und dem Kunden zur Verfügung gestellt [vgl. Österle

2000]. Ein Unternehmen muss sich dabei als Leistungsintegrator verstehen, der nicht alle Leistungen selbst erbringt, sondern aus vielen Quellen zukauft und integriert. Aus diesem Grunde werden Geschäftsnetzwerke entstehen, die so effizient zusammenarbeiten, wie die Prozesse innerhalb eines einzelnen Unternehmens [vgl. Österle 2001]. Eine Vorraussetzung für das Gelingen eines solchen Geschäftsnetzwerkes ist eine gemeinsame Informationsinfrastruktur. Diese Business Collaboration Infrastructure ist die „Verkehrsinfrastruktur“ des Informationszeitalters, welche jedem Unternehmen Zugriff in Echtzeit auf alle prozessrelevanten Informationen aller kooperierenden Unternehmen verschafft [vgl. Österle 2001]. Bild 2-1 zeigt die Business Collaboration Infrastructure mit ihren vier Ebenen und die darin definierten Standards auf.

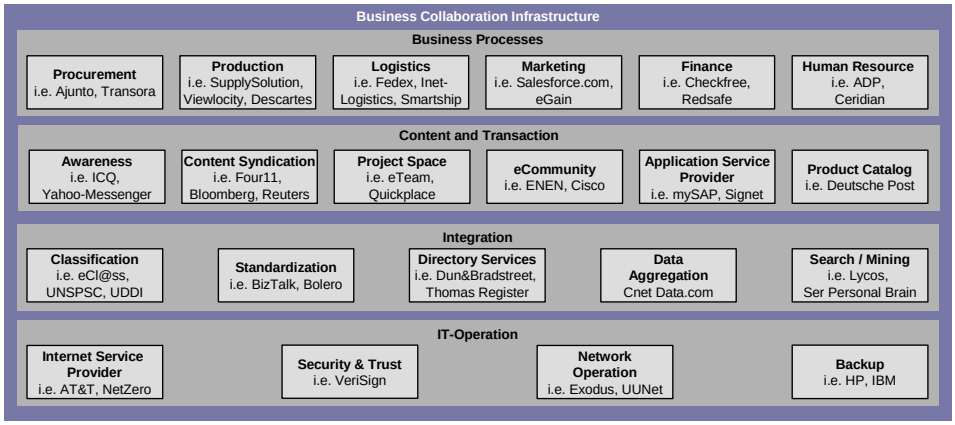


Bild 2-1: Business Collaboration Infrastructure [vgl. Österle 2001]

Damit eine Kooperation erfolgreich ist, werden in der Business Collaboration Infrastructure Standards definiert, die für alle Teilnehmer verbindlich sind [vgl. Österle 2001]. Tabelle 2-1 zeigt die Komponenten einer solchen Infrastruktur, die Bedeutung und dafür eingesetzte Standards auf.

Komponente	Beschreibung	Beispiele für eingesetzte Standards
Handelsvereinbarung	Vertragliche Bedingungen für die Zusammenarbeit der Teilnehmer werden geregelt. Dies sind beispielsweise die Anerkennung der Rechtsverbindlichkeit von elektronisch ausgetauschten Nachrichten oder Haftungsfragen bei Systemausfällen.	Bolero[4] CPA bei ebXML (s. Kapitel 4.1)
Prozesse	Beteiligte Unternehmen einigen sich auf einen gemeinsamen Prozess. Alle partizipierenden Unter-nehmen müssen den Prozess gleich leben, wenn er n:m-fähig sein soll.	ebXML (s. Kap. 4.1) RosettaNet (s. 4.2) CPFR[5]
Applikationen	Beteiligte Applikationen eines Kooperationspro-zesses müssen beispielsweise einen Prognosewert identisch interpretieren. Selbst bei gleichartigen Applikationen bleibt die Zusammenarbeit schwierig. So können beispielsweise bei mySAP Modulen durch das Customizing Probleme auftauchen.	Prozessspezifisch
Daten	Datenstruktur muss klar definiert und dokumentiert werden, damit sie auch austauschfähig sind. Voraussetzung dabei ist die Übersetzung von Daten eines Formates in ein anderes.	Alle Standards, die auf XML basieren. cXML (s. Kap. 4.3) xCBL (s. Kap. 4.4)
Informationstechnik	Ist ein technologischer Standard im Sinne von Middleware und Basis für oben genannte Standards.	TCP/IP, HTML, XML

Tabelle 2-1: Standardisierte Komponenten der Business Collaboration Infrastructure[6]

Standards sind für das Geschäftsmodell des Informationszeitalters zwingend notwendig, da es sich dabei um multinationale Netzwerke handelt. Die erforderlichen Standards definieren einheitliche „Spielregeln“ und eliminieren unterschiedliche nationale Regeln, welche als wettbewerbsverzerrend gelten. Durch Standards werden Unternehmen transparent und kompatibel gestaltet. Anbietern stehen mehrere potentielle Handelspartner für Transaktionen zur Verfügung, was zu einem erhöhten Wettbewerbsdruck führt. Dieser wiederum senkt die Preise, welche an das Unternehmen und den Endkonsumenten weitergegeben werden können [vgl. Storz 2001]. Ergo können Unternehmen durch Standards eine grössere Effizienz im Handel realisieren.

3 Strukturierung von XML Standards

Um die verschiedensten XML-Standards voneinander zu unterscheiden wird in diesem Kapitel ein Modell mit vier Dimensionen entwickelt. Durch dieses Modell wird es ermöglicht eine Datenbank zu erfassen, in welcher eine Firma nach einem Standard zu einem unternehmensspezifischen Bedürfnis suchen kann. Die erste Dimension unterscheidet die drei Ansätze

[4] Für weitere Informationen zu den Verträgen von Bolero ist auf www.softwareag.com/bolero verwiesen.

[5] CPFR ist eine Lösung für die Zusammenarbeit von Unternehmen der Konsumgüterindustrie. Für weiter Informationen zu CPFR ist auf www.cpfr.org verwiesen.

[6] Eigene Darstellung in Anlehnung an [Österle 2001, S. 29ff].

von XML-Sprache, XML-Repository und XML-Framework. Viele XML-Standards sind genau auf einen bestimmten Industrie- oder Dienstleistungssektor zugeschnitten worden, diese werden in der zweiten Dimension behandelt. Eine weitere Dimension setzt sich mit den drei Objekten Partner, Produkt und Kontrakt auseinander. Die letzte Dimension beschreibt die Kooperationsprozesse, für welche eigene XML-Standards geschaffen wurden. Zum Schluss werden diese vier Dimensionen im Standards-Modell miteinander verknüpft. Anhand diesem wurde dann eine Datenbank entwickelt, welche eine Suche nach einem Standard für ein unternehmensspezifisches Problem ermöglicht.

3.1. Sprachen, Repositories und Frameworks

Bei den Standardisierungsbestrebungen der XML-Community können im Wesentlichen drei konzeptionelle Ansätze unterschieden werden: XML-Sprachen, XML-Repositories sowie XML-Frameworks [vgl. Steffen 2001, S. 4ff]. Um die drei Elemente klarer abzugrenzen wird zuerst auf die drei Ebenen der Semiotik eingegangen.

Bei der Gestaltung der Integrationsbeziehungen zwischen Applikationen können drei Ebenen unterschieden werden, die sich parallel zu den Analyseebenen der Semiotik verhalten [vgl. Fleisch 2001, S. 257]. Der Begriff Semiotik bezeichnet die Lehre von Symbolen und Zeichen und kann in die drei Ebenen Syntax, Semantik und Pragmatik aufgespaltet werden [vgl. Steiner 1996].

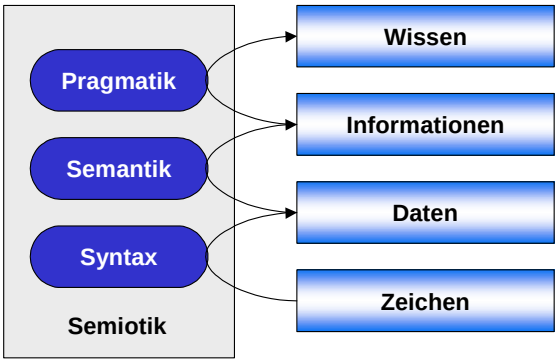


Bild 3-1: Ebenen der Semiotik (in Anlehnung an [Wolf et al. 1999, S. 748])

Bei der Repräsentation von Daten, Informationen und Wissen in Computersystemen findet man im allgemeinen keine Unterschiede bei der Darstellung, es werden immer Zeichen gespeichert und verarbeitet. Erst wenn die Zeichen durch die Syntax, Semantik und Pragmatik in Bezug gesetzt werden, führt dies zu einer Unterscheidung. So ist beispielsweise die Zahl

‚100' ein abstraktes Datum und besitzt eine Syntax, die besagt, dass Zahlen nur Ziffern, ein Komma und ein Vorzeichen besitzen dürfen. Erst durch das Hinzufügen der Semantik bekommen die Daten eine Bedeutung und die Beziehung zur Realität wird hergestellt. Wird dem Datum ‚100' nun eine Geschwindigkeitseinheit hinzugefügt, kann die Information ‚100 km/h' richtig interpretiert werden. Die Information alleine genügt jedoch noch nicht, um eine Handlung auszulösen. Erst durch die Pragmatik entsteht Wissen, welches für Entscheidungen benötigt wird. Im Beispiel würde das Erkennen der Information ‚100 km/h' eines Autofahrers auf dem Tachometer seines Autos und die Signalisierung einer Geschwindigkeitslimite den Fahrer zu der Entscheidung bringen, die Geschwindigkeit zu reduzieren [vgl. Wolf et al. 1999, S. 748].

Im B2B-Bereich haben sich für die drei Ebenen der Semiotik einige Standards für die zwischenbetriebliche Kommunikation durchgesetzt. Tabelle 3-1 ordnet verschiedene Standards den Ebenen der Semiotik zu.

Ebene der Semiotik	Standards
Pragmatik	CPFR Referenzmodell Frameworks wie ebXML oder RosettaNet
Semantik	zur Klassifikation von Leistungen: eCl@ss oder DUNS zur Beschreibung von Leistungen: EAN und diverse XML-Standards wie xCBL oder cXML
Syntax	XML, HTML, CORBA, SMTP oder TCP/IP

Tabelle 3-1: Beispiele für Kommunikationsstandards [vgl. Fleisch 2001, S. 132]

Seit der Veröffentlichung von XML, wird dieser Standard oft als Syntax für den zwischenbetrieblichen Datenaustausch verwendet [vgl. Bussler 2001, S. 7]. Aber auch auf Ebene der Semantik hat XML durch DTDs und XML-Schemata bei den verschiedensten Standardisierungsvorhaben eine dominante Stellung eingenommen, obwohl XML grundsätzlich nicht für semantische Bestimmungen entwickelt wurde [vgl. Bussler 2001, S. 4].

Beim ersten Ansatz, den XML-Sprachen, geht es darum, dass verschiedene Organisationen oder Unternehmungen versuchen einen festen Satz von XML-Tags oder DTDs als Industriestandard zu etablieren. Die XML-Sprachen werden dabei auf bestimmte Geschäftsprozesse zugeschnitten. Meist wurden diese Sprachen von einem Unternehmen oder einer kleinen Gruppe von Unternehmen für ein spezifisches Anwendungsgebiet entwickelt. Aus diesem Grunde herrscht eine Fülle von XML-Sprachen vor. XML-Sprachen befinden sich im Ebenenmodell der Semiotik auf der Stufe Semantik. Da in diesen Sprachen der semantische

Inhalt eines Dokuments definiert wird und dies keinen direkten Einfluss auf die Geschäftsprozesse hat, handelt es sich bei den Sprachen um Daten- oder Dokumentenstandards. Als Beispiele sind hier xCBL, cXML oder CPExchange zu nennen.

Der Ansatz von XML-Repositories besteht darin, die für verschiedene Anwendungszwecke gesammelten XML-Dokumente in Form von Spezifikationen der Tagsets, DTDs oder Style Sheets zur Weiterverwendung bereitzustellen. Bei den Repositories handelt es sich um einen standardisierten Zugriff auf bereits definierte XML-Dokumente. Hier wird lediglich die Möglichkeit geschaffen, spezifiziert XML-Dokumente für eine breite Anwendung zur Verfügung zu stellen. Jedes Standardisierungsvorhaben verwendet für Repository ihre eigene Bezeichnung, so werden dafür auch die Begriffe wie Dictionary oder Vocabulary[7] verwendet. Als Beispiele hierfür sind BizTalk von Microsoft oder XML.org zu nennen.

Bei der letzten Gruppe von Ansätzen, den XML-Frameworks, werden nicht nur die Nachrichtenformate für bestimmte geschäftliche Anwendungen festgelegt, wie es bei den XML-Sprachen geschieht, sondern auch die erforderlichen Abläufe und Regeln definiert. Damit verbunden ist häufig eine umfassende Modellierung unternehmensübergreifender Geschäftsprozesse. XML-Frameworks bestimmen somit Prozessstandards, welche nicht mehr nur Einfluss auf die Ebene der Semantik sondern vorwiegend pragmatischen Inhalt haben. Ein geeignetes Beispiel hierfür ist RosettaNet oder ebXML. Folgende Tabelle soll einen kurzen Überblick über die erste Dimension des Standards-Modells verschaffen.

XML-Community	
XML-Sprachen	- cXML (Commerce XML) - xCBL (Common Business Library) - CPExchange (Customer Profile Exchange)
XML-Repositories	- BizTalk von Microsoft - XML.org
XML-Frameworks	- RosettaNet - ebXML (Electronic Business XML)

Tabelle 3-2: Übersicht der Standardisierungsvorhaben der XML-Community[8]

[7] Vorsicht ist geboten bei Standardisierungsvorhaben welche den Begriff Registry verwenden. Bei ihnen handelt es sich meist nur um eine Datenbank, bei welcher die Informationen eines Dokumentes erfasst werden. So werden beispielsweise im Repository von UDDI die Unternehmensdaten abgespeichert. Dies hat jedoch nichts mit einem Repository zu tun, bei dem verschiedene Geschäftdokumente abgelegt sind.

[8] Tabelle ist eine leicht abgeänderte Version des Originals aus [Steffen 2001, S. 5].

Auf den ersten Blick entsteht der Eindruck, dass es sich dabei um eine Verschachtelung der drei Ansätze handelt. Das Framework legt die geschäftliche Anwendung von verschiedenen Dokumenten, welche in einem Repository abgelegt sind, fest. Die Dokumente selbst sind in einer standardisierten XML-Sprache geschrieben. Dem ist jedoch nicht so, denn die drei Ansätze müssen isoliert betrachtet werden. So besitzt beispielsweise BizTalk sowohl ein Repository als auch ein Framework, jedoch keine eigene Sprache[9]. ebXML hingegen besitzt alle drei Ansätze. Zur Verdeutlichung dieser Thematik wird in Bild 3-2 ein Metamodell vorgestellt.

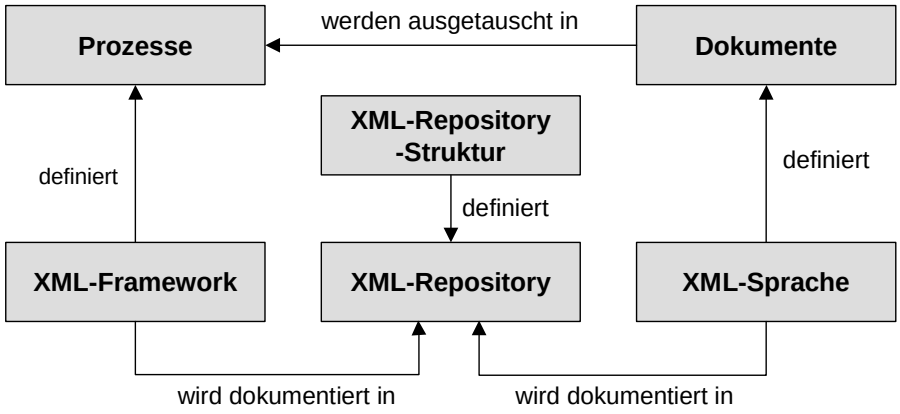


Bild 3-2: Metamodell von XML-Sprache, -Repository und –Framework

Durch dieses Metamodell wird aufgezeigt, wie die Elemente der ersten Dimension miteinander in Verbindung stehen. Nun wird auf die weiteren drei Dimensionen eingegangen, welche ebenfalls als Grundlage für die Entwicklung des Standards-Modell dienen.

3.2. Fokussierung auf Industrie- oder Dienstleistungssektor

Viele XML-Standards werden oft für einen bestimmten Industrie- oder Dienstleistungssektor geschaffen. Diese sollen die spezifischen Geschäftsprozesse oder die Produktkatalogisierung des jeweiligen Sektors unterstützen. Die hier verwendet Sektoren stammen aus dem Repository von xml.org. Weiter gibt es aber auch XML-Standards, die keine Fokussierung

[9] Im Repository von BizTalk kann jeder seine selbstdefinierten Dokumente ablegen, aber auch Dokumente von anderen Standardisierungsvorhaben wie beispielsweise xCBL sind darin enthalten.

aufweisen. Folgende Tabelle soll einen Überblick über die verschiedenen Industrie- und Dienstleistungssektoren und ihre unterstützenden Standards verschaffen.

Industrie- oder Dienstleistungssektor	Spezifischer XML-Standard
Ausbildung / Schule	- Schools Interoperability Framework (SIF)
Autoindustrie	- Automotive Network eXchange (ANX)
Banken	- Bank Internet Payment System (BIPS)
Chemieindustrie	- Chemical Markup Language (CML)
Energieversorgung	- PetroXML
Gesetz	- Legal Electronic Data Exchange Standard (LEDES)
Gesundheitswesen	- Clinical Infosystems Interoperability Network (CISTERN)
Informationstechnologie	- Electronic Component Manufacturer Data Sheet (ECMData)
Medien	- XML for Press and Printers (XPP)
Raumfahrt	- Spacecraft Markup Language (SML)
Transport & Logistik	- Marine Trading Markup Language (MTML)
Telekommunikation	- Telecommunication Interchange Markup (TIM)
Versicherungen	- iLingo for Insurance
Wissenschaften	- Mathematical Markup Language (MathML)

Tabelle 3-3: Industrie- und dienstleistungssektorspezifische XML-Standards[10]

Da mit der Schaffung von XML-Standards erst seit wenigen Jahren begonnen wurde, werden weitere Industrie- und Dienstleistungssektoren, wie beispielsweise die Beratung, hinzukommen. Es besteht auch die Möglichkeit einen Sektor in verschiedene Untergruppen aufzuteilen. So kann bei starkem Wachstum zum Beispiel der Sektor Wissenschaften unterteilt werden in Geografie, Biologie, Mathematik etc..

3.3. Objekte des Business Networking

Die dritte Dimension des Standards-Modells unterscheidet die drei Objekte: Partner, Produkt und Kontrakt. Bezüglich diesen drei Objekten werden beim zwischenbetrieblichen Datenaustausch Informationen versendet. Zahlreiche XML-Standards wurden oder werden speziell für eines dieser Objekte geschaffen.

[10] Die Industrie- und Dienstleistungssektoren stammen aus xml.org und können sich bei einem starken Wachstum an XML-Standards weiter spezifizieren.

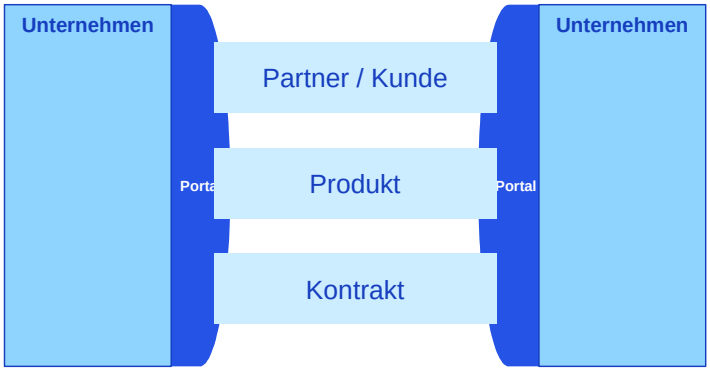


Bild 3-3: Objekte des Business Networking

Das Objekt Partner enthält sämtliche zwischenbetriebliche Daten über Geschäftspartner - sowohl Kunden als auch Lieferanten. Diese Informationen können allgemeiner Art sein, wie Branchen, Name, Adresse etc., aber auch spezifische Erläuterungen enthalten wie zum Beispiel die Anwendungssoftware oder die verwendeten Schnittstellen des Unternehmens. UDDI ist beispielsweise ein solcher XML-Standard, der in diese Kategorie fällt.

Sämtliche Informationen über die Produkte, wie die Produktdimensionen, Produkteigenschaften oder auch die Lagerbestände werden im Objekt Produkt zusammengezogen. In diesem Bereich beschäftigen sich zur Zeit viele Organisationen und Firmen mit einer Lancierung eines geeigneten XML-Standards. Im deutschsprachigen Raum ist vor allem BMEcat zu nennen, welcher als Standard für die Beschreibung und den Austausch von Produktkatalogdaten entwickelt wurde.

Zum Objekt Kontrakt gehören diejenigen Datenaustausche, die mit einer Leistung in Verbindung stehen. Vorwiegend sind dies Daten zur Vertragsabwicklung, also meist beginnend mit einer Offerteanfrage über die Lieferung bis hin zum Finanzausgleich. Da die Vertragsabhandlungen meist unternehmensspezifisch erfolgen, sind vorwiegend umfangreiche XML-Standards wie ebXML, xCBL oder cXML in diese Kategorie einzuordnen.

3.4. Kooperationsprozesse des Business Networking

XML-Standards treten vorwiegend bei Kooperationsprozessen des Business Networkings auf. Das Augenmerk gilt den Makroprozessen, welche in idealtypischer Weise definiert und daher in allen Unternehmen ähnlich sind [vgl. Alt et al. 2001, S. 20]. Kooperationsprozesse beschreiben den zwischenbetrieblichen Leistungsfluss der beteiligten Geschäftspartner und lassen sich gemäss [Fleisch/Österle 2001] in sechs Makroprozesse unterteilen.

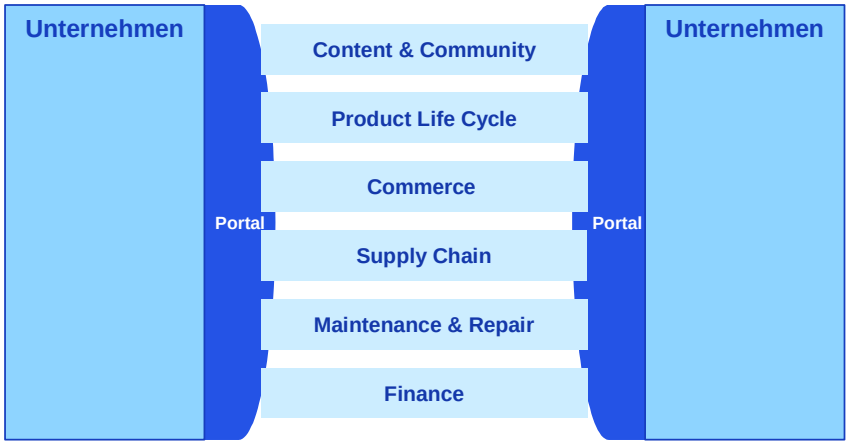


Bild 3-4: Die sechs Kooperationsprozesse des Business Networking [Alt et al. 2001, S. 29]

Nachfolgend werden diese sechs Kooperationsprozesse kurz erläutert und ihnen spezifische XML-Standards zugeordnet.

3.4.1 Content and Community

Im Makro-Kooperationsprozess Content & Community sind sämtliche Teilprozesse untergeordnet, bei denen ein zwischenbetrieblicher Austausch von betriebsinternen Informationen als auch Daten über das Umfeld einer Unternehmung stattfindet. In diese Kategorie fällt somit zum Beispiel der XML-Standard Human Resource Markup Language (HRML), welcher es ermöglicht, Rekrutierungsdaten zwischen verschiedenen Firmen auszutauschen.

3.4.2 Product Life Cycle

Der Kooperationsprozess Product Life Cycle (PLC) fasst den zwischenbetrieblichen Produkt-Datenaustausch zusammen, von der Entwicklung bis zur Entsorgung der Güter. Mikroprozesse sind zum Beispiel der Datenaustausch von Forschungs- und Entwicklungsdaten oder auch der Austausch von Informationen zur Mängel- oder Fehlerbeseitigung. Ein spezifischer XML-Standard für den PLC-Prozess ist die Product Data Markup Language (PDML). Mit einem einzigen, abstrakten Vokabular vereinfacht dieser Standard den Austausch von Produktinformationen.

3.4.3 Commerce

Sämtliche XML-Standards die für den Datenaustausch beim Verkauf oder bei der Verhandlung geschaffen wurden, gehören zum Kooperationsprozess Commerce. Mikroprozesse von Commerce sind beispielsweise die Erstellung eines Produktkataloges, der Informationsaus-

tausch über Lieferkonditionen sowie auch die Suche nach neuen Lieferanten. Spezifische XML-Standards die in diesen Kooperationsprozess fallen sind zum Beispiel eCl@ss, welches ein Standard für Materialklassifikation und Warengruppen ist, oder der XML-Standard UDDI (Universal Discovery, Description and Integration), welcher es ermöglicht einen geeigneten Lieferanten zu suchen und zu finden.

3.4.4 Supply Chain

Der Kooperationsprozess Supply Chain beinhaltet sämtliche Teilprozesse, die sich mit Rohmaterial-, Halbfabrikat- oder Produktbeschaffung, über die Produktion oder Montage bis zur Auslieferung an den Lieferanten oder Endkunden befassen. Da der zwischenbetriebliche Datenaustausch meist unternehmen- oder branchenspezifisch ist, eignen sich in der Supply Chain vorwiegend umfangreichere XML-Standards wie xCBL, cXML oder ebXML. Laut xml.org sind bereits Gespräche für die Schaffung eines eigenen Standards für die Logistik im Gange. Dieser wird eine Optimierung der Routeplanung als auch der Transportbelegung ermöglichen, und somit die betrieblichen Kosten senken.

3.4.5 Maintenance and Repair

Sämtliche After-Sales-Aktivitäten, wie Kundendienst, Garantieleistungen oder Reparaturen, sowie die Bearbeitung dieser Leistungen fallen in die Kategorie von Maintenance and Repair. Da in diesen Bereichen das Customer Relationship Management (CRM) eine wichtige Rolle spielt, treten hier vorwiegend XML-Standards auf, welche speziell für das CRM konzipiert wurden. Als Beispiel ist hier der Standard CIML zu nennen, mit welchem allgemeine Kundendaten erfasst werden. Noch weiter geht CPExchange, in welchem speziell auch Call Center Daten erfasst werden und ein Sales Tracking ermöglicht wird.

3.4.6 Finance

In den Kooperationsprozess Finance gehören alle Teilprozesse der zwischenbetrieblichen Zahlungsabwicklung. Ein Zahlungsprozess beginnt meistens mit der Rechnungsstellung und endet mit einer Einzahlungsbestätigung. Durch die grossen Vorteile von Zeit und Kosten wurde in diesem Bereich schon früh mit dem Erarbeiten von XML-Standards begonnen. Zusätzlich unterstützen viele Banken die Schaffung solcher Standards, da sie durch eine einheitliche Vernetzung stark profitieren. Da jedoch die meisten Klein- und Mittelunterneh-

men bereits eine EDI-Lösung verwenden, haben sich XML-Standards wie BIPS (Bank Internet Payment System) oder OFX (Open Financial Exchange) noch nicht stark durchgesetzt.

3.5. Die vier Dimensionen im Kontext

Bild 3-4 stellt nun diese vier Dimensionen in einen Kontext. Die Dimensionen Industriefokus, Objekte und Prozesse beschreiben Gebiete, für welche spezielle XML-Standards geschaffen wurden. Diese verschiedenen XML-Sprachen finden sich meist in einem Repository wie zum Beispiel xml.org. Auffällig ist dabei, dass XML-Sprachen, die speziell auf einen Industriesektor zugeschnitten wurden, zwar in einem Repository enthalten sind, meist aber kein Framework besitzen. Auf der anderen Seite befinden sich XML-Sprachen, die auf einen bestimmten Prozess ausgelegt sind, auch in einem Repository, sind aber meist zusätzlich Teil eines Frameworks. In der Mitte stehen noch die Sprachen, welche sich auf die Objekte beziehen. Objekte wie Produkt und Partner sind nur gelegentlich Teil eines Frameworks, wobei das Objekt Kontrakt stärker nach rechts tendiert.

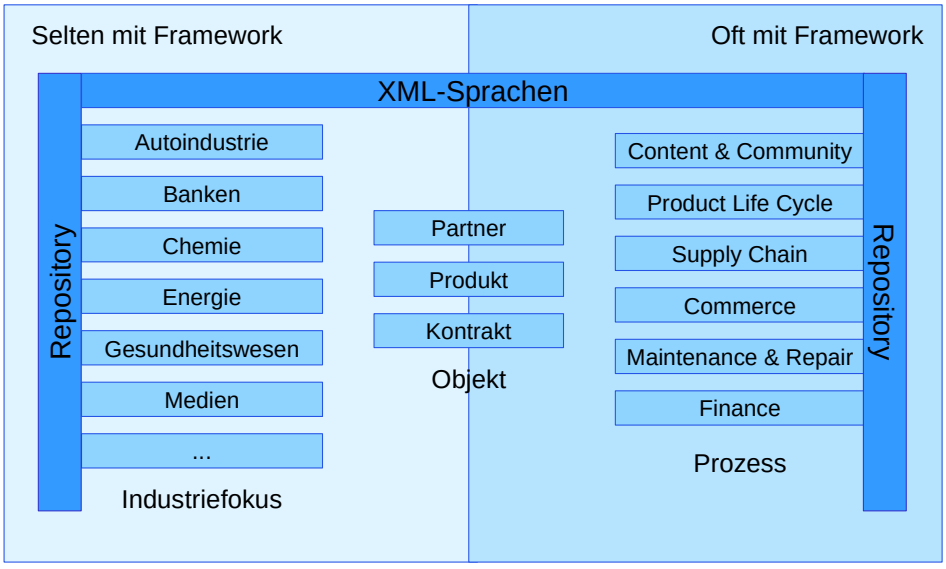


Bild 3-5: Die vier Dimensionen im Kontext[11]

3.6. Aufbau und Struktur einer XML-Standards-Datenbank

Das in diesem Kapitel entwickelte Standards-Modell ermöglicht nun die Erstellung einer Access Datenbank, worin alle zur Zeit entwickelten XML-Standards erfasst werden. Mittels

[11] Eigene Darstellung

dieser Datenbank wird es den Entscheidungsträgern bei der Implementierung einer Business Collaboration Infrastructure ermöglicht, den richtigen Standard für ein unternehmensspezifisches Problem zu finden. Jeder Datensatz enthält allgemeine Informationen über den Namen, den Operator und die teilnehmenden Unternehmen eines Standards. Zudem wird nun eine spezifische Suche durch die vier Dimensionen konzeptioneller Ansatz, Industriefokus, Objekt und Prozess des Standards-Modells mittels Anklicken ermöglicht. Bild 3-6 zeigt das Graphical User Interface (GUI) auf.



Bild 3-6: Access Standards-Datenbank

Eine Suche ist über allen vier Dimensionen hinweg möglich. So kann beispielsweise ein Chemieunternehmen nun einen geeigneten XML-Standard für einen Produktkatalog oder ein Telekommunikationsunternehmen, welches durch den intensiven Handymarkt eine hohe Fluktuation besitzt, ein Framework für den Austausch von Kundendaten finden.

Zur Erstellung dieser Access Datenbank wurde die rudimentäre Standards-Datenbank des Institutes für Wirtschaftsinformatik der Universität St. Gallen verwendet. Diese wurde darauf mit den vier Dimensionen des Standards-Modells erweitert. Zusätzlich wurden duzende neu entwickelte XML-Standards erfasst. Somit besitzt die Datenbank zur Zeit mehr als 130 Datensätze und wird in den nächsten Jahren gewiss noch wachsen. In einem späteren Zeitpunkt wird die Datenbank mittels ASP den Partnern sowie den Mitarbeitern des Institutes

zur Verfügung gestellt. Diese sollen damit nicht nur die Gelegenheit einer Suche erhalten, sondern es soll ihnen ermöglicht werden auch neue Datensätze zu erfassen oder bestehende zu aktualisieren. Durch die stetige Weiterentwicklung der XML-Standards ist es von absoluter Notwendigkeit, dass die Datensätze stetig aktualisiert werden.

Im nächsten Kapitel werden zwölf XML-Standards genauer betrachtet und anschliessend in die oben beschriebene Datenbank aufgenommen.

4 Analyse ausgewählter B2B Standards

Dieses Kapitel dient als Überblick über zwölf ausgewählte B2B-Standards. Sowohl international fortgeschrittene Standardisierungsvorhaben wie auch Standards, welche bis jetzt nur für den deutschen Markt entwickelt wurden, werden vorgestellt. Die Auswahl erfolgte über die Aktualität und die Popularität der Standards und wurde durch die Kontaktpersonen von BEA Systems und PWC (s. Anhang A) bestätigt. Es gibt jedoch noch eine grosse Menge an anderen Standards und deren Vorhaben, welche in Zukunft ebenfalls einen Einfluss auf die Business Collaboration Infrastructure haben werden. Die zwölf Standards sind somit nur als eine Auswahl zu betrachten. Am Ende dieses Kapitels werden die zwölf Standards ins Modell der Standards-Datenbank eingeordnet.

4.1. Electronic Business XML

Im November 1999 starteten die beiden Organisationen UN/CEFACT und OASIS ein Projekt zur Standardisierung von XML-Dokumenten für das e-Business. Dies war die Geburt von electronic business XML (ebXML). Das Ziel ist die Schaffung eines offenen, technischen Frameworks, um XML für den zwischenbetrieblichen Austausch von Geschäftsdaten bei System-zu-System, System-zu-Mensch und Mensch-zu-System in konsistenter und einheitlicher Weise zu ermöglichen [vgl. Huemer 2001, S. 26]. Ein weiteres Ziel dieses Projektes ist die Senkung der Eintrittsbarriere für den elektronischen Datenaustausch, insbesondere für KMUs und Entwicklungsländer [vgl. Logan 2000].

4.1.1 Spezifikation von ebXML

Das ebXML-Konsortium hat die Organisation zur Spezifikation des Standards in sieben Projektgruppen unterteilt, welche für die Umsetzung ihrer jeweiligen Aufgabe verantwortlich sind. Dabei wurden sowohl Projektgruppen gegründet, welche sich mit den technischen Aspekten auseinander setzen, als auch Gruppen die sich für administrative Aufgaben aufopfern. Tabelle 4-1 gibt einen Überblick über die verschiedenen Projektgruppen und ihren Aufgaben.

Projektgruppe	Aufgabe / Umsetzung
Business Process	• Generierung des Metamodells • Definition von Geschäftsprozessen • Interaktion mit anderen Modellen
Core Components	• Definition von Struktur und Inhalt der Kernfunktionen • Kernfunktionen in syntaxneutraler Darstellung • Wiederverwendbarkeit / Erweiterbarkeit sicherstellen
Technical Architecture	• Definition über das „Wie“ bei der Umsetzung von Geschäftsprozessen • Darstellung von Geschäftsprozessen in Unabhängigkeit der technischen Lösung • Definition, Implementierung und zur Verfügungsstellung einer Bibliothek mit Standard Geschäftsprozessen • Definition eines verbindlichen Vokabulars und einer eindeutigen Semantik
Transporting/Routing & Packaging	• Definition über die „Verpackung“ und den Transfer von Geschäftsdokumenten • Physische und/oder logische Adressierung • Routing der „Nachrichten“ • Vereinigung zusammengehörender Nachrichten in eine Collection • Sicherheitsaspekte beim Transport
Registry & Repository	• Entwurf einer detaillierten Vorlage für ein Repository • Definition von Schnittstellen mit anderen XML-Geschäftsprozessstandards • Zusammentragen der Modelle/Vorlagen der Prozesseigentümer • Sicherstellung des Zugriffs auf Registry und Repository über Web-Schnittstellen
Technical Coordination & Support	• Metagruppe des Projekts • Koordination der Arbeitsgruppen • Einhaltung der ebXML-Definitionen/Zeiteinhaltung
Marketing, Awareness & Education	• Ergebnisse der ebXML-Organisation unter das Volk bringen • Organisation von Marketingmassnahmen • Planung von Schulungen

Tabelle 4-1: ebXML-Standardisierungsinitiative: Projektgruppen und ihre Aufgaben (in Anlehnung an [Sanmann 2001])

Im folgenden Abschnitt wird nur auf die grundlegende Architektur und die Funktionsweise von ebXML eingegangen. Für spezifische Informationen zu den einzelnen Projektgruppen ist auf die Webseite von ebXML (http://www.ebxml.org/reports) verwiesen.

Die Architektur von ebXML wird in zwei Teile aufgegliedert. Der erste Teil beschreibt die Zeitspanne, bis sich zwei Partner für eine Kollaboration gefunden haben. Der zweite Teil beschäftigt sich mit dem Datenaustausch während der Beziehung. Bild 4-1 stellt die Architektur von ebXML vor.

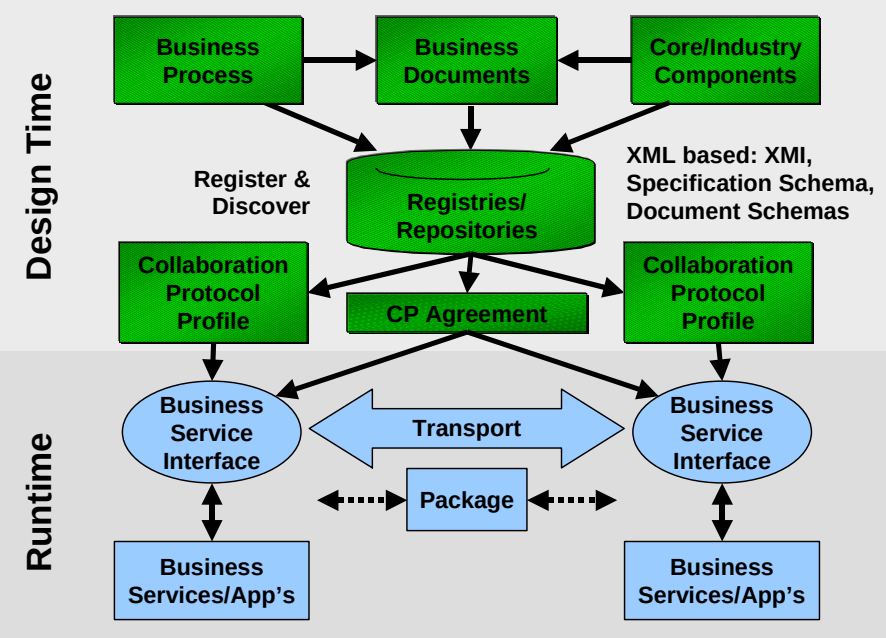


Bild 4-1: Architektur von ebXML [Manes 2001]

Sämtliche Unternehmen, die ebXML unterstützen, sind in einem Repository gespeichert. Dabei handelt es sich nicht nur um Kontaktinformationen, sondern auch um deren grundlegenden Geschäftsprozesse und -dokumente. Finden sich zwei Partner so erstellen diese ein Collaborative Partner Agreement (CPA). Ebenfalls wird bestimmt, welche Prozesse mit welchen Protokollen in Zukunft abgewickelt werden.

Um die Funktionsweise von ebXML zu verdeutlichen, wird nun ein mögliches Szenario in Anlehnung an [Fitzgerald 2001, S. 167ff] beschrieben. Die Firma A ist ein Softwarehersteller und hat soeben eine neue Lösung zur Datenkomprimierung von DVD-Filmen entwickelt und will diese nun auf dem Markt verkaufen. Da Firma A ebXML verwendet, sucht sie nun einen Partner, welcher die Software auf dem Markt verkauft und ebenfalls ebXML unterstützt. Als erstes besucht die Firma A die Webseite ‚Registry and Repository' der ebXML-Community. Diese Webseite enthält Name, Kontaktadressen und Informationen über die Geschäftsprozesse

die mit ebXML unterstütz werden. Diese Institution ist mit UDDI (vgl. Kapitel 4.9) vergleichbar, enthält jedoch nur Informationen über Unternehmen die ebXML verwenden. Firma A entdeckt nun die Unternehmung B, welche sich mit dem Verkauf von Software beschäftigt. A sendet darauf eine Anfrage an B, ob ein Geschäftsinteresse an der neuen Softwarelösung vorhanden sei. Um das Beispiel weiterzuführen, ist diese natürlich daran interessiert.

In einem nächsten Schritt wird zwischen den beiden Unternehmen ein CPA unterzeichnet. Dieses ebXML-CPA-Dokument wurde von IBM übernommen und enthält beispielsweise Informationen über die Handelspartner und ihre Rollen, Transaktionsprotokolle oder auch die Dauer des Abkommens. Haben beide Unternehmen das CPA angenommen, kann nun mit den Geschäftstransaktionen, wie Bestellung, Lagerbestandabfrage oder Zahlungen, begonnen werden. Bild 4-2 gibt einen Einblick in das CPA-Dokument von ebXML.

```
<?xml version = "1.0" encoding = "UTF-8"?>
<!DOCTYPE CollaborationProtocolAgreement SYSTEM
"http://ebxml.org/project_teams/trade_partner/cpp-cpa-v1_0.dtd">
<tp:CollaborationProtocolAgreement
            xmlns:tp="http://www.ebxml.org/namespaces/tradePartner"
            xmlns:xlink="http://www.w3.org/1999/xlink"
            xmlns:ds="http://www.w3.org/2000/09/xmldsig#"
            tp:cpaid="uri:yoursandmycpa"
            tp:version="1.2">
            <tp:Status tp:value="proposed"/>
            <tp:Start>2001-05-20T07:21:00Z</tp:Start>
            <tp:End>2002-05-20T07:21:00Z</tp:End>
            <tp:ConversationConstraints tp:invocationLimit="100/>
            <tp:PartyInfo>
                        <tp:PartyId tp:type="DUNS">123456789</tp:PartyId>
                        <tp:PartyRef xlink:href="http://example.com/about.html"/>
                        <tp:CollaborationRole tp:id="N00">
                            ⋮
                        </tp:CollaborationRole>
                        <tp:Certificate tp:certId="N03"><ds:KeyInfo/>
                        </tp:Certificate>
                        <tp:DeliveryChannel>
                            ⋮
                        </tp:DeliveryChannel>
                        <tp:Transport tp:transportId="N05">
                            ⋮
                        </tp:Transport>
                        <tp:DocExchange tp:docExchangeId="N06">
                                    <tp:ebXMLBinding tp:version="0.98b">
                                        ⋮
                                    </tp:ebXMLBinding>
                        </tp:DocExchange>
            </tp:PartyInfo>
            <tp:PartyInfo>
                ⋮
            </tp:PartyInfo>
            <tp:Packaging tp:id="N0402">
                ⋮
            </tp:Packaging>
            <tp:Comment xml:lang="en-us">buy/sell agreement ...</tp:Comment>
</tp:CollaborationProtocolAgreement>
```

Bild 4-2: CPA-Dokument bei ebXML[12]

Jedes ebXML-Dokument enthält einen ebXML-Header mit drei Attributen. Zuerst wird definiert um welchen XML-Standard es sich handelt, dann folgt die Version des Standards und zum Schluss wird bestimmt um welche Art von Dokument es sich handelt. Dies kann Normal, Error oder Acknowledgment sein. Ein ‚Error' wird verwendet, wenn zum Beispiel die Informationen nicht aus dem Dokument geparst werden konnte. Ist eine Bestellung erfolgreich verbucht worden, wird ein ‚Acknowledgment' an den Besteller zurück gesandt. Handelt es sich um eine Bestandesabfrage oder eine Bestellung wird ‚Normal' verwendet. Nach dem ebXML-Header folgt das Element ‚Manifest'. Hier bekommt jedes Dokument ein Label und eine ID. Ein Label ist eine Kurzbeschreibung der Dokumentart und die ID ist eine eineindeutige Nummer.

Nun folgt der Hauptteil der Message, welcher wiederum einen Header besitzt. Dort werden die beiden Partner sowie die Identifikationsnummer des CPAs angegeben. Weitere Elemente sind Zeitangaben und Routinginformationen. Zuletzt folgt die eigentliche Aktion. So werden bei einer Bestellung beispielsweise die ‚OrderDetails' angegeben. Wie eine Bestellung aber genau aussieht, ist bis jetzt in der Spezifikation der Technical Group noch nicht definiert worden. Es stehen jedoch sowohl ein Dokument eines möglichen Geschäftsprozesses als auch das CPA-Dokument auf der Webseite von ebXML zur Verfügung. Die Spezifikation von ebXML ist zwar schon weit fortgeschritten, jedoch befindet es sich immer noch im Entwicklungsprozess.

4.1.2 Zukunftsaussichten für ebXML

Durch die Zusammenarbeit von UN/CEFACT und OASIS stehen hinter der Entwicklung von ebXML zwei namhafte Organisationen im Bereich des e-Business. Vorgesehen war, dass das Projekt bis im Mai 2001 abgeschlossen ist. Jedoch befinden sich einige Komponenten von ebXML immer noch in der Spezifikation. Durch den enormen Umfang hat sich die Entwicklung von ebXML vorwiegend auf die Prozessarchitektur konzentriert und die Entwicklung von Geschäftsdokumenten wurde sehr gering gehalten. Nach Schätzungen der Gartner Group wird sich ebXML bis Ende 2002 als global führender XML-Standard im EC etablieren [vgl. Abrams 2001b]. Die Stärke von ebXML liegt vor allem darin, dass nicht nur eine Sprache, sondern auch ein Repository und ein Framework angeboten wird. Zudem können mit ebXML

[12] Originalbeispiel von ebXML unter www.ebxml.org/reports

nicht nur die eigenen Dokumente ausgetauscht sondern auch die parallel entwickelten XML-Standards in ebXML integriert werden [vgl. Abrams 2001a].

4.2. RosettaNet

RosettaNet wurde im Februar 1998 in den USA gegründet und ist ein nicht auf Gewinn ausgerichtetes Konsortium von mehr als 400 führenden IT-Unternehmungen.

4.2.1 Standardisierungsinitiative von RosettaNet

Der Geschäftsverkehr bei der Mensch-zu-Mensch Kommunikation ist sehr erfolgreich und effizient, da die Geschäftspartner Prozesse auf der elementarsten Ebene eingehen. Menschen erzeugen und hören Laute, verwenden ein gemeinsames Alphabet um daraus Wörter zu generieren, gebrauchen grammatische Regeln um die Wörter in einen Dialog zu stellen, verwenden Dialoge um Geschäftsprozesse zu formen, und führen schliesslich Verhandlungen durch ein Instrument wie beispielsweise das Telefon.

Eine Analogie herrscht im Geschäftsverkehr des Informationszeitalters vor. Anstelle von Klängen der Menschen werden nun Informationen von Servern über das Internet ausgetauscht. Die Sprachen HTML oder XML haben die Funktion des Alphabets und EC-Applikationen dienen als Instrumente zur Übermittlung von e-Business-Prozessen. Dazwischen besteht jedoch noch ein Mangel an Vereinbarungen, welche die Wörter, die Grammatik und den Dialog regeln, um somit den e-Business Prozess zu unterstützen. Genau an dieser Stelle setzt RosettaNet an [vgl. RosettaNet 2001a, S. 6ff].

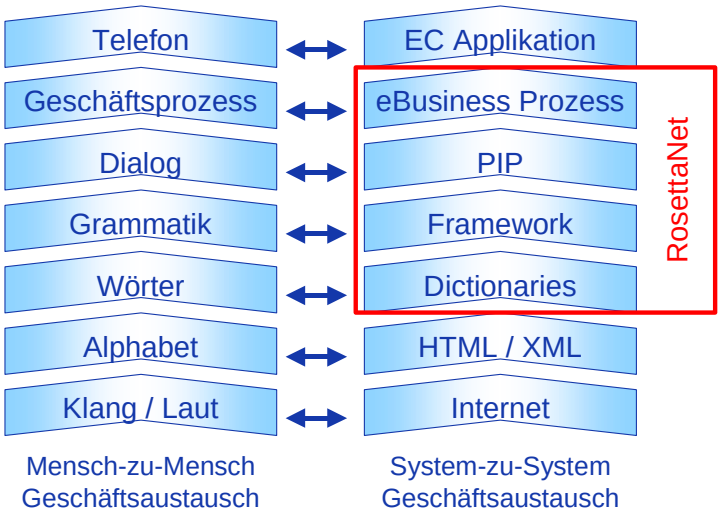


Bild 4-3: Einsatzgebiete von RosettaNet [Huemer 2000a]

RosettaNet stellt zwei Dictionaries zur Verfügung – RosettaNet Business Dictionary und RosettaNet Technical Dictionary –, welche als allgemeine Plattform für Definitionen von Transaktionen zwischen Handelspartnern und von Produkten und Services dienen. Das RosettaNet Implementation Framework (RNIF) stellt Protokolle für eine schnelle und effiziente Implementierung von RosettaNet Standards zur Verfügung. Damit werden die Informationen über den Datenaustausch spezifiziert, wie beispielsweise das Routing, die Paketgrösse oder die Sicherheitsvorkehrungen.

Der Kern dieses Modells sind die sogenannten Partner Interface Processes (PIPs). PIPs sind XML-basierte Dialoge für die Anwendung-zu-Anwendung Kommunikation, welche die Geschäftsprozesse zwischen Handelspartnern definieren [vgl. RosettaNet 2001a]. Weitere Informationen finden sich auf der Homepage von RosettaNet (www.rosettanet.org).

4.2.2 Zukunftschancen von RosettaNet

Der von RosettaNet präsentierte Bezugsrahmen deckt die wesentlichen Aspekte ab, die bei der Einführung einer Business Collaboration Infrastructure zu beachten sind. Es werden Empfehlungen abgegeben, wie man bei der Analyse und der Reorganisation der betroffenen Geschäftsprozesse vorzugehen hat. Die vorgelegten Konzepte werden zwar kompetent diskutiert, wirken allerdings fragmentarisch und nicht ausgereift [vgl. Frank 2001, S. 289f].

Auf der anderen Seite handelt es sich laut Microsoft bei RosettaNet um einen weithin akzeptierten Standard für branchenübergreifende e-Business-Prozesse. Nicht nur verschiedene Anwendungen, Systeme und Plattformen können schnell in die Firmenarchitektur eingebunden werden, sondern auch interne wie externe Geschäftsabläufe und Transaktionen können ohne Aufwand automatisiert werden. Aus diesem Grund hat Microsoft den BizTalk Server Accelerator for RosettaNet entwickelt, der ab Sommer 2001 verfügbar ist [vgl. Donath 2001].

Gemäss RosettaNet ist die gegenwärtige Verwendung ihrer Standards auf ungefähr dreissig grosse Unternehmungen mit jeweils weniger als zehn Handelspartnern limitiert [vgl. Rishel 2001]. Auf der einen Seite befindet sich RosettaNet dadurch noch in den Kinderschuhen, aber auf der anderen Seite gibt es keine andere nicht-proprietäre Lösung von vergleichbaren Supply Chain Standards [vgl. Rishel 2001].

4.3. Commerce eXtensible Markup Language

Im Februar 1999 lancierte Ariba den XML-Standard commerce eXtensible Markup Language (cXML). Um einen möglichst offenen Zugang zu den elektronischen Marktplätzen des B2B-Bereichs zu schaffen, veröffentlichte Ariba gleichzeitig die Spezifikationen von Schnittstellen, die von den eigenen Produkten unterstützt werden [vgl. Frank 2000, S. 18]. cXML kommt vor allem dort zur Anwendung, wo man sich vom traditionellen Geschäftsverkehr mit all seinen Papierdokumenten trennen will [vgl. Ariba 2000, S. 1].

4.3.1 Spezifikation von cXML

Transaktionen werden über cXML-Dokumente abgewickelt, bei denen es sich um einfache Textdateien mit klar definierten Formaten und Inhalten handelt. Dabei werden drei Haupttypen von cXML-Dokumenten unterschieden: Kataloge, Panchoutsitzungen und Bestellaufträge [vgl. Ariba 2000, S. 2ff].

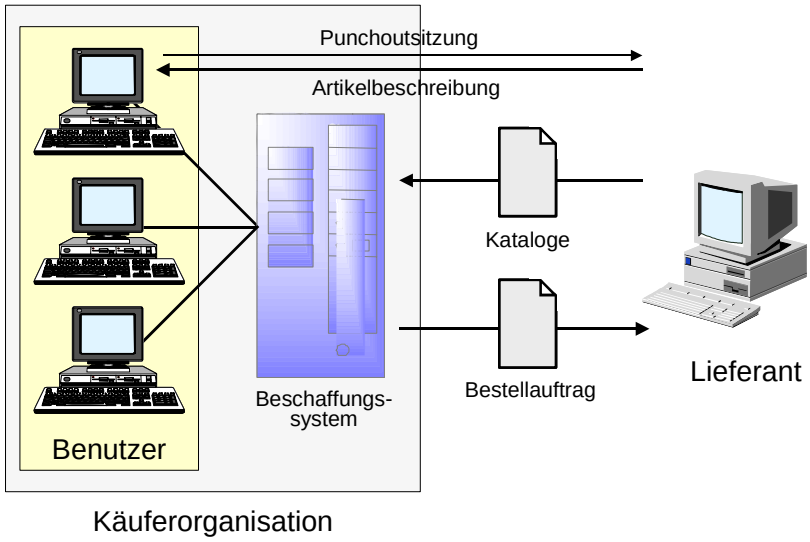


Bild 4-4: Interaktivität zwischen Lieferant und Käufer mit cXML (in Anlehnung an [Ariba 2000, S. 2ff]

Kataloge sind Dateien, in denen Produkt- und Dienstleistungsinhalte an Käuferunternehmen herangetragen werden. Nebst den Produkten und Leistungen sind auch Informationen zu den jeweiligen Preisen in den Katalogen enthalten. Die vom Lieferanten erstellten Kataloge werden vom Käufer gelesen und deren Inhalt in eine interne Datenbank gespeichert. Nachdem das Käuferunternehmen die Kataloge autorisiert hat, wird deren Inhalt für den Benutzer zugänglich. Dieser kann nun Artikel auswählen und Kaufanforderung hinzufügen. Kataloge

lassen sich grundsätzlich für alle Produkte und Dienstleistungen erstellen, weiter können auch optionale Informationen, wie beispielsweise eine Katalogfunktion, hinzugefügt werden.

Bisher wurden nur statische Katalogdateien betrachtet. Punchouts hingegen sind dynamische, interaktive Kataloge, die sich auf der Webseite des Lieferanten befinden. Punchoutkataloge sind Webserveranwendungen, die in einer für Punchoutsitzungen geeigneten Programmiersprache wie ASP oder JavaScript geschrieben sind. Der Benutzer meldet sich zuerst bei einer Beschaffungsanwendung an und durchstöbert dort den lokalen, statischen Katalog. Wählt er dabei einen Punchoutartikel, öffnet sich ein neues Browserfenster und es erfolgt eine Anmeldung auf der Webseite des Lieferanten. Dort kann der Benutzer nun seine gewünschten Artikel auswählen und weitere Funktionen und Leistungen nutzen. Solche Funktionen können beispielsweise eine Suchmaschine, Informationen über Ersatzteile oder ein Konfigurationswerkzeug sein. Hat der Benutzer die Produktauswahl beendet, berechnet die Webseite des Lieferanten die Gesamtkosten – einschliesslich Steuern, Frachtkosten und kundenspezifischer Rabatte – und sendet dann eine ‚PunchOutOrderMessage' im cXML-Format an die Beschaffungsanwendung. Diese kontrolliert dann die Autorisierung des Benutzers und löst darauf einen Bestellauftrag aus und sendet diesen zur Erledigung an die Webseite des Lieferanten zurück [vgl. Ariba 2000, S. 15ff]. Die Bestellung ist somit ausgelöst.

Bestellaufträge können sowohl nach einer Auftragserfassung in einem statischen Katalog als auch nach einer Punchoutsitzung ausgelöst werden. Dabei wird der Bestellauftrag, ein ‚OrderRequest'-Dokument im cXML-Format, an die URL-Adresse des Lieferanten gesendet und dort intern an das zentrale Auftragsverwaltungssystem weitergeleitet. Ist der semantische Inhalt des Dokumentes vom System analysiert worden, sendet dieses ein ‚OrderResponse'-Dokument an den Käufer zurück und beendet somit den Bestellvorgang. Beispiele der oben genannten Dokumente sind in [Ariba 2000] zu finden. Häufig ist es erforderlich, dass ein Bestellauftrag durch den Käufer mit Memos, Zeichnungen oder Faxnachrichten näher erläutert werden muss. Diese Zusatzinformationen können durch eine MIME-Codierung[13] an cXML-Bestellaufträge angehängt werden [vgl. Ariba 2000, S. 41ff].

[13] Für weitere Informationen zum MIME-Standard und dessen Codierung sei auf die folgenden Webweiten verwiesen: http://www.hunnysoft.com/mime; http://www.rad.com/networks/1995/mime/mime.htm

4.3.2 Beurteilung und Zukunftsaussichten für cXML

Nach Angaben von Ariba waren 52 Anwenderunternehmen bei der Entwicklung von cXML beteiligt. Dies lässt daraus schliessen, dass wichtige Anforderungen berücksichtigt wurden [vgl. Frank 2001, S. 285]. Jedoch ist der cXML-Standard noch unvollständig. So fehlen beispielsweise Spezifikationen zu besonderen Lieferbedingungen, wie etwa die Vereinbarung einer Gebühr bei der Rückgabe eines erworbenen Gutes innerhalb eines bestimmten Zeitraums. Auch ist nicht erkennbar, wie die Konfiguration eines Produktes abzubilden ist [vgl. Frank 2000, S. 22]. Ariba ist jedoch intensiv daran, ihren Standard zu etablieren. So laufen zur Zeit Verhandlungen mit der General Services Administration der amerikanischen Regierung. Sollte sich cXML als Standard für dessen Auktionen durchsetzen, wird dies ein grosser Einfluss auf alle Organe der Regierung der USA haben [vgl. EDS 2000, S. 58].

4.4. Common Business Library

Die auf XML- basierende Common Business Library (xCBL) wurde 1997 von Commerce One in Zusammenarbeit mit Veo Systems, ein Pionier in der XML-Entwicklung, ins Leben gerufen. Die Weiterentwicklung von xCBL liegt heute in den Händen von Commerce One. xCBL wird oft mit dem Standardisierungsvorhaben cXML von Ariba verglichen, jedoch ist die Entwicklung von xCBL fortgeschrittener und nach Angaben von Fachexperten steckt mehr Ehrgeiz hinter xCBL [vgl. Abrams 2001b]. Die Verwendung von xCBL ist nicht nur auf die Umgebungen von öffentlichen e-Marktplätzen beschränkt, sondern etabliert sich immer mehr in privaten e-Marktplätzen. Durch die fast sechshundert definierten Elemente mit variierendem Komplexitätsgrad, wird eine Individualisierung stark vereinfacht [vgl. Fitzgerald 2001, S. 201f].

4.4.1 Spezifikation von xCBL

Das xCBL Vocabulary macht es für jeden Anwender möglich, selbst eine Anzahl von XML-Dokumenten zu verfassen und somit an ein spezielles Bedürfnis anzupassen. Bild 4-5 gibt einen allgemeinen Überblick über die verschieden Kategorien, bei denen xCBL-Dokumente zur Anwendung kommen.

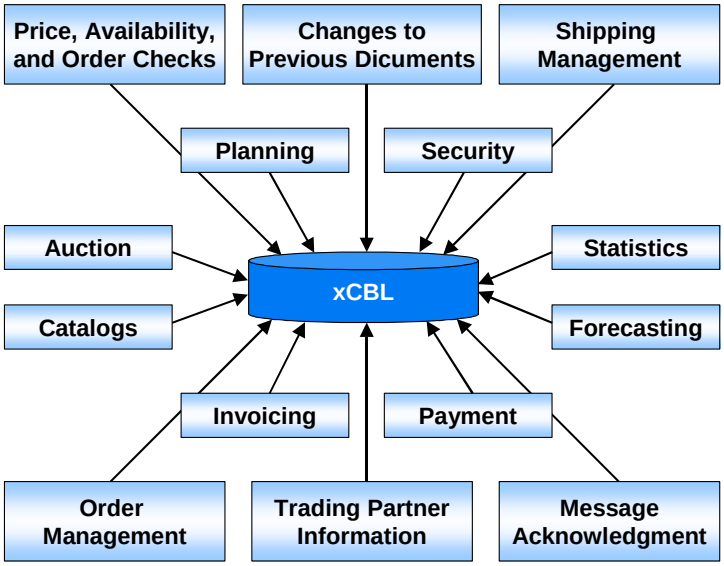


Bild 4-5: Unterkategorien der Common Business Library (in Anlehnung an [Fitzgerald 2001, S. 186])

In der folgenden Spezifizierung wird nur eine einfache Bestellung, eine elementare Rechnung und der Aufbau einer Auktion beschrieben [vgl. Fitzgerald 2001, S. 187ff]. Für weitere Informationen zu den obigen Kategorien ist auf das über 1'500seitige Whitepaper von Commerce One verwiesen [CommerceOne 2000].

Entscheidet sich eine Unternehmung für ein Bestellwesen mit xCBL, ergeben sich mehrere Optionen für den Bestellprozess. Zuerst kann der Käufer eine Bestellanfrage beim Lieferant aufgeben. Dieser antwortet darauf mit einem ‚OrderRequest', was einer Offerte entspricht. Ist der Käufer mit dem Preis und den Lieferbedingungen zufrieden, sendet er eine ‚Order' an den Lieferanten zurück. Hat dieser die Bestellung erhalten und zur Verarbeitung weitergeleitet, sendet der Lieferant einen ‚OrderResponse' als Bestätigung an den Käufer zurück. Zusätzlich kann der Käufer durch einen ‚ChangeOrder' Änderungen an der Bestellung vornehmen. Das einzige Muss-Element bei einem Bestellprozess mit xCBL ist die ‚Order'. Alle drei anderen Optionen – ‚OrderRequest', ‚OrderResponse' und ‚ChangeOrder' – sind von den Partnern frei wählbar. Folgende ausführlichen Informationen zu einer Bestellung mit xCBL beziehen sich deshalb nur auf das Muss-Element ‚Order'.

Jede Bestellung mit xCBL besitzt einen ‚OrderHeader', ein ‚OrderDetail' und ein ‚Order-Summary'. Im Header sind die Identifikationsnummern des Käufers und des Lieferanten, sowie diejenige der Bestellung enthalten. Weitere Attribute die im Header definiert werden, sind das Erstellungsdatum, die Währung, die Kommunikationssprache, die Adresse der beiden Partner, sowie die Zeitdifferenz vom Käufer zum Lieferant. Zusätzlich wird in einem

‚PurposeCode' der Ziel und Zweck des Dokuments spezifiziert. Mögliche Codes sind: ‚Original', ‚Cancellation', ‚Replace', ‚Confirmation', ‚Duplicate', ‚Change', ‚InformationOnly' oder ‚Other'. Im zweiten Teil werden die gewünschten Artikel, die Menge und weitere Attribute, die aus Bild 4-6 zu entnehmen sind, definiert.

```
<OrderDetail>
     <BaseItemDetail>
          <LineItemNum>1</LineItemNum>
          <SupplierPartNum>
               <PartNum>
                    <Agency AgencyID="AssignedBySupplier"/>
                    <PartID>12345</PartID>
               </PartNum>
          </SupplierPartNum>
          <ItemDescription>100 MB Zip Diskette</ItemDescription>
          <Quantity>
          <Qty>000000000001.000</Qty>
               <UnitOfMeasure><UOMCode>EA</UOMCode></UnitOfMeasure>
          </Quantity>
          <Transport Direction="SupplierToBuyer">
               <Mode>Air</Mode>
               <Mean>Express</Mean>
               <Carrier>Fedex</Carrier>
               <CustShippingContractNum>CTOP123</CustShippingContractNum>
               <ShippingInstruction>Please handle with care</ShippingInstruction>
          </Transport>
          <OffCatalogFlag>false</OffCatalogFlag>
     </BaseItemDetail>
     <BuyerExpectedUnitPrice>
          <Price><UnitPrice>00000000010.0000</UnitPrice></Price>
     </BuyerExpectedUnitPrice>
</OrderDetail>
```

Bild 4-6: OrderDetail einer Bestellung mit xCBL [CommerceOne 2000]

Im letzten Teil, dem ‚OrderSummary', werden die Preise der bestellten Artikel, sowie die Steuern zusammengerechnet und die Währung nochmals definiert. Zusätzlich können noch Angaben über den Steuersatz gemacht werden.

Analog zur Bestellung besteht auch die Rechnung aus drei Hauptteilen – dem ‚InvoiceHeader', dem ‚InvoiceDetail' und dem ‚InvoiceSummary'. Jede Rechnung erhält eine Referenznummer, welche im Header bestimmt wird. Weiter werden darin identische Attribute, wie das Ausstelldatum, die Kommunikationssprache, die Identifikationsnummern der Partner und deren Adresse, definiert. Der ‚PurposeCode' bei einer Rechnung kann hingegen nur die Werte ‚Original', ‚Duplicate' oder ‚Other' annehmen. Wird der Wert ‚Other' verwendet, kann mit einem zusätzlichen Tag die Bedeutung spezifiziert werden. Mögliche Werte dafür sind ‚CreditInvoice', ‚CorrectedInvoice' oder ‚DebitInvoice'. Weiter werden im ‚InvoiceHeader' Informationen über die Fertigstellung der Lieferung aufgeführt. Im zweiten Teil, dem ‚InvoiceDetail' wird der bestellte Artikel angegeben, sowie die Menge, die Einheit und der Preis pro Artikel. Aus obiger Bestellung sind dies somit die Artikelnummer 12345 für

die eine, bestellte Zip-Diskette, mit einem Preis von zehn in der Währung USD. Jeder Artikel bekommt dabei einen ‚PaymentStatus', welcher die Zahlungsart beschreibt. Normalerweise wird der Wert ‚PaidInFull' gesetzt. Je nach Situation kommen andere, selbsterklärende Werte zur Anwendung. Dies sind beispielsweise ‚PartPaid', ‚NotPaid', ‚OverPaid' oder ‚FreeOf-Charge'. Im letzten Teil dem ‚InvoiceSummary' werden schliesslich die Preise der bestellten Artikel zusammen gerechnet und mögliche Steuern daraufgeschlagen. Wie bei den einzelnen Artikeln kann auch analog die ganze Rechnung einen ‚InvoicePaymentStatus' zugewiesen bekommen.

Eine Auktion ist eine weitere Art um Güter zu verkaufen, dies jedoch zu einem variablen Preis. Hat beispielsweise ein Lieferant im Januar noch ein Stapel an Weihnachtsartikel, die im nächstes Jahr nicht mehr aktuell sein werden, kann er eine Auktion starten, anstelle diese abzuschreiben. Um solche Auktionen durchführen zu können, hat Commerce One ein xCBL Auktionsmodell entwickelt. Dabei errichtet der Initiator durch ein ‚AuctionCreate'-Dokument eine Auktion. Dieses Dokument wird an einen Server gesendet, der einen Auktionsservice anbietet. Idealerweise ist dies der Marktplatz www.marketsite.com von Commerce One, aber auch andere Marktplätze, welche eine Auktionsserviceplattform anbieten, sind möglich. Wurde die Auktion erfolgreich eingerichtet, sendet der Auktionsserver ein ‚AuctionCreate-Response'-Dokument als Bestätigung an den Initiator zurück. Während der Auktion kann der potentielle Käufer durch e-Mails sein Preisangebot kundgeben. Hat die Versteigerung ihr Verfallsdatum erreicht, sendet der Auktionsserver ein ‚AuctionResult'-Dokument mit sämtlichen Angaben des Meistbietenden an den Urheber. Zur Bestätigung sendet dieser darauf eine ‚AuctionResultResponse'-Dokument an den Server zurück. Im folgenden wird nur auf das ‚AuctionCreate'-Dokument eingegangen. Für weitere Informationen zu den anderen Dokumenten ist auf die Homepage von xCBL oder Commerce One[14] verwiesen.

Ein ‚AuctionCreate'-Dokument besteht wie die meisten xCBL-Dokumente aus einem Header, einem Detail und einem Summary. Der ‚AuctionCreateHeader' enthält Informationen über das Erstellungsdatum, das Start- und Enddatum der Auktion, sowie die Adresse des Initiators und die Korrespondenzsprache. Weiter muss ein ‚PurposeCode' identisch zur Bestellung angegeben werden. In einem zusätzlichen Tag muss der Initiator festlegen, ob es sich um eine angebots- oder nachfragebasierte Auktion handelt. Bei einer angebotsbasierten Auktion verkauft der Lieferant seine Artikel an den Meistbietenden, dabei wird das Attribut auf ‚true'

[14] www.xcbl.org oder www.commerceone.com/solutions/business/auction.html

gesetzt. Sucht ein Käufer den günstigsten Anbieter für ein Produkt, handelt es sich um eine nachfragebasierte Auktion. Hier wird das Attribut auf ‚false' gesetzt. Im mittleren Teil, dem ‚AuctionCreateDetail', wird der Gegenstand präzisiert, der versteigert werden soll. Ist dies beispielsweise ein Reiseführer, so wird die Kategorie Buch gewählt und als Identifikationsnummer die ISBN-Nummer[15] zur eindeutigen Identifizierung angegeben. Je nach Portal des Auktionsanbieter muss eine bestimmte Hierarchie eingehalten werden. Falls eine solche Hierarchie notwendig ist, wird diese ebenfalls im zweiten Teil spezifiziert. Schliesslich muss noch die Anzahl der zu versteigernden Artikel bekannt gegeben werden. Das ‚AuctionCreate-Summary' ist der optionale letzte Teil. Hier wird lediglich zusammengefasst, wie viele unterschiedliche Artikel gemeinsam versteigert werden.

4.4.2 Zukunftsaussichten für xCBL

Das Standardisierungsvorhaben von Commerce One ist gegenüber anderen stark fortgeschritten [vgl. Frank 2000, S. 28] und dokumentiert fast sechshundert Elemente in der xCBL-Bibliothek [vgl. Fitzgerald 2001, S. 202]. Die Weiterentwicklung dieser Bibliothek wird durch organisatorische Massnahmen unterstützt, was auf eine rasche Aktualisierung der Datenbanken schliessen lässt. Um die Jahrtausendwende gab es ein Boom an Softwareunternehmen, welche alle Lösungen für private wie öffentliche e-Marktplätze anboten. Doch nur wenige von Ihnen haben die Entwicklung weitergeführt und sich auf dem Markt etabliert. Commerce One hat mit xCBL eine führende Position in der Entwicklung von Infrastrukturen für private e-Marktplätze eingenommen [vgl. Cecere et al. 2001].

Ein Nachteil an den xCBL-Dokumenten ist, dass sie von gängigen XML-Parsern nur eingeschränkt, nämlich syntaktisch, validiert werden können. Um dies zu umgehen, wird eine dedizierte Software benötigt und zur Zeit ist es nicht absehbar, in welchem Umfang sich ein Markt für solche Systeme, beziehungsweise Komponenten, bilden wird [vgl. Frank 2001, S. 286].

4.5. Simple Object Access Protocol

Um die Kommunikation zwischen Applikationen anzutreiben, starteten 1999 Microsoft, UserLand Software und DevelopMonitor die Entwicklung des Simple Object Access Protocol

[15] Die internationale Standard-Buchnummer (ISBN) wird jedem herausgebrachtem Buch durch den Verlag zugeordnet. Diese international eindeutige ID wird bereits seit 1969 eingesetzt [vgl. Encarta 2000a].

(SOAP), welches einen Prozessaufruf von einem entfernet System erlaubt. Die erste Version von SOAP operierte nur über HTTP, was im letzten Jahr zu einem Update zur Version 1.1 führte. SOAP ist dadurch mit den meisten Internet Message Systems, wie beispielsweise SMTP oder FTP, einsetzbar [vgl. Hess 2001]. SOAP ist eine Metasprache, die es erlaubt standardisierte XML-Dokumente anzuhängen und deshalb beschäftigen sich Organisationen der XML-Frameworks immer intensiver mit SOAP. BizTalk (s. Kapitel 4.6) verwendet bereits SOAP und auch ebXML wird in näherer Zukunft über eine Unterstützung entscheiden [vgl. Weitzel et al. 2001, S. 120]. RosettaNet hingegen haben sich noch nicht darüber geäussert[16].

4.5.1 Spezifikation von SOAP

SOAP kann nicht nur Dokumente zwischen Server austauschen, wie bis anhin gesehen, sondern auch der Austausch spezieller Nachrichten, welche auf einem entfernten Server einen Prozess auslösen, wird unterstützt. SOAP basiert auf dem Konzept von Remote Procedure Call (RPC). Die grundlegende Idee von RPC ist, dass ein Programm auf einem Computer, welcher mit einem Netzwerk verbunden ist, ein Prozessaufruf bei einer Anwendung auf einem anderen Computer auslösen kann. Der aufgerufene Computer, welcher im Büro nebenan oder auch auf einem anderen Kontinent stationiert ist, kann den Aufruf annehmen und an ein lokales Computerprogramm weiterleiten. Dieses gibt dann einen Wert oder eine sonstige Antwort aus und die Informationen werden an den ursprünglichen Computer zurückgesendet.

Um die Eigenschaften von SOAP aufzuzeigen, wird ein einfaches, fiktives Beispiel einer Börsenabfrage aufgezeigt. Bild 4-7 zeigt den Aufbau einer SOAP-Nachricht ohne den vorausgehenden HTTP-Header[17].

[16] vgl www.rosettanet.org

[17] Für Informationen über den HTTP-Header ist auf [W3C 2000],[W3C 2001] und [Fitzgerald 2001] verwiesen.

```
<SOAP-ENV:Envelope
 xmlns:Soap="http://schemas.xmlsoap.org/soap/envelope">
     <SOAP-ENV:Header>
          <program:Info SOAP-ENV:mustUnderstand="1"
           xmlns:program="urn:eddys-stockomatic:program.xsd">
               <program:Name>Stockomatic</program:Name>
               <program:Version>0.83</program:Version>
          </program:Info>
     </SOAP-ENV:Header>
     <SOAP-ENV:Body>
          <stocks:GetQuote
           xmlns:stocks="urn:eddys-stockmatics:stocks.xsd">
               <stocks:Symbol>GTW</stocks:Symbol>
               <stocks:Time>2001-09-05T16:00:00.000-05:00</stocks:Time>
          </stocks:GetQuote>
     </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

Bild 4-7: Beispiel eines Anfrage-Dokuments mit SOAP [Fitzgerald 2001]

Jedes SOAP-Dokument beginnt mit einem ‚Envelope', das als eine Art Behälter, in welchem die SOAP-Elemente enthalten sind, fungiert. Das ‚Envelope' enthält die Deklaration eines XML-Namespaces, welcher üblicherweise mit dem Präfix SOAP-ENV beginnt. Es ist anzuraten Präfixe zu verwenden, da SOAP-Dokumente meist mehr als ein Namespace beinhalten. Eine Zurückverfolgung wird somit stark vereinfacht. Das zweite Element, der ‚EnvelopeHeader', ist optional einsetzbar, muss jedoch bei seiner Verwendung immer direkt nach dem ‚Envelope'-Element stehen. Die Unterelemente eines Headers sind nicht vorgegeben, werden jedoch solche verwendet, muss deren Namespace angegeben werden. Zusätzlich muss das ‚mustUnderstand'-Element auf eins oder null gesetzt werden. Eine eins bedeutet, dass der Server des Partners die Informationen im Header verstehen muss. Ist dies nicht der Fall, muss der Server eine Error-Meldung an den Initiator zurücksenden. Wird das ‚mustUnderstand'-Element auf null gesetzt, ist es dem Sender egal, ob der Server des Partners die Informationen im Header versteht oder nicht, es folgt also keine Error-Meldung bei Unverständnis. In obigem Beispiel wird somit im Namespace die Stelle angegeben, wo das Computerprogramm installiert ist. Das letzte Element ist der ‚EnvelopeBody', welcher Informationen über die gewünschte Datenabfrage enthaltet. Im Beispiel ist dies die Kursabfrage eines Wertpapiers mit der Bezeichnung GTW zu einem bestimmten Zeitpunkt.

Für Furore sorgte SOAP dadurch, dass das zurückgesendete Antwort-Dokument die angefragte Information enthält, und nicht wie beispielsweise bei xCBL ein einfacher ‚OrderResponse' als Bestellbestätigung zurückkommt. Bild 4-8 zeigt das Antwort-Dokument, welches bei einer Kursabfrage eines Wertpapiers an den Initiator zurückgesendet wird.

```
<SOAP-ENV:Envelope
 xmlns:Soap="http://schemas.xmlsoap.org/soap/envelope">
     <SOAP-ENV:Body>
          <stocks:GetQuote
           xmlns:stocks="urn:eddys-stockmatics:stocks.xsd">
               <stocks:Price stocks:currency="USD">63.97</stocks:Price>
          </stocks:GetQuote>
     </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

Bild 4-8: Beispiel eines Antwort-Dokuments mit SOAP [Fitzgerald 2001, S. 242]

Sowohl der Namespace wie auch dessen Präfix sind identisch zum Anfrage-Dokument. Analog ist auch die abgefragte Information im ‚EnvelopeBody' des Dokuments enthalten. Der ausgegebene Wert ist somit 63.97 und die Währung, welche mit drei Buchstaben bestimmt wird, ist mit einem Attribut im Namespace spezifiziert. Im Beispiel sind dies USD (US Dollars), aber auch weitere Abkürzungen wie EUR (Euros), GBP (Britische Pfund) oder CHF (Schweizer Franken) sind mögliche Werte für die Währung.

SOAP beinhaltet auch ein optionales ‚Fault'-Element, welches über Fehler und Statusinformationen berichtet. Dieses Element erscheint in einem Antwort-Dokument, wenn das vorausgegangene Anfrage-Dokument einen Fehler enthielt. In vier Unterelementen wird beschrieben, wo die Ursache des Fehlers zu finden ist. Das erste davon beschreibt die Stelle, wo der Fehler aufgetreten ist. Dabei kann einer von vier Werten angenommen werden: ‚Server', ‚Client', ‚VersionMismatch' oder ‚MustUnderstand'. Während ‚Client' und ‚Server' den Missetäter ziemlich eindeutig erklären, handelt es sich bei ‚VersionMismatch' um einen Fehler im Namespace und bei ‚MustUnderstand' um ein nicht verstandenes Element im Header. Das zweite Unterelement gibt Informationen, welche für den Menschen einfach zu verstehen sind, wie beispielsweise ‚Server overload – Try again later'. Steigt zum Beispiel das Netzwerk oder ein Verteilserver aus, wird dies im dritten Unterelement kund gegeben. Schliesslich meldet das letzte Unterelement Fehler, welche bei der Rechtschreibung im Body aufgetreten sind.

4.5.2 Beurteilung und Zukunftsaussichten für SOAP

Bei der Verwendung von SOAP gibt es weder Einschränkungen von Seite der Programmiersprachen und Internetprotokollen noch von Seite der Chipsätze und Systeme. Verwenden zwei Partner SOAP für den zwischenbetrieblichen Datenaustausch, müssen ihre internen Systeme nicht mit einander kompatibel sein und sie müssen auch nicht dem Web ausgesetzt werden. Einzige Anforderungen bei Verwendung von SOAP sind eine geringe Anzahl an Ressourcen,

welche den Prozess weiterverarbeiten. So müssen beide Partner zum Beispiel in der Lage sein, Datenpakete übers Internet zu erhalten und zu versenden, Zugriff auf einen XML-Parser haben und fähig sein, Informationen von XML in interne Applikationen zu übersetzen und umgekehrt [vgl. Knox/Hess 2001].

Nach Annahmen der Gartner Group wird SOAP in Zukunft eine Hauptrolle als Protokoll für Webservicezugriffe einnehmen. Im Jahr 2004 wird die zu diesem Zeitpunkt aktuelle Version von SOAP eines von zwei oder drei der weitverbreitetsten Protokolle für den Zugriff auf Webservices sein. Mehr als 30 Prozent des Datenaustausches zweier Anwendungen über das öffentliche Internet werden mittels SOAP erfolgen, und dies mit einer Wahrscheinlichkeit von 70 Prozent [vgl. Natis/Driver 2001]. Doch SOAP weist auch Nachteile auf, welche Einfluss auf dessen Verbreitung haben. So ist SOAP beispielsweise langsamer als andere Nachrichtenformate bei der Kommunikation zwischen zwei Applikationen. Dies kann bei grossen Systemen zu unerwünschten Performanceproblemen führen. Zudem werden kompliziertere RPCs oder auch eine synchrone, bidirektionale Kommunikation nicht unterstützt. Weiter sind in SOAP keine Sicherheitselemente eingebaut, obwohl es dafür bestimmt wurde, populäre Sicherheitstechnologien, wie beispielsweise Kerboros[18], zu unterstützen [vgl. Knox/Hess 2001].

4.6. BizTalk

BizTalk wurde 1999 von Microsoft ins Leben gerufen und besteht aus drei Komponenten [vgl. Huemer 2000]. Die erste Komponente ist das BizTalk Framework, das als Gerüst zur Verwendung von BizTalk-konformen Geschäftsdokumenten fungiert. Hier wird eine Menge von XML-Formatvorschriften und Elementen, die jede BizTalk-Nachricht berücksichtigen muss, definiert. Die zweite Komponente ist das BizTalk Repository, wo XML-Schemata in einer Datenbank abgelegt sind und jedem Interessierten über das Portal www.biztalk.org zur Verfügung gestellt werden. Die letzte Komponente ist der BizTalk Server, der die zentrale Verwaltung und die Konfiguration des ‚Message-Routings' übernimmt. Der BizTalk-Server regelt, dass die Nachrichten ankommen und geparst werden können, während das Framework zusammen mit dem Repository sicherstellt, dass die Nachrichten verstanden werden. Im folgenden wird nur auf das BizTalk Framework eingegangen, für weitere Informationen zum

[18] Informationen zu Kerboros sind unter http://web.mit.edu/kerberos/www/ zu finden.

BizTalk Server ist auf die Homepage von Microsoft und zum Repository auf die Webseite der Organisation verwiesen[19].

4.6.1 Spezifikation von BizTalk Framework

Eine BizTalk-Applikation ist eine Anwendung, welche BizTalk-Dokumente lesen und schreiben sowie mit einem BizTalk Framework Compliant Server (BFC) kommunizieren kann. Um BizTalk anzuwenden, ist ein BFC-Server notwendig – mit anderen Worten der BizTalk Server 2000 von Microsoft. Zwar wird in der BizTalk-Framework-Dokumentation nur darauf hingewiesen, dass der Server BFC-kompatibel sein muss, derjenige von Microsoft sei jedoch nicht verbindlich. Nach Angaben von [Fitzgerald 2001, S. 256] bezeichnet sich BizTalk als eine Erweiterung von SOAP, und verwendet diesen, als hätte er sich schon als soliden Standard durchgesetzt. SOAP ist aber von einem grossen Teil der Internet-Community nicht als Standard akzeptiert – noch nicht. Um die Eigenschaften von BizTalk aufzuzeigen, wird nun auf ein einfaches Beispiel aus [Microsoft 2000, S. 10] für ein BizTalk-Dokument eingegangen. Bei diesem kurzen Beispiel handelt es sich um eine Bestellung des Buches ‚Essential BizTalk' welches der Käufer ‚Book Lovers' beim Lieferanten ‚Book Orders' mittels einer SOAP-Nachricht bestellt. Eine komplette BizTalk-Nachricht enthält ein BizTalk-Dokument, sowie optional HTTP-Headers, MIME-Headers oder Attachments.

```
<SOAP-ENV:Envelope
 xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
 xmlns:xsi="http://www.w3.org/1999/XMLSchema-instance">
     <SOAP-ENV:Header>
          <eps:endpoints SOAP-ENV:mustUnderstand="1"
           xmlns:eps="http://schemas.biztalk.org/btf-2-0/endpoints"
           xmlns:agr="http://www.trading-agreements.org/types/">
               <eps:to>
                    <eps:address xsi:type="agr:department">Book Orders</eps:address>
               </eps:to>
               <eps:from>
                    <eps:address xsi:type="agr:organization">Book Lovers</eps:address>
               </eps:from>
          </eps:endpoints>
          <prop:properties SOAP-ENV:mustUnderstand="1"
           xmlns:prop="http://schemas.biztalk.org/btf-2-0/properties">
               <prop:identity>uuid:74b9f5d0-33fb-4a81-b02b-5b760641c1d6</prop:identity>
               <prop:sentAt>2000-05-14T03:00:00+08:00</prop:sentAt>
               <prop:expiresAt>2000-05-15T04:00:00+08:00</prop:expiresAt>
               <prop:topic>http://electrocommerce.org/purchase_order/</prop:topic>
          </prop:properties>
     </SOAP-ENV:Header>
     <SOAP-ENV:Body>
          <po:PurchaseOrder xmlns:po="http://electrocommerce.org/purchase_order/">
                <po:Title>Essential BizTalk</po:Title>
          </po:PurchaseOrder>
     </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

[19] Für Repository: www.biztalk.org ; Für BizTalk Server: www.microsoft.com/biztalk/default.asp

Bild 4-9: Beispiel eines BizTalk-Dokuments

Da eine BizTalk-Nachricht ebenfalls eine SOAP-Nachricht ist, besteht das Hauptelement aus einem SOAP-Envelope, mit dem dazugehörigen Namespace. In einem zweiten Namespace, dem XML-Schema-Instance, werden mehrere Attribute für die direkte Verwendung in XML-Dokumenten definiert. Wird ein optionaler SOAP-Header verwendet, folgt dieser direkt nach dem ‚Envelope'-Element. In einer BizTalk-Nachricht enthält dieser Header sogenannte BizTags als Unterelemente. Tabelle 4-2 listet die sechs möglichen BizTags auf.

BizTag	Präfix	Namespace
endpoints	eps	http://schemas.biztalk.org/btf-2-0/endpoints
properties	prop	http://schemas.biztalk.org/btf-2-0/properties
services	svc	http://schemas.biztalk.org/btf-2-0/services
manifest	fst	http://schemas.biztalk.org/btf-2-0/manifest
process	prc	http://schemas.biztalk.org/btf-2-0/process
receipt	rct	http://schemas.biztalk.org/btf-2-0/receipt

Tabelle 4-2: BizTags und ihre Namespaces [Fitzgerald 2001, S. 259]

Analog zu SOAP muss auch bei einer BizTalk-Nachricht das ‚mustUnderstand'-Element auf eins oder null gesetzt werden. BizTalk verwendet Namespaces um die Version zu deklarieren – wird eine neue Version lanciert, wechselt auch der Namespace. Die aktuellen Namespaces sind ebenfalls aus Tabelle 4-2 zu entnehmen. Nun folgenen im Header die sechs, zum Teil optionalen BizTags. In obigem Beispiel sind dies nur die beiden Elemente ‚endpoints' und ‚properties', da es dem User überlassen ist, welche optionalen Elemente er in seinem Dokument verwenden will. Tabelle 4-3 zeigt auf, welche Attribute in welchem Element definiert werden.

Element	Attribute
endpoints	Informiert den Server und die Applikationen über das Routing und bei Empfang der BizTalk-Nachricht. Enthält Informationen über die partizipierenden Unternehmen. Dies kann über die DUNS-Nummer oder über die Postadresse erfolgen.
services (optional)	Bittet um eine Verifizierung bei vertrauenswürdigen Dokumenten. Das Attribut ‚mustUnderstand' muss dabei auf eins gesetzt werden.
properties	Enthält eine Dokument-ID, ein Ausstelldatum und ein Verfallsdatum
process (optional)	Liefert einen Kontext für die Beschreibung und Verständlichkeit der Geschäftsprozesse
manifest (optional)	Enthält Informationen über die Lieferbedingung, Anzahl der Pakete oder Transportweg
receipt (optional)	Informiert über die Art und Weise einer gewünschten Empfangsbestätigung

Tabelle 4-3: Attribute der BizTalk-Header-Elemente (in Anlehnung an [Fitzgerald 2001, S. 260-267])

Der Body eines BizTalk-Dokuments enthält das sogenannte Geschäftsdokument. Dieses wird jedoch bei der BizTalk-Spezifiaktion nicht definiert. Es ist dem Verwender überlassen, ob er sich eigene XML-Dokumente anfertigt, solche aus dem Repository übernimmt, oder ob er diejenigen von xCBL oder cXML verwendet. Die einzige Bedingung an die Geschäftsdokumente ist, dass sie keine BizTags enthalten dürfen, da diese immer im Header definiert werden müssen. Der Body kann auch gleichzeitig mehrere dieser Geschäftsdokumente enthalten. Weiter werden bei BizTalk die identischen Fehlermeldungen wie bei SOAP unterstützt.

BizTalk unterstützt weiter die Versendung von Empfangsbestätigungen. Ein BizTalk-‚Receipt' ist eine SOAP-Nachricht, welche den Erhalt einer vorausgegangener BizTalk-Nachricht bestätigt. Dabei werden zwei Arten von Bestätigungen unterschieden. Die erste Variante ist der ‚DeliveryReceipt'. Hier wird lediglich der Eingang des Dokuments mittels der Dokumenten-ID und der Empfangszeit bestätigt. ‚CommitmentReceipt' geht da noch weiter und informiert darüber, ob der Server das Dokument auch angenommen und weiterverarbeitet hat. Diese Informationen befinden sich immer im Header des Dokuments, der Body bleibt immer leer. Für weitere Informationen zu den Empfangsbestätigungen und über die Möglichkeit der Anhängung von Attachments ist auf die BizTalk-Spezifiaktion [Microsoft 2000] verwiesen.

4.6.2 Zukuftsaussichten für BizTalk

Das BizTalk Framework kombiniert mit dem BizTalk Server 2000 ist eines der vollständigsten B2B-Systeme, die es auf dem Markt gibt [vgl. Fitzgerald 2001, S. 270]. Zusätzlich kommt hinzu, dass Commerce One und Microsoft im April 2001 eine strategische Allianz unterzeichnet haben, welche eine künftige Zusammenarbeit bei der Schaffung von XML-basierenden e-Commerce Standards und Initiativen beinhaltet [vgl. Cecere et al. 2001]. Aus diesem Grunde wird es wohl nur noch eine Frage der Zeit sein, bis die grosse Anzahl an xCBL-Dokumenten auch im Repository von BizTalk zu finden sind, zusätzlich wird deren Verwendung bei einer BizTalk-Nachricht bereits unterschützt.

Jedoch ist die Lösung von Microsoft ziemlich kostspielig, da der BizTalk Server 2000 auf Windows 2000 Server, SQL Server 2000 und Visio 2000 aufbaut [vgl. Fitzgerald 2001, S. 255]. Besitzt ein Unternehmen bereits diese Voraussetzungen, kann es zu einer günstigen Lösung kommen. Muss die Firma sich jedoch alle diese Produkte anschaffen muss mit einem Preis von mehreren zehntausend Dollars gerechnet werden [vgl. Fitzgerald 2001, S. 270].

4.7. Business Transaction Protocol

Bea Systems, Inc. ist eines der führenden Softwareunternehmen für e-Business Infrastrukturen. Ende Januar 2001 verkündete BEA die Gründung eines technischen Komitees für Geschäftstransaktionen. Dies erfolgte durch OASIS in Zusammenarbeit mit Sun Microsystems, Interwoven und Bowstreet. BEA offerierte dabei ihr Business Transaction Protocol (BTP) um die weltweite Verwendung offener Industriestandards zu fördern und umfangreiche B2B-Entwicklungen zu erleichtern. BTP wird bereits bei ‚BEA WebLogic Collaborate' als eine Schlüsseltechnologie mitgeliefert und bildet die Basis für das Standardisierungsvorhaben [vgl. Ehrhardt 2001].

4.7.1 Spezifikation von BTP

Um die Eigenschaften von BTP aufzuzeigen, wird ein Modell für Transaktionsprotokolle, inklusive der Rollen, welche die unterschiedlichen Systemkomponenten während dem Lifecycle einer Transaktion einnehmen, vorgestellt [vgl. Dalal/Takacsi-Nagy 2001, S. 7ff]. Das Modell besteht aus Handelspartnern, welche eine Entität, wie z.B. eine Unternehmung, repräsentieren und an einer oder mehreren Geschäftstransaktionen teilnehmen. Dabei besitzt der Handelspartner einen B2B-Server, welcher Applikationen für den Nachrichtenaustausch

„hostet“. Eine Transaktion ist eine Serie von Nachrichten, die zwischen verschiedenen Handelspartner ausgetauscht werden, um einen gemeinsamen Geschäftsprozess durchzuführen. Eine Transaktion wird immer bei einer Applikation eines Handelspartner initiiert, daher wird diese auch als Initiator bezeichnet. Die Applikationen der Handelspartner, die bei der Transaktion teilnehmen, werden als partizipierende Partner bezeichnet. Der B2B-Server eines Handelspartners unterstützt einen Transaktionskoordinator, der die Transaktionen mit BTP durchführt. Dabei unterscheidet man den Hauptkoordinator, der die Aufforderung zur Errichtung einer Transaktion vom Initiator erhält, und die untergeordneten Koordinatoren, welche bei einer Transaktion mit dem Hauptkoordinator kooperieren. Bild 4-10 veranschaulicht das Modell und enthält zusätzlich den Lifecycle einer Transaktion.

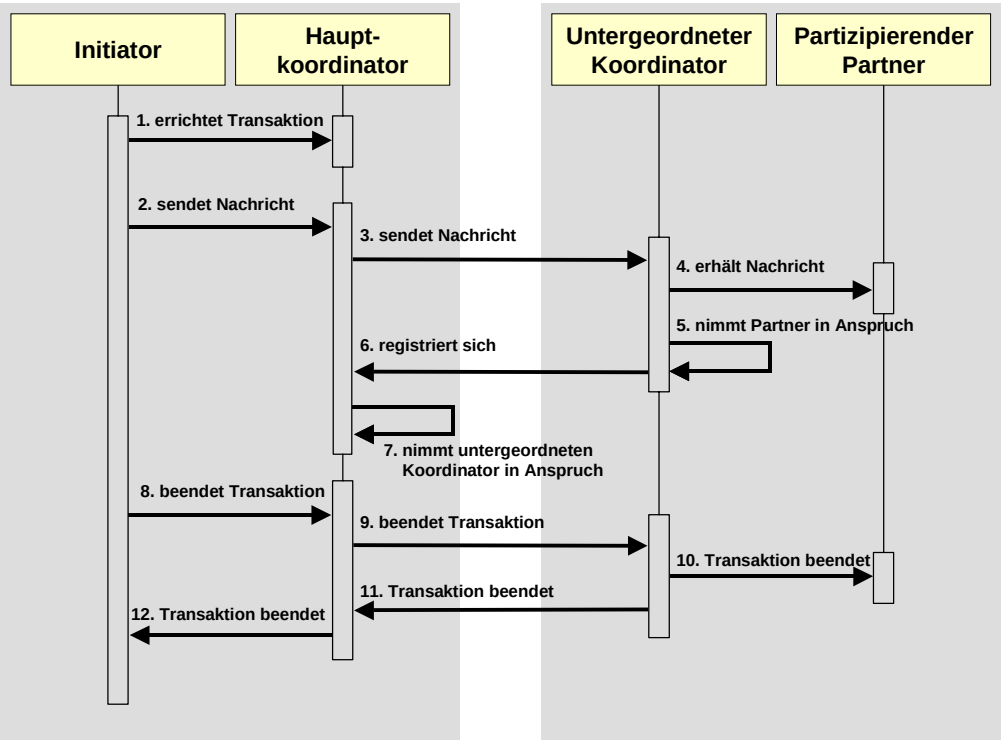


Bild 4-10: Der Lifecycle einer Transaktion [Dalal/Takacsi-Nagy 2001, S. 8]

Wenn eine Transaktion durch den Initiator errichtet wird, erhält diese durch den Hauptkoordinator eine global eindeutige ID. Die Transaktion ist somit in einem aktiven Zustand, was nun einen Austausch von Geschäftsnachrichten erlaubt. Jede Nachricht, die in einer Transaktion gesendet wird, enthält einen Transaktionskontext, welcher die Koordinatoren sowohl bei der Identifizierung und Zuordnung der Nachricht als auch bei der Ausführung der richtigen Aktion unterstützt. Der Transaktionskontext besteht aus drei Komponenten. Der ‚TransaktionIdentifier’ ist die erste Komponente und enthält eine global eindeutige ID, welche vom Hauptkoordinator generiert wird. Die zweite Komponente ist der ‚TransactionTyp’, welcher

die Geschäftstransaktion definiert. ‚MainCoordinatorsURL' ist die letzte Komponente und enthält Informationen über den Hauptkoordinator der Transaktion.

Die Handelspartner werden bei der Transaktion mittels einer Nachricht, welche den Transaktionskontext beinhaltet, angestossen. Wenn die Nachricht den Server des Handelspartners erreicht hat, fängt der Koordinator diese Nachricht und extrahiert daraus den Transaktionskontext. Ist die Transaktion für den Koordinator unbekannt, merkt dieser sich die Transaktions-ID und speichert den Transaktionskontext, bevor er die Nachricht an die Applikation des Handelspartners zur Weiterverarbeitung weiterleitet. Danach nimmt der Koordinator die Applikation des angestossenen Partners in Anspruch - die Applikation wird somit zum partizipierenden Partner. Gleichzeitig wird der Koordinator zum untergeordneten Koordinator, da dieser durch einen ‚RegisterRequest' dem Hauptkoordinator bekannt gibt, dass er in die Transaktion involviert ist. Der ‚RegisterRequest' enthält sowohl die URL des untergeordneten Koordinators als auch den Transaktionskontext.

Der Initiator ist der einzige Partner, der die Erlaubnis hat eine Transaktion zu beenden. Hat er die Absicht dies tun, sendet dieser einen ‚TerminateRequest' an den Hauptkoordinator. Darauf führen sowohl der Hauptkoordinator als auch alle untergeordneten Koordinatoren gemeinsam das ‚TerminationProtokoll' aus. Sobald das ‚TerminationProtkoll' gestartet wurde, befindet sich die Transaktion im Status ‚Terminating'. Erst wenn das Protokoll beendet ist wird auf den Status ‚Terminated' gewechselt. Für weitere Informationen zum ‚TerminationProtokoll' ist auf [Dalal/Takacsi-Nagy 2001, S. 12ff] verwiesen. Tabelle 4-4 beschreibt die sieben Nachrichten, die für einen Transaktionskoordinator erforderlich sind [vgl. Dalal/Takacsi-Nagy 2001, S. 19f].

© HSG / IWI / CCBN / 18

Nachricht	Beschreibung
Create Transaction	• Wird vom Initiator an den Koordinator gesendet. Dieser generiert eine global eindeutige ID. Der Koordinator nimmt darauf die Rolle des Hauptkoordinators für die neu kreierte Transaktion ein.
Register Request	• Wird vom untergeordneten an den Hauptkoordinator zur Registrierung gesendet. • Enthält den Transaktionskontext und die URL des untergeordneten Koordinators.
Leave Transaction	• Wird vom partizipierenden Partner an seinen lokalen Koordinator gesendet, um die Absicht eine Transaktion zu verlassen anzuzeigen. • Enthält den Transaktionskontext.
Unregister Request (optional)	• Wird vom untergeordneten an den Hauptkoordinator gesen-det, falls dieser eine ‚LeaveTrasaction'-Nachricht erhält. • Enthält den Transaktionskontext und die URL, welche bei der Registrierung verwendet wurde.
Terminate Request	• Wird bei der Bekanntgabe einer Transaktionsbeendung versendet. • Wird sowohl vom Initiator an den Hauptkoordinator als auch von diesem an die untergeordneten Koordinatoren versendet. • Enthält den Transaktionskontext, sowie ein Resultat über den Transaktionsablauf (success oder error).
Terminate Completion Notification	• Wird vom untergeordneten an den Hauptkoordinator gesendet zur Bekanntgabe der Vollendung des ‚Termination'-Protokolls auf seiner Seite. • Enthält den Transaktionskontext, sowie die URL des untergeordneten Koordinators.
Query Status	• Wird vom untergeordneten an den Hauptkoordinator gesendet, um den Transaktionsstatus wieder zu erlangen.

Tabelle 4-4: Beschreibung der BTP-Nachrichten
(in Anlehnung an [Dalal/Takacsi-Nagy 2001, S. 19f])

4.7.2 Zukunftschancen für BTP

Da es sich bei BTP noch um ein sehr junges Standardisierungsvorhaben handelt, wird es schwierig dessen Zukunftschancen zu beschreiben. Auch die offizielle Dokumentation zum BTP-Standard ist noch nicht veröffentlicht worden. Einige Voraussetzungen, welche auf ein weite Akzeptanz schliessen lassen, werden bereits erfüllt. So wird BTP bereits beim ‚Bea WebLogic Collaborate' mitgeliefert und auch andere Standardisierungsvorhaben wie SOAP und ebXML werden durch dieses Protokoll unterstützt [vgl. Dalal/Takacsi-Nagy 2001, S. 7]. Lang anhaltende Geschäftstransaktionen über mehrere Unternehmen hinweg stellen eine einmalige Herausforderung an eine Business Collaboration Infrastructure. Die interdependenten Transaktionen entlang mehrerer Unternehmen müssen koordiniert werden, um sicherzustellen, dass das Ergebnis einer Transaktion zuverlässig ist. Mit BTP wird eine Lösung für dieses Problem vorgeschlagen [vgl. Cover/OASIS 2001].

4.8. Customer Profile Exchange

Customer Profile Exchange (CPExchange) ist ein auf XML basierender Standard für die Beschreibung und den zwischenbetrieblichen Austausch von Kundendaten und -profilen. Dieser wurde vom CPExchange Network entwickelt, das im November 1999 in San Francisco gegründet wurde. Mitglieder sind unter andrem IBM, Oracle, Siebel, Sun/Netscape sowie Lucent Technologies, Broadvision und Vignette [vgl. Ihlenfeld 2001]. Durch den Austausch von Kundendaten, wie beispielsweise Name, Personalausweisnummer, Steuernummer, Einkommen oder Kaufverhalten, kann diesem somit ein besserer Kundenservice angeboten werden. Durch einen gezielten Einspruch sollen die Kunden eine Weitergabe der Daten verhindern können und somit ihren Datenschutz sicherstellen [vgl. Bleich 2000].

4.8.1 Spezifikation von CPExchange

Es gibt viele Informationsarten und -quellen zur Erstellung eines Kundenprofils. Informationen können beim Surfen, bei einem Anruf im Call Center oder auch beim Kaufverhalten gewonnen werden. Um diese Informationsflut zu organisieren, stellt CPExchange verschiedene Kategorien und Unterkategorien zur Verfügung [vgl. Bohrer/Holland 2000, S. 22ff].

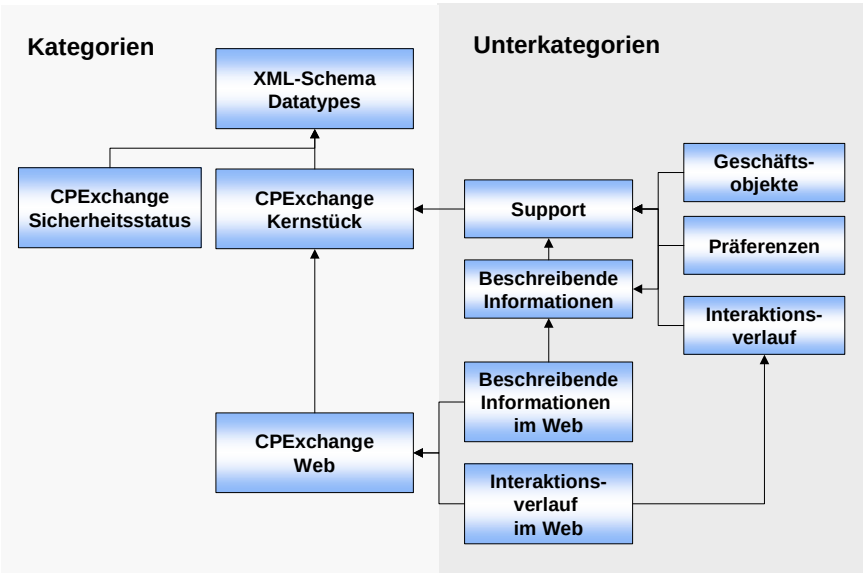


Bild 4-11: Kategorisierung bei CPExchange (in Anlehnung an [Bohrer/Holland 2000])

Jeder Datentransfer mit CPExchange kann Informationen über die Privatsphäre eines Kunden enthalten, dies jedoch nur mit der Zustimmung des Betroffenen. Werden solche Daten ausgetauscht, erhält das XML-Dokument einen ‚PrivacyPolicyHeader', der Informationen sowohl über den sendenden als auch über den empfangenden Geschäftspartner, über die

Rechtslage der beiden Partner, sowie die Deklaration des Sicherheitsstatus der nachfolgenden Daten beinhaltet. Durch die Deklaration kann bestimmt werden, wie, wer und wie lange die Daten künftig weiterverwendet werden können.

In der Unterkategorie ‚Support' wird jedem Kunden eine ID zugeordnet. Weiter wird angegeben zu welchem Zeitpunkt das Dokument erzeugt und wann es das letzte Mal modifiziert wurde. Zusätzlich enthält das Dokument eine Zeitspanne, in der die Gültigkeit der Daten angegeben wird. Werden neue Attribute eines Kunden erfasst, beispielsweise die Haarfarbe, muss dies dem System des Partners durch Tags mitgeteilt werden, damit die Daten korrekt gespeichert werden können.

Die Unterkategorie ‚Beschreibende Informationen' enthält die grundlegenden Informationen über eine Person, Organisation oder über die Rolle die von diesen wahrgenommen wird. Solche Informationen sind beispielsweise der Name, die Adresse, die Nationalität, die Telefonnummer und e-Mail-Adresse, die bevorzugte Sprache oder auch demografische Informationen wie Geschlecht, Geburtsdatum, Zivilstand, Beruf, Hobbys, Raucher/Nichtraucher oder Ausbildungsstand. In diese Unterkategorie fallen zusätzlich Informationen über das Surfverhalten des Kunden, wie die URL, Anzahl Besuche, verwendete Links oder letzter Zugriff auf eine bestimmte Internetseite.

Jede Handlung des Kunden wird in der Unterkategorie ‚Interaktionsverlauf' protokolliert. Dies können Begegnungen bei einem Meeting, aber auch per Telefonanruf, Fax, e-Mail oder einfach bei einem Besuch auf der Webseite des Unternehmens sein. In einer weiteren Kategorie werden die Präferenzen des Kunden erfasst. Diese können entweder aus dem Verhalten des Kunden abgeleitet oder explizit auf der Webseite nachgefragt werden. Die letzte Kategorie enthält sämtliche Informationen über die Geschäftsobjekte. Darin werden die gekauften Produkte mit Name und Nummer, sowie die Zahlungsart erfasst. Gibt der Kunde seine Kreditkarten- oder Kontonummer an, wird auch diese darin abgelegt. Ein sehr umfangreiches Musterdokument kann auf www.cpexchange.org heruntergeladen werden.

4.8.2 Zukunftsaussichten für CPExchange

Auf der einen Seite ist es sicherlich spannend zu sehen, welche Möglichkeiten mit diesem Standard geschaffen werden. Ein illustrierendes Beispiel gibt der Mitinitiator Brandley Husick: Wenn ein Kunde Probleme mit seinem neuen Laptop hat, kann dieser bei irgendeinem Kundensupport anrufen, auch wenn es derjenige der Konkurrenz ist, und der Mitarbeiter

am Telefon weiss, ohne nachfragen zu müssen, um welches Gerät es sich handelt, und wo es gekauft wurde [vgl. Bleich 2000]. Durch den Austausch von umfassenden Kundenprofilen soll vorwiegend ein besserer Kundenservice angeboten werden.

Doch auf der anderen Seite sind nicht alle von dieser Vorstellung fasziniert. Obwohl der Nutzer einen Sicherheitsstatus für seine Daten eingeben kann, ist dies noch lange keine Garantie, dass seine Privatsphäre nicht verletzt wird. Da geht das W3C schon weiter. Sie sind an der Entwicklung eines P3P-Standards[20], der verstärkt auf den Datenschutz setzt. Dieser bietet eine Transparenz für den Surfer, da dieser Datenschutzeinstellungen in einem zusätzlichen Browserfenster manuell vornehmen kann. Um diesem Standard jedoch eine Chance zu geben, benötigt es die Unterstützung der Browserhersteller. Microsoft plant in seiner nächsten erscheinenden Browserversion diesen Standard zu implementieren [vgl. Schmidt 2001]. Laut der Federal Trade Commission (FTC) wird vor allem die Akzeptanz beim Enduser über die Weiterentwicklung von CPExchange entscheiden, diese sei zur Zeit jedoch nicht vorhanden [vgl. Thibodeau 2001]. Im europäischen Raum kommen zusätzlich noch rechtliche Aspekte hinzu, während in Amerika beim Datenschutz nur die Selbstkontrolle vorherrscht.

4.9. Universal Description, Discovery, and Integration

Universal Description, Discovery, and Integration (UDDI) ist ein Projekt, das von Ariba, IBM und Microsoft initiiert wurde. Allen Unternehmungen, die im Web vertreten sind, wird hier die Möglichkeit geboten, auf einer einheitlichen weltweiten Plattform, die eigene Firma darzustellen und Kontakte zu anderen Firmen zu finden [vgl. UDDI.org 2000b].

4.9.1 Aufbau von UDDI

UDDI präsentiert eine neue Form eines Businessverzeichnis, welches aus drei Teilen besteht: Weisse, Gelbe und Grüne Seiten. Die weissen Seiten enthalten allgemeine Angaben, wie sie beispielsweise auch im Telefonbuch sind. Attribute die Unternehmen darin angeben sind zum Beispiel der Name, die Adresse und die üblichen Kontaktinformationen. Die gelben Seiten geben Aufschluss über die Produkte und Leistungen, die das Unternehmen anbietet. Schliesslich informieren die neuen grünen Seiten über die Protokolle, welche das Unternehmen im elektronischen Geschäftsverkehr nutzt [vgl. UDDI.org 2000a].

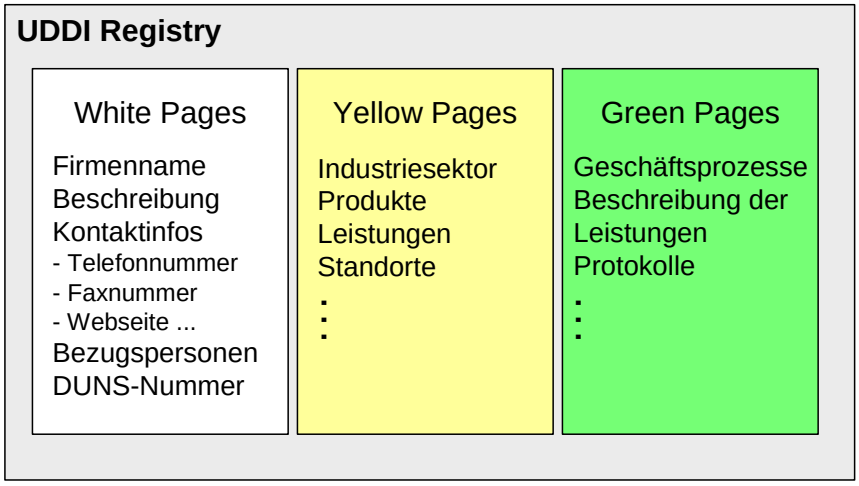


Bild 4-12: UDDI Businessverzeichnis [UDDI.org 2000a]

Die UDDI Business Registry (UBR) ist ein globales, öffentliches Online-Verzeichnis und gibt Unternehmen jeder Grösse die Möglichkeit, ihre Angebote standardisiert zu beschreiben, Einträge anderer Firmen zu durchsuchen und Protokolle, welche potentielle Geschäftspartner im e-Business nutzen, zu recherchieren [vgl. ECIN 2001]. Nicht nur im B2B-Bereich soll UDDI einen grossen Einfluss haben, auch Surfer bei ihrer Suche sollen unterstützt werden. Ein beliebtes Beispiel ist der Pizzabäcker. Steht seine Dienstleistung in der Registry, so könnte der Surfer mit der Suche nach ‚Pizza essen' und der Eingabe seines Standorts die nächste Pizzeria finden – und dies weltweit genormt. In dieser Weise können Konsumenten, Getränkelieferanten und die Hersteller von Pizzaöfen gleichermassen mit dem Verzeichnis arbeiten [vgl. Borchers 2001].

4.9.2 Zukunftsaussicht für UDDI

Bereits mehr als 7000 Unternehmen und Organisationen sind ein Bestandteil der UDDI-Community geworden. Zudem wurden mehrere tausend Leistungen und Produkte in der Registry eingetragen. Trotz diesen erfolgsversprechenden Zahlen, repräsentieren die meisten Leistungen nicht die Wirklichkeit. Für geschäftskritische Prozesse wie beispielsweise die Kreditgenehmigung, der Bestellstatus oder die Rechnungsstellung werden keine Leistungen angeboten. Zur Zeit müssen Unternehmen dies lediglich als eine Möglichkeit wahrnehmen, ihre Leistungen in eine vorgegebene Taxonomie einzutragen. Einige Unternehmen verwenden UDDI-Verzeichnisse nicht, da sie ihren komparativen Konkurrenzvorteil dadurch verlieren würden [vgl. Plummer 2001].

[20] Informationen zum P3P-Standard sind unter http://www.w3.org/P3P/ zu finden.

Anderer Unternehmen sehen gerade darin ihre Chance mit UDDI die richtigen, geeigneten Kooperationspartner zu finden und somit einen komparativen Konkurrenzvorteil gegenüber anderen Unternehmen zu schaffen [vgl. Sleeper 2001]. Von grosser Wichtigkeit für das Überleben von UDDI ist, dass das Verzeichnis weiter wächst. Dies soll dadurch geschehen, dass sowohl die Registrierung als auch die Suche weiterhin kostenlos bleibt [vgl. Reinwarth 2001]. Zudem wird UDDI von anderen XML-Standards wie SOAP [vgl. McKee et al. 2001, S. 56ff] oder BTP [vgl. Dalal/Takacsi-Nagy 2001, S. 19] unterstützt.

4.10. Data Universal Numbering System

Die DUNS-Nummer (Data Universal Numbering System) ist ein neunstelliger Zahlencode, der 1962 von Dun & Bradstreet entwickelt wurde, um Unternehmen weltweit eindeutig identifizieren zu können. Nahezu 20 Millionen Datensätze wurden bereits in der Datenbank von Dun & Bradstreet erfasst [vgl. FIZ-Technik 2001]. Gerade im B2B-Bereich wird dieses einzigartige Identifizierungsmerkmal vermehrt eingesetzt, um die DUNS-Daten mit Informationen anderer Unternehmensdaten zu verknüpfen [vgl. D&B 2000].

4.10.1 Spezifikation von DUNS

Die Worldbase von Dun & Bradstreet wird in 18 Einzeldatenbanken den Unternehmen bereitgestellt. So hat beispielsweise die Datenbank ‚Germany' mehr als eine halbe Million Datensätze oder die Datenbank ‚Switzerland, Austria' ungefähr 360'000 Einträge [vgl. D&B 1998]. Jeder Datensatz enthält die DUNS-Nummer und weitere vorgegebene Attribute, welche aus Bild 4-13 zu entnehmen sind.

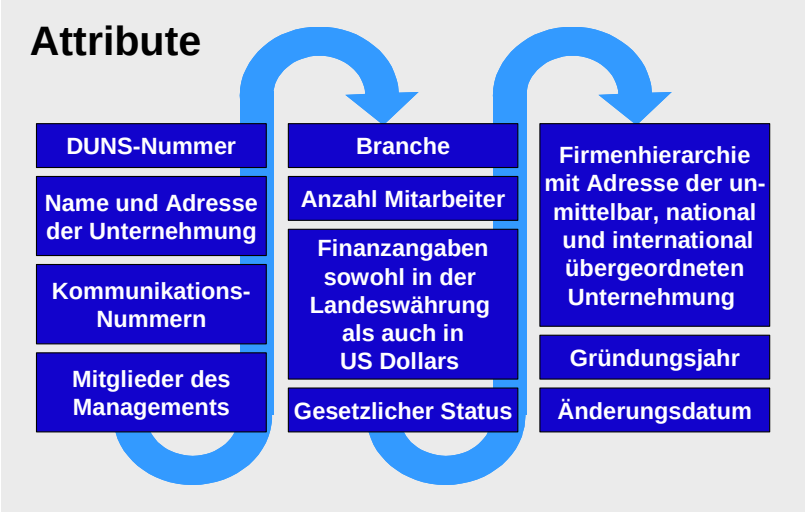


Bild 4-13: Attribute eines DUNS-Dokuments[21]

Die Dun & Bradstreet Datenbanken werden genutzt für die Marktforschung, die Untersuchung von Konzernverflechtungen oder helfen bei der Suche nach Lieferanten, Geschäftspartner oder Neukunden [vgl. FIZ-Technik 1998]. Durch die DUNS-Dokumente sind die Informationen über die Unternehmen weltweit standardisiert.

4.10.2 Zukunftsaussichten für DUNS

Mit der DUNS-Nummer werden Unternehmen eindeutig ihren Muttergesellschaften, Schwestern und Filialen zugeordnet. Darüber hinaus ist eine schnelle Identifizierung der jeweiligen Standorte und Wirtschaftszweige möglich. Die Dun & Bradstreet Datenbanken bestehen seit fast vierzig Jahren und werden von führenden Organisationen wie ISO oder ANSI bereits seit mehreren Jahren genutzt. Die DUNS-Nummer kam bereits via EDI zum Einsatz, beispielsweise bei der Beschaffung von Waren und Dienstleistungen der Regierung der Vereinigten Staaten [vgl. D&B 2000] und wird auch einen grossen Stellenwert beim zwischenbetrieblichen Datenaustausch via XML-Dokumenten einnehmen. XML-Standards wie beispielsweise UDDI unterstützen die Verwendung der DUNS-Nummer.

4.11. WebService Definition Language

Die WebService Definition Language entstand aus den beiden Sprachen NASSL von IBM und SDL von Microsoft. Die derzeit aktuellste Version wurde im März 2001 von namhaften IT-Unternehmen dem W3C vorgelegt [vgl. Giehl 2001]. WSDL ist ein auf XML-basierendes Vokabular für die Beschreibung von WebServices. Die Kommunikation und den Nachrichtenaustausch wird jedoch mittels SOAP vorgenommen. Ebenso bestimmt WSDL nicht, wie solche Services veröffentlicht und weiterverbreitet werden – dies übernimmt UDDI [vgl. Hoyer 2001].

4.11.1 Spezifikation der WSDL-Dokumente

Ein WSDL-Dokument definiert Services als eine Ansammlung von Endpunkten eines Netzwerkes, den sogenannten ‚Ports', und den Nachrichten, welche beim Datenaustausch verwendet werden [vgl. W3C 2001b]. Um die Verständlichkeit von WSDL zu vereinfachen, werden in der Tabelle 4-5 die sieben Element eines WSDL-Dokuments kurz erläutert .

[21] Ein Musterdokument steht bei [D&B 1998] zur Verfügung.

Element	Beschreibung
Types	Ein Speicher, wo DTD abgelegt werden.
Message	Eine Nachricht (DTD) die kommuniziert wird.
Operation	Eine Beschreibung einer Aktion, die durch den Service unterstütz wird.
Port Type	Ein Set von Operationen, die durch einen oder mehrer Endpunkte unterstützt werden.
Binding	Eine konkrete Spezifikation des Protokolls und des Datenformats für einen einzelnen Port Type.
Port	Ein einzelner Endpunkt, der aus einer Kombination von Bindings und Netzwerkadressen definiert wird
Service	Eine Ansammlung von verwandten Endpunkten

Tabelle 4-5: Elemente von WSDL-Dokumenten

Im folgenden wird WSDL anhand eines einfachen Beispiels aus [W3C 2001b] klarer verdeutlicht. Ein Unternehmen bietet einen WebService an, welcher eine Abfrage von Aktienkursen beinhaltet. Aus diesem Grund wird ein ‚Stockquote'-Service definiert. Der Service verwendet SOAP mit HTML als Transportprotokoll und bietet eine Methode ‚GetLastTradePrice'. Diese Methode erwartet eine ‚TickerSymbol' vom Typ ‚String' und gibt das Element ‚Price' vom Typ ‚Float' zurück. Bei den Types werden nun die verwendeten Datentypen mittels XML-Schemata definiert. In der ‚Message' werden darauf die auftretenden Nachricht in logische Einheiten unterteilt und jedem dieser Einheiten ein Typ zugeordnet. Im Beispiel sind dies die Einheiten ‚GetLastTradePriceInput' und ‚GetLastTradePriceOutput' welchen die Typen ‚String' und ‚Float' zugeordnet werden. Bei den ‚PortTypes' werden die möglichen Operationen und die dazugehörigen Nachrichten festgelegt. Zusätzlich lässt sich hier auch eine Fehlermeldung definieren. Bei ‚Binding' wird nun ein ‚PortTyp' an ein konkretes Protokoll (SAOP) angebunden. Bild zeigt eine solche Bindung auf.

```
<binding name="StockQuoteSoapBinding" type="tns:StockQuotePortType">
     <soap:binding style="document" transport="http://schemas.xmlsoap.org/soap/http"/>
     <operation name="GetLastTradePrice">
          <soap:operation soapAction="http://example.com/GetLastTradePrice"/>
          <input> <soap:body use="literal"/>
          </input>
          <output>
          <soap:body use="literal"/>
          </output>
     </operation>
</binding>
```

Bild 4-14: Binding eines WSDL-Dokuments

Das ‚Binding'-Element bildet das Verbindungsglied zwischen technischen Definitionen (Types) und formalen Beschreibungen (Message, PortType). Das ‚Service'-Element fasst schliesslich mehrere verwandte ‚Ports', welche einer Bindung eine klar definierte Adresse zuordnen, zusammen. Weitere Ausführungen zu diesem Beispiel sind unter [W3C 2001b] zu finden.

4.11.2 Zukunftsaussichten von WSDL

WSDL findet starke Unterstützung von zahlreichen, namhaften IT-Unternehmen und wird bereits bei UDDI verwendet. Somit ist die Weiterverwendung von WSDL stark von UDDI abhängig. Ebenso starke Unterstützung findet WSDL in den neuen .Net-Services von Microsoft.

Abschliessend ist zu sagen, dass WSDL nur mit den Standardisierungsvorhaben von SAOP und UDDI – also im Dreiklang – überlebensfähig ist. Da diese Vorhaben bereits stark etabliert sind, wird WSDL mit hoher Wahrscheinlich der neue Standard für die Beschreibung und Definition von WebServices sein.

4.12. BMEcat

Ende 1998 wurde eine Arbeitsgruppe aus wissenschaftlichen Instituten, Einkäufern grosser Unternehmen und dem BME (Bundesverband Materialwirtschaft, Einkauf und Logistik) gegründet um diesen neuen Standard zu spezifizieren [vgl. Hümpel/Schmitz 2001, S. 205]. BMEcat ist ein auf der XML-Technologie basierender Standard für die Beschreibung und den Austausch von Produktkatalogdaten im e-Business und ist der mit Abstand am weitest verbreitete Austausch-Standard für elektronische Produktkataloge im deutschsprachigen Raum [BMEcat 2001]. Mit der Hilfe von BMEcat spezifischen DTDs bekommen BMEcat-Katalogdokumente eine bestimmte Semantik und können eindeutig interpretiert werden [Pastoors et al. 2001].

4.12.1 Struktur eines BMEcat-Dokuments

Jedes Katalogdokument im BMEcat-Format wird mit einem ‚BMECAT'-Element eingeleitet. und besteht aus einem Header und einem Body. Der Header enthält transaktionsunabhängige Daten, wie Versionsnummer und Informationen über Kunde, Lieferant und Vertrag. Aus Bild 4-14 sind die im Header geregelten Informationen zu entnehmen.

```
<header>
          <catalog>
                    <language>DE</language>
                    <catalog_id>123</catalog_id>
                    <catalog_version>120</catalog_version>
                    <catalog_name>XML-Diplomarbeit FErni</catalog_name>
                    <currency>CHF</currency>
          </catalog>
          <buyer>
                    <buyer_name>Käufer und Partner</buyer_name>
                    <address type="buyer">
                              <name>Käufer und Partner</name>
                              <contact>Herr Käufer</contact>
                              <street>Kaufhaus 2</street>
                              <zip>1234</zip>
                              <city>Kaufstadt</city>
                              <country>CH</country>
                    </address>
          </buyer>
          <supplier>
                    <supplier_name>Lieferer AG</supplier_name>
                    <address type="supplier">
                              <name>Lieferer AG</name>
                              <street>Sepp-Leier-Strasse 3</street>
                              <zip>5678</zip>
                              <city>Lieferhausen</city>
                              <country>CH</country>
                    </address>
          </supplier>
</header>
```

Bild 4-15: Header eines BMEcat-Katalogdokuments [vgl. Pastoors et al. 2001, S. 16]

Im zweiten Teil, dem Body, werden je nach Transaktion[22] unterschiedlich viele Elemente der Produktinformation unterschieden. Im Folgenden wird der Fall betrachtet, bei dem der Partner keinen Produktkatalog besitzt. Es findet also eine Transaktion eines neuen Produktkatalogs statt und nicht eine Erweiterung oder Änderung eines bestehenden. Der Body besteht aus drei Teilen, dem Katalogsystem, dem Artikel und der Zuordnung eines Artikels zu einer Kataloggruppe.

Katalogsysteme werden genutzt, um eine hierarchische Gliederung der einzelnen Artikel zu definieren und somit dem Benutzer durch eine einfache Navigation die Suche eines Artikels zu vereinfachen. Zunächst werden Kataloggruppen erzeugt, untereinander verknüpft und in einer Baumstruktur aufgebaut. Jede Kataloggruppe wird identifiziert mit einer „ID“, ihrem Namen und der ihr übergeordneten Kategoriegruppe.

Danach werden die eigentlichen Artikel mit ihren Daten angegeben. Die Reihenfolge, in der die Artikel aufgelistet werden, spielt zunächst keine Rolle, da eine Zuordnung zu den einzelnen Kataloggruppen erst am Ende eines Dokumentes vorgenommen wird. Attribute eines Artikels sind die Artikelnummer, die Kurzbeschreibung, der Preis etc. pp.. Bild 4-15 zeigt den Aufbau eines BMEcat-Katalogdokumentes auf.

[22] Bei einem BMEcat-Katalogdokument werden drei Transaktionsarten unterschieden: Übertragung eines neuen Katalogs, Aktualisierung von Produkten und Aktualisierung von Preisen [Pastoors et al. 2001].

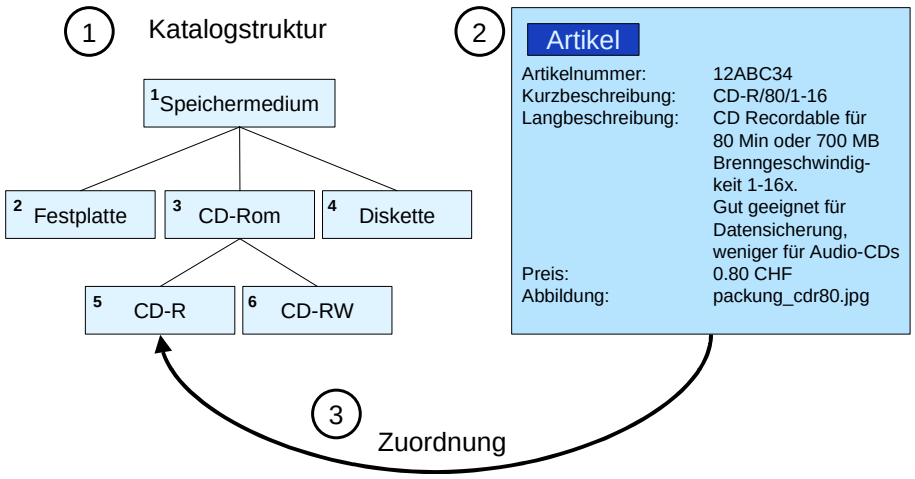


Bild 4-16: Aufbau eines BMEcat Katalogdokuments

Für die Zuordnung eines Artikels in eine Kataloggruppe sind nur sehr wenig Informationen notwendig. Es muss lediglich die Artikelnummer mit der ID der Artikelgruppe verknüpft werden. Je nach Transaktionsart werden unterschiedliche Datenmengen ausgetauscht. So ist zum Beispiel bei der Hinzufügung eines weiteren Artikels keine Katalogstruktur notwendig, da die Datenbank des Benutzers lediglich aktualisiert wird [vgl Pastoors et al. 2001].

4.12.2 Zukunftsaussichten für BMEcat

Die erste Version von BMEcat wurde ein Jahr nach Gründung der Arbeitsgruppe veröffentlicht und Anfang 2002 ist die Erweiterung zur Version 2.0 geplant [vgl. Schmitz/Kelkar 2001]. Grund für die erfreuliche Leistung ist sicherlich die massgebliche Beteiligung der Universität Essen und des Fraunhofer IOA. Bereits eine grosse Anzahl von Softwarehersteller unterstützen den BMEcat-Standard in ihren e-Business-Produkten [vgl. Hümpel/Schmitz 2001]. Zur Zeit ist BMEcat jedoch noch unvollständig, da man sich auf sogenannte C-Artikel konzentriert, deren Modellierung weniger aufwendig ist [vgl. Frank 2001, S. 287]. Zudem müssen noch weitere Geschäftstransaktionen, wie beispielsweise der zwischenbetriebliche Datenaustausch von Bestellungen oder Rechnungen, standardisiert werden, um eine durchwegs automatisierbare e-Business-Lösung realisieren zu können [vgl. Hümpel/Schmitz 2001, S. 217].

4.13. eCl@ss

Da immer mehr Unternehmungen ihre Beschaffungsvorgänge elektronisch abwickeln, ist es notwendig die Datenstrukturen zu standardisieren. Aus diesem Grund erarbeiteten führende

deutsche Unternehmen den Standard eCl@ss zur Klassifikation von Materialien und Warengruppen. Dabei erhalten die erfassten Materialien eine eineindeutige achtstellige Klassifizierungsnummer.

4.13.1 Spezifizierung von eCl@ss

eCl@ss ist gekennzeichnet durch einen vierstufigen, hierarchischen Klassifikationsschlüssel und durch ein aus 18'000 Begriffen bestehendes Schlagwortregister. Für jede dieser vier Hierarchiestufen sind genau zwei Stellen in der eCl@ss-Nummer vorgesehen. Auf jeder Ebene sind somit bis zu hundert Gruppen denkbar.

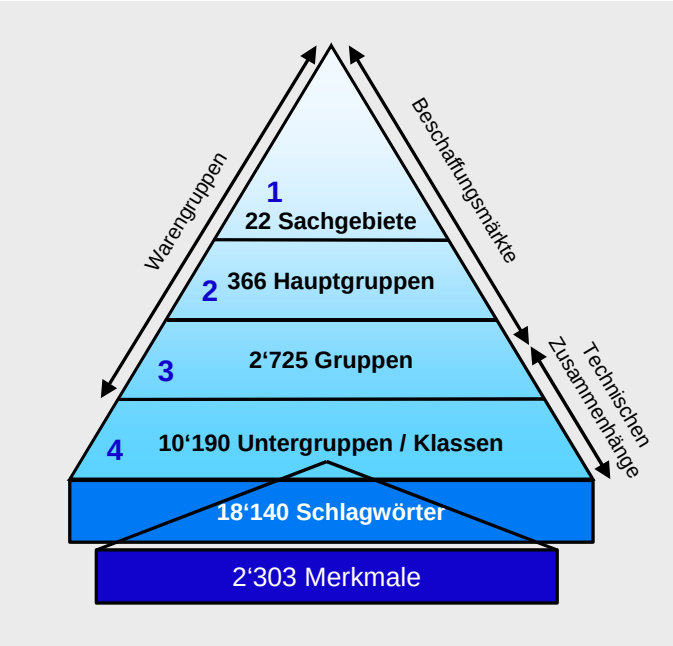


Bild 4-17: Die vier Ebenen der eCl@ss-Hierarchie [vgl. Tradecosmos 2001]

Braucht ein PC-Supporter beispielsweise eine Soundkarte für seinen neuen Computer, so sucht er zuerst im Sachgebiet ‚Kommunikationstechnik, Bürotechnik’ – Nummer 24 – und findet dort die Hauptgruppe ‚Hardware (Informationstechnik)’ – Nummer 01. Diese Hauptgruppe ist wieder in mehrere Gruppen aufgeteilt, darunter auch die für den PC-Supporter relevante Gruppe ‚Peripheriegerät, Zubehör (PC)’ – Nummer 07. Die Gruppen sind weiter in unterschiedliche Klassen gegliedert, wo sich auch die Klasse ‚Soundkarte (PC-Zubehör)’ – Nummer 02 – befindet. Durch diese Klassifikation erhalten alle Produkte einen eindeutigen Klassifikationsschlüssel. In obigem Beispiel sind nun alle Soundkarten für PCs mit der Nummer 24-01-07-02 klassifiziert.

Durch das umfangreiche Schlagwortregister können auch Klassen ohne detaillierte Kenntnis der Hierarchie gefunden werden. So kann beispielsweise der PC-Supporter per Schlagwort ‚Soundkarte' direkt auf die Klasse ‚Soundkarte (PC-Zubehör)' zugreifen. Weiter besteht die Möglichkeit mittels Merkmalen, die richtige Klasse zu finden. Der Nutzer kann über Leistungseigenschaften, wie beispielsweise Anschlüsse oder Ausgangsleistung, das Produkt der spezifischen Klasse zuordnen. Werden die eCl@ss-Nummern via XML-Dokumente weitergenutzt, wird für die Klassifikation die Schreibweise ‚_eclass12345678_' verwendet. Die Soundkarten für einen PC haben somit die Klassifikationsnummer eclass24010702, und dies weltweit standardisiert.

4.13.2 Zukunftsaussichten für eCl@ss

Durch den benutzerfreundlichen Zugang entweder über die Hierarchie oder über Schlagwörter kann sowohl der Experte als auch der gelegentliche Nutzer in der Klassifikation navigieren [vgl. jCatalog 2001]. Zur Zeit ist eCl@ss aber erst im deutschsprachigen Raum ein Begriff zur Materialklassifikation. Zwar ist eCl@ss nun auch in englischer Sprache verfügbar [vgl. IW-Köln 2000, S. 13], aber um sich international zu etablieren, müssen weitere Sprachen unterstützt werden. Schliesslich richtet sich eCl@ss insbesondere auch an KMUs, welche sich oft auf den regionalen Markt, und somit auf die regionale Sprache konzentrieren.

Die Überlebenschancen von eCl@ss sind sehr stark von den Kooperationen mit SAP und BMEcat abhängig. Während sich eCl@ss-Nummern bereits mit BMEcat-Dokumenten versenden lassen, liegt die Zusammenarbeit mit SAP erst in der Gründung von Projektgruppen, welche eine Nutzung von eCl@ss als Klassifikationssystem in SAP erarbeiten [vgl. IW-Köln 2000, S. 13]. Laut Detlef Sacher von Henkel wird eCl@ss bereits innerhalb des Konzerns für Materialien und Dienstleistungen im elektronischen Produktkatalog verwendet, und in ein bis zwei Jahren soll die eCl@ss-Klassifikation in ihren SAP-Systemen eingesetzt werden und somit ihren Mitarbeitern eine weltweit einheitliche Basis zur Verständigung schaffen [vgl. Marwick-Ebner 2001].

4.14. Zuordnung ins Standard-Modell

Sowohl bei ebXML als auch bei RosettaNet handelt es sich um Frameworks. Ebenso bieten beide Standards ein Repository an, welches bei RosettaNet jedoch als Dictionary bezeichnet wird, aber die selbe Funktion wie ein Repository erfüllt. Die Spezifikation der Sprachen steht

nicht im Hauptinteresse der beiden Organisationen ist deshalb bei beiden Standardisierungsvorhaben noch nicht beendet. Bei ebXML sind bis jetzt zwei Dokumente in ihrer eigenen XML-Sprache verfasst worden, weiter werden aber auch andere Sprachen wie beispielsweise diejenige von xCBL unterstützt. Die bei RosettaNet verwendeten Dokumenttypen enthalten eine Fülle von Spezifikationshüllen, deren Inhalte werden jedoch bis anhin nicht weiter spezifiziert [vgl. Frank 2000, S. 54]. Deshalb ist zu sagen, dass beide zwar eine Sprache besitzen, diese jedoch nicht die wichtigsten Elemente in ihrem Framework sind. Die Objekte, die mit den beiden Standards abgedeckt werden können, sind nicht eingeschränkt. Beide lassen zusätzlich eine Anbindung an UDDI zu, obwohl ebXML über eine eigene Registry verfügt. Ebenso gibt es wenige Einschränkungen bei den Prozessen, welche abgebildet werden. Wobei die Ausrichtung auf Commerce und Supply Chain Oberhand erhält. Beide Organisationen geben keinen Fokus für einen Industriesektor vor, obwohl sie sich auf bestimmte Sektoren konzentrieren. So sind dies bei ebXML die Banken, Versicherungen und die Logistikunternehmen und bei RosettaNet die IT-Branche [vgl. Frank 2000, S. 58]. Analog besitzt auch BizTalk von Microsoft ein Framework und ein Repository, jedoch keine eigene Sprache. Die von BizTalk verwendete Sprache ist eine Spezifikation von SOAP. Da es sich bei SOAP um eine Sprache für den Aufruf von Informationen in Datenbanken der Partnerunternehmen handelt, gibt es sowohl beim Industriefokus als auch bei den Objekten und Prozessen keine Einschränkungen.

Bei der Entwicklung von cXML und xCBL steht die Sprache im Mittelpunkt und es ist nur noch eine Frage der Zeit, bis auch sie ihre Frameworks spezifizieren oder diese von einer anderen Organisation übernehmen. So wird beispielsweise bei cXML die Punchout-Sitzung schon ziemlich klar definiert, andere Elemente wie der Onlinekatalog jedoch weniger. Mit den über 600 definierten Elementen im Vocabulary, kann man bei xCBL jedoch von einem Repository sprechen. Sowohl bei xCBL als auch bei cXML findet keine Einschränkung auf einen Industriesektor statt [vgl. Frank 2000, S. 29, 35]. Bei beiden Standards ist jedoch eine Fokussierung auf die Objekte Produkt und Kontrakt, sowie die Ausrichtung auf die Prozesse Commerce, Supply Chain und Finance festzustellen.

Sowohl DUNS als auch eCl@ss sind keine eigentlichen Sprachen, sondern lediglich Identifikationsschlüssel. So kann beispielsweise mittels eCl@ss eine klare Klassifizierung vorgenommen werden und durch DUNS jedes Unternehmen eindeutig identifiziert werden. Es ist jedoch möglich sowohl das Dokument von DUNS als auch die Klassifizierung von eCl@ss

mittels einer XML-Sprache zu spezifizieren. eCl@ss ist dabei auf keinen Industriesektor fokussiert, obwohl es zum jetzigen Zeitpunkt noch Einschränkungen für bestimmte Branchen gibt. Die Weiterentwicklung ist jedoch durch das IW-Köln gewährleistet. Das Objekt des Standardisierungsvorhabens ist das Produkt und die dadurch betroffenen Prozesse sind Commerce, Supply Chain, Maintenance and Repair und PLC. DUNS hingegen hat den eindeutigen Fokus auf das Objekt Partner, welches Einfluss auf die Prozesse Content and Community und Commerce hat.

UDDI ist eine Sprache welche DUNS unterstütz und eine Suche nach einem Geschäftspartner vereinfacht. UDDI besitzt kein Repository, da es nur Datensätze über Unternehmen enthält und keine weiteren Dokumente, die ausgetauscht werden können. Ob es jedoch ein Framework besitzt oder nicht, darüber lässt sich weiter diskutieren, da ein Prozess zur Kontaktierung eines potentiellen Partners vorgegeben wird, dieser aber nicht verpflichtend ist. Bei CPExchange und BMEcat hingegen handelt es sich eindeutig um XML-Sprachen, welche sich auf die Objekte Produkt bei BMEcat und Kunde bei CPExchange beziehen. Durch die klare Konzentration auf ein bestimmtes Objekt aller drei Standards, sind dadurch auch die miteinbezogenen Prozesse bestimmt.

BTP von BEA ist ein Protokoll, welches die Transaktion über mehre Unternehmen hinweg überwacht. BTP ist sowohl eine Sprache, als auch ein Framework. Durch die sieben bereits bestehenden Nachrichten, kann bei BTP auch von einem Repository gesprochen werden, obwohl dies nur sehr begrenzt ist. BTP konzentriert sich vorwiegend auf das Objekt Kontrakt und somit auf die Prozesse Finance, Commerce und Supply Chain.

Bild 4-17 ordnet die behandelten B2B-Standards in das in Kapitel 3 entworfene Standards-Modell ein. Da es sich bei den ausgewählten Standards kaum um spezifische Industriestandards handelt, wurden diese aus Kapitel 3.2 ebenfalls in die Abbildung mit einbezogen.

© HSG / IWI / CCBN / 18

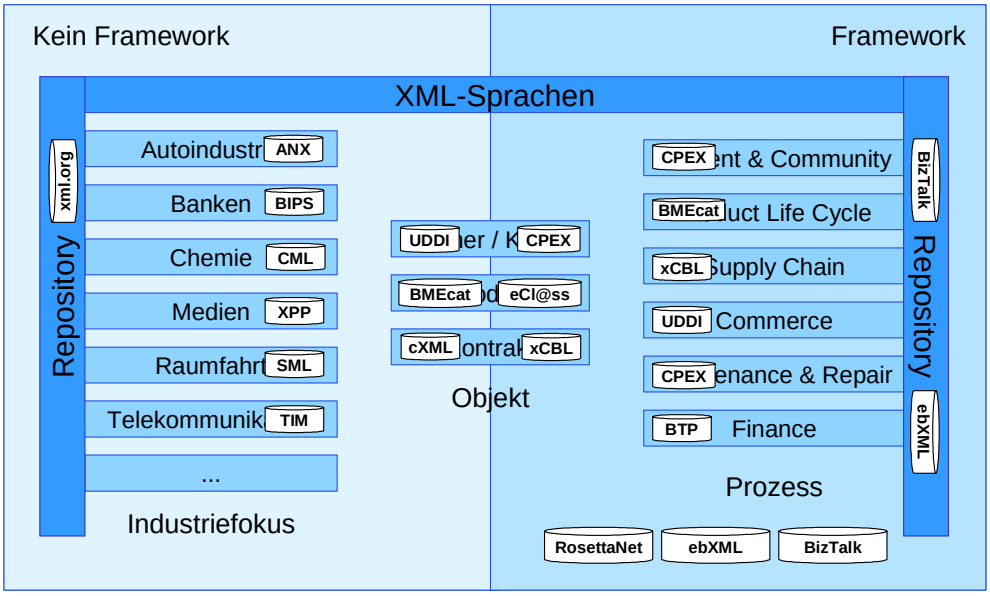


Bild 4-18: Ausgewählte B2B-Standards im Standards-Modell

Um die Darstellung nicht zu überblähen und die Übersichtlichkeit zu gewährleisten, wurden die Standards absichtlich nicht jedem berechtigtem Prozess oder Objekt zugeordnet. Sonst würde der Standard ebXML, welcher sich weder auf bestimmte Objekte noch Prozess fokussiert, die ganze Darstellung alleine in Anspruch nehmen. Eine ausführliche Tabelle zu den ausgewählten Standards ist in Anhang A zu finden.

5 Beispiel: Web Services und Business Collaboration Infrastructure in der Telekommunikationsbranche

Die im vorherigen Kapitel genannten Standards bieten Einsatzmöglichkeiten in Business Collaboration Infrastructures für elektronische Geschäftstransaktionen. Für die Realisierung von standardisierten elektronischen Kooperation sind jedoch neben Standards weitere Komponenten notwendig, die elektronische Transaktionen ermöglichen, wie das folgende Beispiel zeigt. Eine häufigere Verwendung von XML-Nachrichten treten bei Web Services auf, durch welche der Partner oder Kunde über einen Webbrowser in der Datenbank des Unternehmens bestimmte Daten abfragen oder seine Datensätze ändern kann. In diesem Kapitel wird das Projekt der Phonecom dem Leser genauer erläutert und somit die Thematik von XML-Standards verdeutlicht. Das Beispiel veranschaulicht einen typischen Kontext, in dem aus Sicht eines Unternehmens Fragestellungen um Standards auftauchen können. Die Informationen zu diesem Praxisbeispiel stammen vorwiegend aus dem Interview mit Herrn

von Ammon von Bea Systems und den Kontakten zum Telekommunikationsunternehmen (s. Anhang B).

5.1. Kundenservice der Phonecom

Die Phonecom ist ein führendes europäisches Telekommunikationsunternehmen mit Niederlassungen in über 15 verschiedenen Ländern in Australien, Nordamerika, Europa und Asien. Phonecom hat über 100'000 Mitarbeiter und macht einen jährlichen Umsatz von zirka 30 Milliarden Euro [vgl. Datapro 2001]. Um ihren Kunden – nahezu 30 Millionen Festnetz und über 10 Millionen mobil – einen besseren Kundenservice anzubieten, sollen die bestehenden isolierten Prozesse und Systeme miteinander integriert werden. Hierzu implementiert PHONECOM eine Business Collaboration Infrastructure auf Basis der ‚Bea WebLogic Integration'. Mittels eines Web Services für den Kooperationsprozess Commerce werden zuerst die Grosskunden wie beispielsweise aus der Auto- oder Chemieindustrie bedient, Kleinkunden werden erst in einem späteren Zeitpunkt hinzugefügt. Web Services zeichnen sich dadurch aus, dass sich dabei um einen Datenaustausch zwischen Applikationen handelt [vgl. Saarinen 2001]. Auf Seite des Kunden ist dies der Webbrowser und auf Seite von PHONECOM die internen Applikationen.

In diesem Beispiel wird der Prozess betrachtet, bei dem ein Grosskunde eine 2 Megabit Datenleitung für den Internetanschluss bei PHONECOM bestellt. Die Kündigung der bis anhin verwendeten Leistung wird dabei ausser Betracht gelassen. Auch die Erfassung der Kundendaten und die Informationen über die Zahlungsabwicklung werden nicht beleuchtet.

5.2. Ausgangssituation: Prozess- und Systemarchitektur

Bis anhin mussten die Kunden im Call Center der PHONECOM anrufen um beispielsweise eine neue Dienstleistung zu abonnieren oder abzubestellen. Ein Mitarbeiter hat darauf die Daten in der internen Applikation aktualisiert. Durch die Implementierung der Business Collaboration Infrastructure wird es dem Kunden in einem ersten Schritt ermöglicht, über einen Webbrowser neue Services zu bestellen oder bestehende zurückzuziehen. Zudem wird es Neukunden ermöglicht sich selbst in der Datenbank zu erfassen und somit einen neuen Service zu bestellen. Um die Implementierung der BCI bei PHONECOM zu veranschaulichen, wird in der nachfolgenden Beschreibung das oben erwähnte Prozessbeispiel verwendet. Der Verkauf bei PHONECOM ist durch den Aussendienst geregelt. Dieser besucht die

potentiellen Neukunden und die bestehenden Kunden und arbeitet mit ihnen eine Offerte aus, welche dann durch die Unterzeichnung zum Kontrakt wird. Die Daten werden dann im Bestellsystem erfasst und an die Rechnungsstelle, an den Kundendienst und an das Bestellwesen weitergeleitet. Diese Stellen erfassen die Daten nochmals in ihren Applikationen und fügen weitere relevante Zusatzdaten hinzu. Darauf gibt das Bestellwesen den Auftrag an den Netzwerkdienst, welcher die Datenleitung für den Grosskunden frei schaltet und allfällige Installationen vornimmt. Ist die Installation erfolgreich beendet, wird das Bestellwesen informiert, welches darauf die Betriebsbereitschaft der Rechnungsstelle und den Kundendienst meldet.

Durch diesen Prozess werden Daten in vier Applikationen eingegeben und verwaltet. Dies sind das Bestellsystem, das Billingsystem, das Servicesystem und das Anschlussverwaltungssystem. Das Bestellsystem wird für das Order Management eingesetzt und dient somit als Grundlage sowohl für den Verkauf als auch für das Bestellwesen. In der Rechnungsstelle wird das Billingsystem genutzt, worüber auch die ganze Buchhaltung abläuft. Für den Customer Care verwendet das Personal im Kundendienst das Servicesystem und schliesslich operiert der Netzwerkdienst mit einer weiteren eigenen Applikation.

5.3. Zielsetzung: Prozess- und Systemarchitektur

Durch die Implementierung der Business Collaboration Infrastructure soll nun die Mehrfacherfassung von Daten vermieden und die Automatisierung dieses Prozesses gefördert werden. Bild 5-2 zeigt den Prozessablauf nach der Implementierung der BCI vor.

© HSG / IWI / CCBN / 18

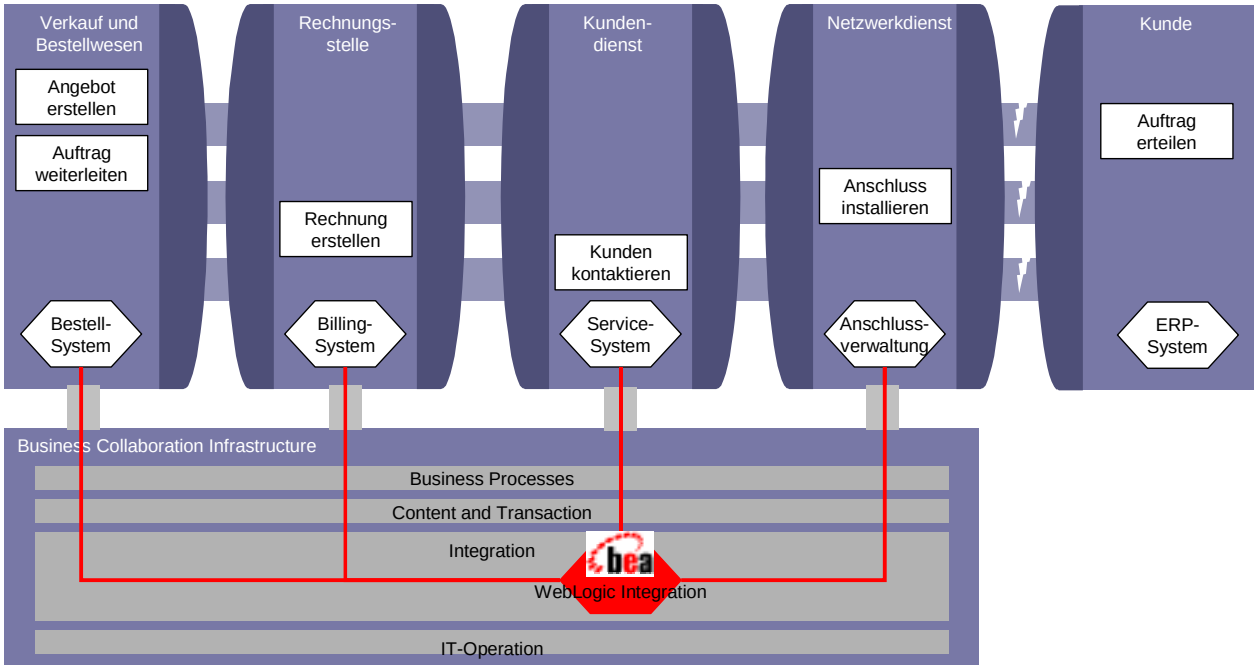


Bild 5-1: Prozess- und Systemarchitektur nach Implementierung der BCI

Nach der Implementierung der BCI bleibt die Erzeugung der Offerte der Verkaufsstelle zugeordnet. Wird die Offerte vom Kunden unterzeichnet, wird der Kontrakt geschlossen. Dies löst einen Kontrakt Event aus, welcher die erfassten Daten an das Billingsystem bei der Rechnungsstelle übermittelt. Nachfolgend werden die Zusatzinformationen wie beispielsweise die Rechnungsnummer an das Bestellsystem des Bestellwesen weitergeleitet. Sämtliche gesammelten Informationen werden darauf an das Servicesystem des Kundendiensts gesendet, dies geschieht wieder durch die Erzeugung eines Events. Sind die Daten im Servicesystem erfasst worden, wird der Event BW Info ausgelöst, wodurch die interne Bestellung erfasst wird. Nun ist die Bestellung mit sämtlichen Daten gespeichert und die auszuführende Leistung kann gestartet werden. Dies geschieht durch den Bestell-Event, welcher die Informationen an das Netzwerksystem übermittelt. Ist die Installation beendet, wird dies dem Bestellsystem mittels dem Event ‚EndInstallation' mitgeteilt. Darauf löst das Bestellsystem einen ‚RFS'-Event aus, welcher den anderen Applikationen bestätigt, dass die Leistung installiert ist und für den Kunden ab sofort zur Verfügung steht.

Wesentlich für das Gelingen des Projekts ist eine Integrationsplattform, welche den akkuraten Datenaustausch gewährleistet. PHONECOM setzt dabei auf den WLI von BEA, welcher sich besonders durch die Bus-Technologie und die standardisierten Schnittstellen auszeichnet.

5.4. Exkurs: Bea WebLogic Integration

Die Bea WebLogic Intergration (WLI) beabsichtigt die Entwicklung und den Einsatz von neuen Applikationen zu beschleunigen, die Integrationssorgen zu minimieren und die Total Cost of Ownership[23] für IT-Initiativen zu reduzieren. Die auf Standards basierende WLI ist eine einzelne Plattform, welche errichtet wurde, um die neuen Web- und Wireless- Applikationen schnell in das bestehende System und somit in die komplexen Geschäftsprozesse zu integrieren und mit anderen Geschäftspartner zu verknüpfen. WLI besteht aus vier Kernkomponenten, welche eine integrierte Plattform für Unternehmen bieten. Bild 5-3 veranschaulicht die Plattform mit ihren vier Komponenten.

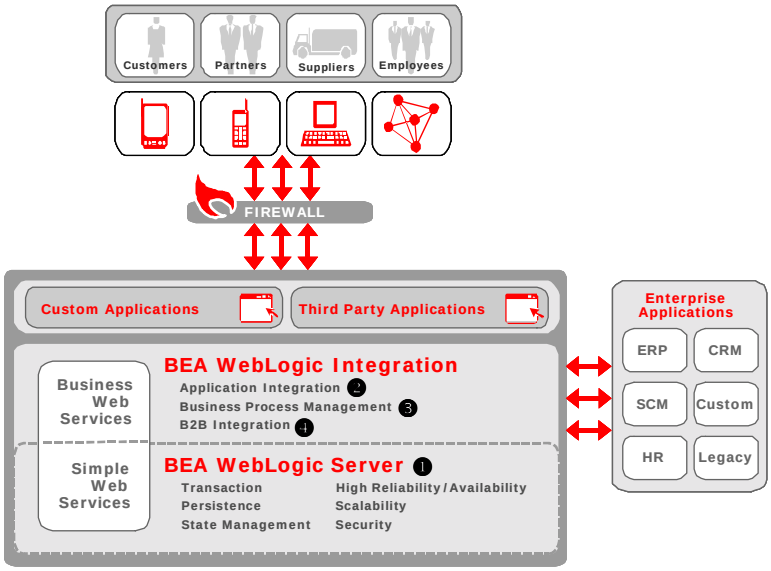


Bild 5-2: Bea WebLogic Integration ([vgl. BEA 2001] und [vgl. Saarinen 2001])

Die erste Komponente ist der Applikationsserver, welcher die Funktionalität von Web- und Wireless-Applikationen gewährleistet. Die zweite Komponente ist die Application Intergration, dessen Lösung auf der J2EE Connector Architecture (J2EE CA) basiert und somit eine rasche Datenintegration ermöglicht. Zusätzlich werden dadurch die Kosten der Integration von neuen Kundenapplikationen gesenkt. Die Funktionalität eines umfassenden Geschäftsprozess-Managements, welches die dritte Komponente ist, ermöglicht den Unternehmen komplexe e-Business-Prozesse zu entwerfen, auszuführen und kontinuierlich zu optimieren. Die letzte Komponente ist die B2B Integration, welche die Eintrittsbarrieren für weitere Unternehmen

[23] Unter dem von ihr kreierten Begriff Total Cost of Ownership versteht die Gartner Group die Gesamtkosten, die ein Datenverarbeitungssystem, insb. ein PC, während seiner Nutzungsdauer im Unternehmen verursacht. [Stahlknecht 1998]

senken soll. Zudem schafft sie eine sichere, prozessfokussierte Umgebung für das Relationship-Management und die Interaktion mit Händler, Verkäufer und Lieferanten[24].

Bea unterscheidet dabei zwei Arten von Web Services. Dies sind zum einen die Simple Web Services, welche sich auf der Ebene des WebLogic Servers zur Verfügung gestellt werden. Standards die auf dieser Ebene unterstützt werden sind beispielsweise SOAP und UDDI. Simple Web Services zeichnen sich dadurch aus, dass es sich dabei um einen Nachrichtenaustausch von einer Applikation zu einer anderen handelt. Die gesendeten Nachrichten sind dabei keine Transaktionen[25]. Auf der anderen Seite gibt es Business Web Services, welche sich nicht mehr durch eine Nachricht, sondern durch einen Workflow über mehrere Unternehmen auszeichnen. Business Web Services sind auf er Ebene der WebLogic Integration, welche Standards wie ebXML, BTP oder RosettaNet unterstützt [vgl. Saarinen 2001].

5.5. Weitere Projektschritte

Die groben Projektziele, die bei PHONECOM verfolgt werden, sind die Verbesserung der internen Prozesse, die Automatisierung derer, die Kostenreduktion und die Verkürzung der Zeitspanne bei der Einführung neuer Produkte. Ein weiteres wichtiges Ziel ist die Verbesserung des Kundenservices und somit der Aufbau einer starken Kundenbindung. Wie die einzelnen Ziele jedoch in Zahlen aussehen, ist bis dato noch nicht bekannt. Von grösster Wichtigkeit ist die Implementierung der WLI mit all seinen Schnittstellen. Die Implementierung sollte jedoch bis Mitte 2002 erfolgt sein, und erst dann können und werden die weiteren Projektschritte definiert.

5.6. Einsatzmöglichkeiten von XML-Standards

XML-Standards lassen sich überall dort verwenden, wo Daten zwischen verschiedenen Applikationen ausgetauscht werden. Im Praxisbeispiel von PHONECOM sind dies die Stellen wo die internen Applikationen Bestellsystem, Billingsystem, Servicesystem und das Anschlussverwaltungssystem miteinander kommunizieren. Dort bieten sich Standards wie beispielsweise SOAP oder BTP an. Für den Austausch von Geschäftsdokumenten mit Partner

[24] Für ausführlichere Informationen zu WLI und J2EE CA ist auf die Webseiten www.bea.com und http://edocs.bea.com verwiesen. Zudem ist es auf der erstgenannten Webseite möglich eine aktuelle Version von WLI runterzuladen und zu testen.

[25] Transaktionen zeichnen sich dadurch aus, dass sie ganz oder gar nicht durchgeführt werden, von einem konsistenten Zustand wieder in einen solchen übergehen, nicht von einer anderen Transaktion beeinflusst werden können und eine dauerhafte Wirkung auf die Daten haben [vgl. Hennekeuser/Peter 1994, S. 181].

werden jedoch Standards eingesetzt, welche sogleich die Prozesse standardisieren. Aus diesem Grund werden für den zwischenbetrieblichen Datenaustausch vorwiegend die XML-Frameworks wie ebXML, RosettaNet oder BizTalk verwendet.

Weiter ist auch eine Publizierung der angebotenen Services in der Registry von UDDI denkbar. So könnte dann ein anderes Untenehmen auf die von PHONECOM angebotenen Services aufmerksam werden. Durch beispielsweise ein cXML-Dokument kann sich der Kunde nun direkt über Produkteigenschaften und Leistungen von PHONECOM informieren, ohne dass es zu einem Telefon- oder e-Mail-Kontakt kommt [vgl. Copeland 2001]. Es finden sich aber nicht nur Einsatzmöglichkeiten der XML-Standards bei den Prozessen, wie jetzt beim Prozess Commerce gezeigt, auch Industrien oder Objekte können unterstützt werden. So kann beispielsweise PHONECOM mit Wettbewerbern eine Kollaboration eingehen, da beide von einer hohen Fluktuation im Handymarkt betroffen sind. Mittels CPExchange können sie somit Informationen über ihre Kunden austauschen und dadurch die Abwanderung mindern.

Wie stark man bei PHONECOM auf XML setzen wird, ist noch unklar, da man bei der Entwicklung der Adapter eine möglichst kostengünstige Variante sucht. Bea verwendet bei der Entwicklung von Adaptern den J2EE CA Standard, welcher auf Java basiert. Mit hoher Wahrscheinlichkeit wird jedoch XML im Datenaustausch zwischen dem Browser und der Schnittstelle am WLI die dominante Stellung einnehmen. Welche Standards für die einzelnen Prozesse verwendet werden, ist noch nicht bestimmt worden, aber eine Verwendung von cXML und SOAP ist denkbar, da diese Standards bereits von WLI unterstütz werden [vgl. Saarinen 2001].

Wie in diesem kurzen Einblick in ein Praxisbeispiel veranschaulicht wurde, spielt die Wahl eines XML-Standards derzeit nur eine geringe Rolle. Für verschiedene zukünftige Szenarien sind die untersuchten Standards eine wichtige, realistische Option. Die Verwendung und Auswahl von Standards sind jedoch nur eine Teil-Fragestellung beim Aufbau einer Lösung zum elektronischen Austausch von Geschäftsdokumenten. Dies wird im nächsten Kapitel nochmals verdeutlicht, denn von den Fragen die das Management stellt, beziehen sich aktuell nur wenige auf die Wahl eines Standards.

© HSG / IWI / CCBN / 18

6 Fragen des Managements

Beim Aufbau einer Lösung für den elektronischen Austausch von Geschäftsdokumenten sind mehrere Aspekte zu berücksichtigen. So stellt sich das Management Fragen zum Prozess, zu den Objekten und der Branche, wie sie bei der Entwicklung des Standards-Modells eine wichtige Rolle eingenommen haben. Weiter werden bei der Implementierung einer Business Collaboration Infrastructure eine Menge an technischen Fragen gestellt, welche im zweiten Teil dieses Kapitels betrachtet werden.

6.1. Strategische Fragen

Das Management muss sich in einem ersten Schritt Gedanken über die strategische Ausrichtung ihrer Unternehmung machen. So muss definiert werden, welcher Prozess in Zukunft mit einer Business Collaboration Infrastructure abgedeckt werden soll und welche Objekte einen Einfluss auf diesen Prozess haben. Findet beispielweise eine Reorganisation des Bestellprozesses statt, sind darin die Partner, die Kunden und der Kontrakt enthalten. Zudem muss sich das Management fragen, welche Kunden in Zukunft mit den Web Services bedient werden sollen. Dies kann einerseits ein besserer Service für bestehende Kunden, andererseits aber auch die Gewinnung von Neukunden sein. Schliesslich ist auch ein Fokus auf eine Industriebranche eine weitere strategische Entscheidung.

Für die Neudefinierung des Bestellprozesses, welcher im Kooperationsprozess Commerce zugeordnet ist, kann das Management mit Hilfe der in Kapitel 3.6 entwickelten XML-Standards-Datenbank nun nach einem XML-Framework suchen. Dabei generiert die Datenbank die Ausgaben ebXML, RosettaNet und BizTalk. Nun muss sich das Management mit den standardisierten Bestellprozessen der drei Organisationen auseinander setzen. Ein zusätzlicher wichtiger Entscheidungsgrund liegt bei der Überlebensfähigkeit des Standards. So wird dabei ebXML die grösste Wahrscheinlichkeit zugesprochen [vgl. Abrams 2001b].

Damit sich das Management nicht mit den Dokumentinhalten auseinandersetzen muss, stützt es sich am besten auf einen bereits definierten Standard. Um diesen nun ausfindig zu machen ist ebenfalls eine Suche in der XML-Standards-Datenbank möglich. Ein möglicher XML-Standard für den Bestellprozess wäre beispielsweise xCBL.

XML-Standards können sowohl auf strategischer wie auch auf operativer Ebene Einfluss auf den Geschäftsbetrieb nehmen. Deshalb ist es für das Management wichtig, dass es sich für den

XML-Standard entscheidet, welcher das unternehmensspezifische Problem am besten lösen kann. Die in diesem Arbeitsbericht entwickelte XML-Standards-Datenbank kann dem Management dazu als Entscheidungsgrundlage dienen. Entscheidet sich eine Unternehmung für die Implementierung einer Business Collaboration Infrastructure, muss sie sich jedoch noch mit zahlreichen weiteren Fragen, welche vorwiegend technischer Herkunft sind, beschäftigen. Auf diese Fragen wird nachfolgend eingegangen.

6.2. Fragen an Software-Anbieter

Bei der Implementierung einer Business Collaboration Infrastucture werden vom Management grundsätzlich Fragen an einen Software-Anbieter zu drei Kategorien gestellt. Die drei Kategorien sind allgemeine Fragen zur Unternehmung, welche die Leistung anbietet, die Fragen über das Produkt an sich und die Fragen über die entstehenden Kosten. Am meisten Ermittlungen werden jedoch zum Produkt und dessen Integration gemacht. Die Informationen in diesem Kapitel stammen aus dem Interview mit Herrn von Ammon (Anhang BS) und dem dadurch erhaltenen Request for Information (RFI) eines Automobilherstellers.

6.2.1 Fragen zur Unternehmung

In einem ersten Schritt werden die Basisdaten eines Unternehmen ermittelt. Diese enthalten die Kontaktinformationen, die Anzahl der Mitarbeiter, Umsatzahlen und Informationen über die Beteiligung an Standardisierungsgremien. In einem nächsten Schritt ist es für das Management von Relevanz, in welchen Allianzen das Unternehmen mitwirkt. Dadurch kann festgestellt werden, in welchem Umfeld sich das Unternehmen befindet. In einem letzten Schritt werden Fragen über die angebotene Unterstützung gestellt. Dabei geht es nicht nur um den Beistand bei der Entwicklung des Projekts, sondern auch um den Support bei der Integration. Grundsätzlich wird bei der Entwicklung eines Projektes ein allgemeiner Support angeboten, welcher aus interner Schulung und Beratung besteht. Bei der Integration eines Produktes werden jedoch meist externe Systemintegratoren wie beispielsweise IMG, PWC oder KPMG beigezogen.

6.2.2 Fragen zum Produkt

Ein standardisierter Fragekatalog, welcher vom Management mehrerer Unternehmen verwendet werden kann, eignet sich vielleicht bei den Fragen zur Unternehmung, jedoch nicht

bei den Fragen zum Produkt. Diese sind spezifisch zum bereits verwendeten System einer Unternehmung zu stellen und ihnen wird die Hauptfunktion bei der Entscheidung des Managements zugemessen. Analog zu den Unternehmensinformationen werden auch beim Produkt Basisdaten gefordert. Dies sind Informationen über den Namen und die Version des Produktes, den Zeitpunkt des Markteintritts, die unterschiedlichen Sprachversionen und die angebotenen Schulungen. Zudem werden Fragen über die benötigten Mitarbeiter bei der Implementierung und deren Skills gestellt. Ein letzter wichtiger Punkt sind die Reverenzen. Sie geben einer Unternehmung Aufschluss über Erfahrung und Professionalität des Softwareunternehmens. Weiter werden nun die entscheidungsrelevanten Fragen gestellt. Tabelle 6-1 stellt neun Unterpunkte vor, die bei der Erstellung eines Fragekatalogs in Betracht gezogen werden sollten, und zeigt einige Aspekte auf, die in den einzelnen Unterkategorien geklärt werden sollten.

© HSG / IWI / CCBN / 18

Untergruppe	Aspekte
Architektur	Wie sieht die Systemarchitektur aus? Welche Teilsysteme sind darin enthalten? Erklärung? Welche Adapter werden benötigt? Bereits vorhanden? Wo sind die Grenzen der Skalierung?
Repository	Existiert ein Repository? Welche Datenarten sind enthalten? Gibt es Metadaten? Können Dokumente zwischen Repositories ausgetauscht werden? Was ist dazu nötig?
Anbindung	Welche Protokolle werden unterstützt? Wie sieht das Transportverfahren aus? Ist ein synchroner Transport möglich? Wird ein File-Transfer unterstützt? Wie? Wird Messaging unterstützt? Wie?
Sicherheit	Findet eine Identifikation und Authentisierung statt? Verfahren? Wie sieht die Einbindung von Firewalls aus? Werden Dateninhalte kryptografisch verschlüsselt?
Schnittstellen auf Datenebene	Wie sind die generellen Adaptereigenschaften? Welche Adapter sind verfügbar? Welche Nachrichten Standards werden unterstützt? Wie sehen die Adapter für Standardanwendungen (z.B. SAP) aus? Welche müssen neu entwickelt werden? Wie sieht Entwicklung aus?
Transformation	Welche Data Mapping Tools sind vorhanden? Eigenschaften? Wird XML verwendet? Welche Standards werden unterstützt? Können Informationen automatisch geparst werden? Welche Funktionen werden unterstützt? Gibt es Einschränkungen bei der Definition eigener Funktionen?
Prozess-management	Welche Tools zur Prozessmodellierung sind vorhanden? Können auch andere verwendet werden? Gibt es ein Metadatenmodell? Spezifikation? Was sind die Eigenschaften des Workflows? Wie sehen Benutzeroberflächen solcher Tools aus?
Entwicklung	Wie sieht die Modellierung aus? Ist eine Trennung von Entwicklung-, Test- und Produktionsumgebung möglich? Wie sieht die Entwicklungsumgebung aus? Welche Tools werden bei der Entwicklung benötigt?
Betrieb	Welche Betriebssysteme werden benötigt? Welche Datenbankserver werden unterstützt? Welche weitere Infrastruktur wird benötigt? Wie erfolgt die Inbetriebnahme? Wie wird das Systemmanagement unterstützt?

Tabelle 6-1: Aspekte bei Produktfragen

6.2.3 Fragen zu den Kosten

Die dritte und letzte Kategorie beschäftigt sich mit sämtlichen Kosten, die bei einer Implementierung und Instandhaltung einer Business Collaboration Infrastructure anfallen. Neben den allgemeinen Fragen zur Kostenberechung der Lizenzen lassen sich grundsätzlich diese Fragen wieder in drei Unterkategorien aufteilen. Die erste Unterkategorie enthält sämtliche Fragen die sich mit den Einmalkosten beschäftigen. Dies sind Informationsbeschaffungen über die Kosten beispielsweise eines einzelnen Adapters oder einer bestimmten benötigten Hardware. Die zweite Unterkategorie beschäftigt sich mit den Fragen über die Projektkosten. Da diese meist sehr schwer zu berechnen sind, wird beispielsweise Auskunft über die Stunden- oder Tagessätze von Beratern oder Programmierer eingeholt. Zudem werden Informationen über die Kosten einer Adapteranpassung oder –entwicklung verlangt. Die letzte Unterkategorie beschäftigt sich mit den sogenannten Runtime-Kosten, welche für das Unternehmen nach der Implementierung anfallen. Sie enthält Fragen über die entstehenden Kosten bei einem Wartungsvertrag, einem Releasewechsel oder beim Upgrade eines Adapters. Es entsteht der Eindruck, dass auch die Lizenzkosten in die Unterkategorie der Runtime-Kosten einbezogen werden müssen. Einige Softwareunternehmen verrechnen jedoch die Lizenzgebühr direkt in die Entwicklungskosten, wobei es sich dann um eine einmalige Lizenzgebühr handelt. Aus diesem Grunde werden Fragen über die Kostenberechung der Lizenzen am Anfang und über alle Unterkategorien hinweg gestellt.

6.3. Fazit

Entscheidet sich ein Unternehmen für die Implementierung einer Business Collaboration Infrastructure, muss sich das Management mit zahlreichen Frage auseinander setzen. Auf strategischer Ebene sind dies Fragen, welche teilweise mit XML-Standards definiert oder beantwortet werden können. Die in diesem Arbeitsbericht entwickelte XML-Standards-Datenbank kann dazu als Einscheidungsgrundlage dienen. Aktuell beschäftigt sich das Management jedoch meist mit technischen Fragen, welche Antworten zur technischen Realisierbarkeit der ganzen BCI hervorrufen.

7 Zusammenfassung und Ausblick

„... When you design a schema, do not try to solve the problems of every conceivable user of the vocabulary: focus on the one who asked you to do it. Scope it to

their precise requirements as best as you and they can define them. Spend a lot more time on defining these before you write a line of XML."

(Len Bullard, Intergraph, 2000)

7.1. Zusammenfassung

Seit der Veröffentlichung der Metasprache XML durch das W3C vor etwas mehr als drei Jahren haben sich zahlreich Softwareunternehmen in verschiedenen Konsortien zusammengeschlossen und mit der Entwicklung neuer auf XML basierenden Standards begonnen. Dutzende dieser DeFacto-Standards werden momentan veröffentlicht, obwohl sie sich meist noch in der Entwicklung oder Weiterentwicklung befinden. Einige Organisationen setzen sich zudem mit der Architektur eines bestimmten Geschäftsprozesses auseinander und publizieren somit ihren Prozessstandard. Durch die immense Datenflut ist es für einen Entscheidungsträger kaum mehr überblickbar, welcher Standard sein unternehmensspezifisches Problem am besten abdeckt. Aus diesem Grunde wurde ein Modell mit vier Dimensionen entwickelt, welches einen Überblick über die verschiedenen Standardisierungsvorhaben bietet. Anhand dieses Modells wurde die sogenannte XML-Standards-Datenbank entwickeln, in welcher die Entscheidungsträger den geeigneten Standard selektieren und finden können.

Um die Unterschiede zwischen den Standards aufzuzeigen, wurden zwölf zur Zeit aktuelle Standards, ihre Ziele und ihre Zukunftsaussichten aufgezeigt und zur Verdeutlichung ins Standards-Modell eingeordnet. Da sich diese Standards noch in der Weiterentwicklung befinden, ist es jedoch notwendig, dass die Datenbank permanent aktualisiert wird. Denn nur somit kann eine geeignete Entscheidungsgrundlage geschaffen werden.

Anhand eines Praxisbeispiels der PHONECOM wurde beleuchtet wo und wie ein XML-Standard seine Verwendung findet. Aus diesem Beispiel lässt sich jedoch auch ableiten, dass die Wahl eines Standards nur einen geringen Anteil bei der Implementierung einer Business Collaboration Infrastructure einnimmt. Wichtiger sind die Fragen der Integrität der Enterprise Application Integration (EAI) und der entstehenden Kosten. Eine kollaborative Plattform wird nur dann eingesetzt, wenn sie für die partizipierenden Partner einen Zusatznutzen erbringt und dies sich dabei mit den Kosten im Verhältnis hält. Wohin sich die Evolution der Standards begibt, ist noch ziemlich unklar, jedoch ergeben sich einige Szenarios die in Zukunft eintreten können. Diese werden im nächsten Kapitel kurz dargelegt und basieren auf Studien der

© HSG / IWI / CCBN / 18

Gartner Group und auf Informationen der Kontaktpersonen aus der IT-Branche. Ein Verzeichnis mit den Kontaktpersonen ist in Anhang B zu finden.

7.2. Ausblick

Die meisten Unternehmen, welche bis jetzt eine konventionelle EDI-Lösung für den zwischenbetrieblichen Datenaustausch verwendeten, sind nun an der Entwicklung, ihre Daten über das Internet zu versenden. Wie gross der Zuwachs von Neukunden und insbesondere den KMUs ist, wurde bis anhin von keiner Organisation oder Unternehmung erforscht und publiziert.

Heutzutage wird versucht mittels XML zahlreiche verschiedene Definitionen durchzusetzen. XML besitzt jedoch auch Nachteile, welche nur selten erwähnt werden. So macht aus technischer Sicht der Einsatz von XML als einheitliches Datenaustauschformat nicht immer Sinn. Die Daten werden durch die Repetition von Tags relativ gross und als einziger String übermittelt, was zu enormen Performanceeinbussen führen kann. Vorteil hingegen ist eine relativ einfache Veränderbarkeit oder Lesbarkeit wie beispielsweise zur Überwachung. Auch die Skalierung ist durch XML-Standards einfacher zu realisieren.

Die Akzeptanz in der betrieblichen Verwendung wird über die Weiterentwicklung der vorgestellten Standardisierungsvorhaben entscheiden. Die Gartner Group stellt dazu drei mögliche Szenarien vor, wie die Verwendung von XML-Standards im Jahre 2006 aussehen kann [vgl. Abrams 2001a]. Das erste Szenario geht davon aus, dass zwei unabhängige XML-Sprachen, welches xCBL und cXML sein werden, die Welt der XML-Sprachen beherrschen und auf Serverebene gegenseitig ausgetauscht werden können. Die in diesen Sprachen verfassten Dokument werden in einem einzelnen Web-basierten Repository vereinigt und über Metadaten klar definiert. Über die Registry von ebXML wird es den Unternehmen ermöglicht die geeigneten Partner für einen bestimmten Service zu finden. Das zweite Szenario geht davon aus, dass eine Unternehmung im Jahr 2006 mindestens 75 XML-Standards in ihre Business Collaboration Infrastructure integrieren muss und zusätzlich nicht weniger als zwölf Framework-Standards adoptiert werden sollten, um mit potentiellen Partnern über eine Registry zu kommunizieren. Zusätzlich werden aus diesem Grund viele Unternehmen eigene Lösungen mit ihren engsten Handelspartnern entwickeln, welche dann nicht in der Registry abgelegt werden. Das letzte Szenario geht davon aus, dass ebXML mit seinem ganzen Konstrukt zur grundlegenden Sprache für die Interoperabilität des EC wird. Unabhängige

© HSG / IWI / CCBN / 18

internationale Organisationen kontrollieren die Prozesse, welche durch ebXML abgedeckt werden. Durch die kontinuierliche Weiterentwicklung hat sich ebXML zu einer kompletten technischen Infrastruktur für das EC entwickelt.

In welche Richtung wir uns bewegen werden, wird uns die Zukunft zeigen. Auf jeden Fall wird es notwendig werden, die verschiedenen Standards miteinander zu verbinden, damit die Übersichtlichkeit nicht definitiv verloren geht. So ist beispielsweise die Verknüpfung der beiden Repositories von ebXML und UDDI nur noch eine Frage der Zeit. Schliesslich soll eine Unternehmung nur in einer Datenbank suchen müssen, um einen geeigneten Handelspartner zu finden.

© HSG / IWI / CCBN / 18

Anhang A: B2B-Standards im Modell

		ebXML	RosettaNet	BizTalk	SOAP	BTP	xCBL	cXML	UDDI	DUNS	WSDL	eCl@ss	BMEcat	CPExchange
Art	Sprache	☑	☑	-	☑	☑	☑	☑	☑	☑	☑	☑	☑	☑
	Repository	☑	☑	☑	-	☑	☑	-	-	-	-	-	-	-
	Framework	☑	☑	☑	-	☑	-	-	-	-	-	-	-	-
Industriefokus	Banken	☑	-	-	-	-	-	-	-	-	-	-	-	-
	IT	-	☑	-	-	-	-	-	-	-	-	-	-	-
	Transport & Logistik	☑	-	-	-	-	-	-	-	-	-	-	-	-
	Versicherungen	☑	-	-	-	-	-	-	-	-	-	-	-	-
	Kein Branchenfokus	☑	☑	☑	☑	☑	☑	☑	☑	☑	☑	☑	☑	☑
Objekt	Partner / Kunde	☑	☑	☑	☑	-	-	-	☑	☑	☑	-	-	☑
	Produkt	☑	☑	☑	☑	-	☑	☑	-	-	-	☑	☑	-
	Kontrakt	☑	☑	☑	☑	☑	☑	☑	-	-	-	-	-	-
Prozess	Content & Community	☑	☑	☑	☑	-	-	-	☑	☑	☑	-	-	☑
	Product Life Cycle	☑	☑	☑	☑	-	-	-	-	-	☑	☑	☑	-
	Supply Chain	☑	☑	☑	☑	☑	☑	☑	-	-	☑	☑	☑	-
	Commerce	☑	☑	☑	☑	☑	☑	☑	☑	☑	☑	☑	☑	☑
	Maintenance & Repair	☑	☑	☑	☑	-	-	-	-	-	☑	-	-	☑
	Finance	☑	☑	☑	☑	☑	☑	☑	-	-	☑	-	-	-

Tabelle A-0-1: Einordnung der B2B-Standards ins Standard-Modell

© HSG / IWI / CCBN / 18

Anhang B: Kontaktpersonen aus der IT-Branche

Hauptkontaktperson	
Dr. Rainer von Ammon BEA Systems GmbH Max-Planck-Str. 5 85609 Aschheim-Dornach Deutschland www.bea.com	Interview bezüglich XML-Standards, Fragen des Managements und Praxisbeispiel mit anschliessender Diskussion vor Ort: Datum: 29. 8. 2001, Zeit: 14.00 – 17.15 Uhr zusätzlich Telefon- und e-Mail-Kontakte

Kontaktperson	
(Name dem Autor bekannt) PHONECOM GmbH (Adresse dem Autor bekannt)	Telefon- und e-Mail-Kontakte bezüglich Praxis-beispiel bei PHONECOM

Kontaktperson	
Martin Koch boost GmbH Curierstr. 4 70563 Stuttgart Deutschland www.boo.st	Interview mit anschliessender Diskussion und Ein-führung in WLI vor Ort. Datum: 18. 9. 2001, Zeit: 14.15 – 17.45 Uhr Telefon- und e-Mail-Kontakte bezüglich WLI

Kontaktperson	
Daniel Hümbeli Principal Consultant PricewaterhouseCoopers Affolternstr. 52 8050 Zürich www.pwc.ch	Telefon- und e-Mail-Kontakte bezüglich Klassifizierung und Zukunftsaussichten für XML-Standards

Literatur

[Abrams 2001a]
Abrams, C., Collaborative Marketplace Standards Scenarios for 2006, Gartner Group, Dokument Nr. SPA-13-1127, 2001a.

[Abrams 2001b]
Abrams, C., XML Builds the Collaborative Marketplace, Gartner Group, Dokument Nr. DF-13-0461, 2001b.

[Alt et al. 2001]
Alt, R., Leser, F., Puschmann, T., Reichmayr, C., Business Networking Architektur, Version 0.8, Institut für Wirtschaftsinformatik, St. Gallen, 2001.

[Ariba 2000]
Ariba, cXML Benutzerhandbuch, Version 1.1, April 2000, cXML.org, 2000.

[Ballnus 2000]
Ballnus, R., Erfolg mit EDI und e-Commerce, Tectum Verlag, Marburg, 2000.

[BEA 2001]
BEA, BEA WebLogic Integration, BEA Systems, Inc., San Jose, 2001.

[Bleich 2000]
Bleich, H., Austausch von Kundenprofilen für den e-Commerce, Verlag Heinz Heise, http://www.heise.de/newsticker/data/hob-08.12.00-000/, 10.9.2001.

[BMEcat 2001]
BMEcat, Neue Version des e-Commerce-Standards BMEcat, bmecat.org, http://www.bmecat.de, 25.7.2001.

[Bohrer/Holland 2000]
Bohrer, K., Holland, B., Customer Profile Exchange (CPExchange) Specification, 20. October 2000, Version 1.0, Internationl Digital Enterprise Alliance, 2000.

[Borchers 2001]
Borchers, D., UDDI: Neue Standards braucht das Web, ZDNet, http://www.zdnet.de/kolumne/200101/mdb81_00-wc.html, 3.9.2001.

[Bussler 2001]
Bussler, C., B2B Protocol Standards and their Role in Semantic B2B Integration Engines, in: Data Engineering, 24 (2001) 1, S. 3-11.

[Cecere et al. 2001]
Cecere, L., Hope-Ross, D., Peterson, K., Spencer, C., Reilly, G., Commerce One: A Visionary in Uncertain Times, Gartner Group, Dokument Nr. C-13-8786, 2001.

[CommerceOne 2000]
CommerceOne, XML Common Business Library xCBL - XML Interconnectivity Guide, Version 2.0.1, Commerce One, Pleasanton, CA, 2000.

[Convergence 2001]
Convergence, Warum offene Standards?, Convergence, http://www.linuxtv.net/convergence/openstandards.xml, 8.8.2001.

[Copeland 2001]
Copeland, M.L., Has UDDI created a competitive edge?, Web Services Architect, http://www.webservicesarchitect.com/content/articles/copeland01.asp, 10.10.2001.

[Cover/OASIS 2001]
Cover, R., OASIS, Bea Presents Proposed Business Transaction Protocol Version 1.0 to OASIS TC, The XML Cover Pages, http://xml.coverpages.org/ni2001-03-08-b.html, 24.9.2001.

[D&B 1998]
D&B, Erläuterungen und Suchhinweise für Dun & Bradstreet Worldbase, Dun & Bradstreet, 1998.

[D&B 2000]
D&B, Was ist D&B D-U-N-S Nummer?, Dun & Bradstreet Deutschland, http://www.dbgermany.com/dunsno/dunsno.htm, 10.9.2001.

[Dalal/Takacsi-Nagy 2001]
Dalal, S., Takacsi-Nagy, P., Proposal for Transaction Protocol - Version 1.0, Bea Systems, Inc., 2001.

[Datapro 2001]
Datapro, Global Networking Services Vendors: Comparison Columns, Garnter Group, 2001.

[Donath 2001]
Donath, A., Microsoft: BizTalk Server Accelerator for RosettaNet, golem.de, http://www.itnews.de/0105/13739.html, 26.7.2001.

[ECIN 2001]
ECIN, UDDI ist online, ECIN, http://www.ecin.de/news/2001/05/03/02022.htm, 3.9.2001.

[EDS 2000]
EDS, XML Standards Profiles - Prepared for the United States Postal Services, Electronic Data Systems Corporation, 2000.

[Ehrhardt 2001]
Ehrhardt, D., BEA Leads New OASIS Technical Committee to Develop Open Industry Standards for Business Transaction Management, Bea Systems, Inc., http://www.bea.com/press/releases/2001/0131_OASIS_Tech_Committee.shtml, 24.9.2001.

[Encarta 2000a]
Encarta, ISBN, in: Microsoft Encarta Enzyklopädie 2000, Microsoft Corporation, 2000a.

[Encarta 2000b]
Encarta, Standard, in: Microsoft Encarta Enzyklopädie 2000, Microsoft Corporation, 2000b.

[Felden 2000]
Felden, C., WI-Seminar WS 2000/2001 - Ausgewählte Themen des Electronic Commerce, Universität Duisburg, 2000.

[Fitzgerald 2001]
Fitzgerald, M., Building B2B Applications with XML - A Resource Guide, John Wiley & Sons, Inc., New York, 2001.

[FIZ-Technik 1998]
FIZ-Technik, Datenbankbeschreibung Dun & Bradstreet, FIZ Technik im Branchenmarkt Maschinenbau, http://www.maschinenbau-info.de/technologie/db/m_duns.htm, 10.9.2001.

[FIZ-Technik 2001]
FIZ-Technik, Firmendatenbanken weltweit, Fachinformationszentrum Technik e.V., http://www.fiz-technik.de/recherche/recherchieren_dun.htm, 19.7.2001.

[Fleisch 2001]
Fleisch, E., Das Netzwerkunternehmen, Springer-Verlag, Berlin, 2001.

[Fleisch/Österle 2001]
Fleisch, E., Österle, H., Vom elektronischen Schaufenster zum Prozessportal, in: io management, 70 (2001) 4, S. 18-24.

[Frank 2000]
Frank, U., Vergleichende Betrachtung von Standardisierungsvorhaben zur Realisierung von Infrastrukturen für das e-Business, Institut für Wirtschaftsinformatik, Universität Koblenz-Landau, Koblenz, 2000.

[Frank 2001]
Frank, U., Standardisierungsvorhaben zur Unterstützung des elektronischen Handels: Überblick über anwendungsnahe Ansätze, in: Wirtschaftsinformatik, 43 (2001) Nr. 3, S. 283-293.

[Giehl 2001]
Giehl, A., ProSa: WebServices, Computer Sciencs Department, Universität Hamburg, 2001

[Hennekeuser/Peter 1994]
Hennekeuser, J., Peter, G., Rechner-Kommunikation für Anwender, Springer-Verlag, Berlin, 1994.

[Hess 2001]
Hess, D., SOAP, UDDI, and WSDL: An Introduction, Gartner Group, 2001.

[Huemer 2000]
Huemer, C., B2B-Systeme im E-Business, WS 2000/01 - Teil 5, Fakultät für Wirtschaftswissenschaften und Informatik, Universität Klagenfurt, 2000.

[Huemer 2001]
Huemer, C., Electronic Business XML - Grundlagen und Nutzen, in: XML in der betrieblichen Praxis, dpunkt.verlag, Heidelberg, 2001, S. 13-28.

[Hümpel/Schmitz 2001]
Hümpel, C., Schmitz, V., BMEcat - Produktkatalogeaustausch per XML, in: XML in der betrieblichen Praxis, dpunkt.verlag, Heidelberg, 2001, S. 205-217.

[Ihlenfeld 2001]
Ihlenfeld, J., Standard zum Austausch von Online-Kundendaten, Golem Network News, http://www.golem.de/0006/8202.html, 10.9.2001.

[Hoyer 2001]
Hoyer, S., ProSa: Higher Knowledge, eService Metadata, Computer Sciencs Department, Universität Hamburg, 2001

[ISO 2001a]
ISO, What are standards?, International Organisastion for Standardization, http:\\www.iso.ch/iso/en/aboutiso/introduction/index.html, 8.8.2001.

[ISO 2001b]
ISO, Why is international standardization needed?, International Organization for Standardization, http://www.iso.ch/iso/en/aboutiso/introduction/whyneeded.html, 8.8.2001.

[IW-Köln 2000]
IW-Köln, ecl@ss: Standard für Materialklassifikation und Warengruppen - Anwendung im e-Commerce, Institut der deutschen Wirtschaft Köln Consult GmbH, 2000.

[jCatalog 2001]
jCatalog, eCl@ss - Standard für Materialklassifikation und Warengruppen, jCatalog Software AG, http://ww.jcatalog.de/products/buyfast/eclass.htm, 25.7.2001.

[Knox/Hess 2001]
Knox, R., Hess, D., XML: The Inner Workings to Future E-Commerce Success, Gartner Group, Dokument Nr. TU-12-9528, 2001.

[Leser 2001]
Leser, F., Elektronsiche Bestellung in EDI und XML, Institut für Wirtschaftsinformatik, St. Gallen, Dokument Nr. BE HSG/CCBN/5, 2001.

[Logan 2000]
Logan, D., FAQ: Summary of XML Standards Bodies, Gartner Group, Dokument Nr. QA-11-6294, 2000.

[Manes 2001]
Manes, A.T., ebXML Architecture, in: Proceedings of the O'Reilly Conference on Java, Sun Microsystems, 2001.

[Marwick-Ebner 2001]
Marwick-Ebner, A.v., Eine Nummer für alle Fälle, Aktiv Wirtschaftszeitung, http://www.div-aktiv.de/aktiv05-01/aktiv-5.htm, 10.9.2001.

[McGrath 1998]

McGrath, S., XML by Example, Prentice Hall PTR, Upper Saddle River, USA, 1998.

[McKee et al. 2001]
McKee, B., Ehnebuske, D., Rogers, D., UDDI Version 2.0 API Specification - Open Draft, UDDI.org, 2001.

[Microsoft 2000]
Microsoft, BizTalk Framework 2.0: Document and Message Specification, Microsoft Corporation, 2000.

[Möhr/Schmidt 1999]
Möhr, W., Schmidt, I., SGML und XML - Anwendungen und Perspektiven, Springer-Verlag, Heidelberg, 1999.

[Natis/Driver 2001]
Natis, Y., Driver, M., Java JAX: Java Platform Takes a Step to Web Services, Gartner Group, Dokument Nr. E-12-7761, 2001.

[Österle 2000]
Österle, H., Neue Geschäftsmodelle im Zeitalter der Digitalen Wirtschaft, http://www.gitverlag.de/pit/images/022000/12neuemodelle.html, 17.7.2001.

[Österle 2001]
Österle, H., Geschäftsmodell des Informationszeitalters, in: Business Networking in der Praxis, im Druck, Springer-Verlag, Berlin, 2001, S. 17-37.

[Pastoors et al. 2001]
Pastoors, T., Kelkar, O., Schmitz, V., BMEcat V1.2 für Einsteiger, Universität Essen BLI, Fraunhofer IAO, Stuttgart, 2001.

[Plummer 2001]
Plummer, D., UDDI Goes Live: Is Anybody Watching?, Gartner Group, Dokument Nr. E-13-9658, 2001.

[Reimers 1995]
Reimers, K., Normungsprozesse - Eine transaktionskostentheoretische Analyse, Betriebswirtschaftlicher Verlag Dr. Th. Gabler GmbH, Wiesbaden, 1995.

[Reinwarth 2001]
Reinwarth, M., UDDI und B2B, Connector Gesellschaft für Kommunikation und Bera-tung mbH, http://www.connector.de/publikationen/newsflash/2001_04/newsflash_0104.htm, 3.9.2001.

[Rishel 2001]
Rishel, W., RosettaNet: A Refreshing Approach to Standards, Gartner Group, Dokument Nr. TU-13-6990, 2001.

[RosettaNet 2001a]
RosettaNet, Background and Information, RosettaNet, Santa Ana, California, 2001a.

[RosettaNet 2001b]
RosettaNet, Welcome to RosettaNet: RosettaNet Overview, http://www.rosettanet.org, 27.8.2001.

[Saarinen 2001]
Saarinen, M., Web Services: Building Services-Based Architecture, Bea Systems Inc., San Jose, 2001.

[Sanmann 2001]
Sanmann, A., ProSa: ebXML - Der neue Standard im E-Business?, Computer Sciencs Department, Universität Hamburg, 2001.

[Schmidt 2001]
Schmidt, S., Sevice wichitger als Datenschutz im Internet, werbeagentur.de, http://www.werbeagentur.de/magazin/news, 10.9.2001.

[Schmitz/Kelkar 2001]
Schmitz, V., Kelkar, O., BMEcat Version 1.2 - Der Produktkatalogstandard für das e-Business, Universität Essen bli, Fraunhofer IAO, Stuttgart, 2001.

[Schmoll/Nommensen 1996]
Schmoll, T., Nommensen, O., EDI - Wettbewerbsvorteile durch Electronic Business, Markt&Technik Buch- und Software Verlag GmbH, Haar bei München, 1996.

[Sleeper 2001]
Sleeper, B., Why UDDI Will Succeed, Quietly: Two Factors Push Web Services Forward, The Stencil Group, www.stencilgroup.com/ideas_scope_200104uddi.html, 27.7.2001.

[Stahlknecht 1998]
Stahlknecht, P., Total Cost of Ownership, Lexikon der Wirtschaftsinformatik, http://www.wi1.uni-erlangen.de/buecher/lexikon/tco.html, 2.10.2001.

[Steffen 2001]
Steffen, T., XML/EDI Standardisierung: Ein Überblick, in: XML in der betrieblichen Praxis, dpunkt.verlag, Heidelberg, 2001, S. 1-12.

[Steiner 1996]
Steiner, P., Semiotik, Semantik, Pragmatik - eine Einführung, Arbeitsbereich Linguistik, Westfälische Wilhelms-Universität, Münster, 1996.

[Storz 2001]
Storz, C., Institutioneller Wandel und die Globalisierung von Standards, Universität Marburg, http://www.uni-marburg.de/japanz/projekte/pro_stan.htm, 9.10.2001.

[Swatch 2001]
Swatch, In this place we need just one local time, Swatch, http://www.swatch.com/alu_beat/internet_time_brochure.pdf, 12.8.2001.

[Thibodeau 2001]

Thibodeau, P., Budding B2B Standard Faces Big Problems, in: ComputerWorld, 35 (2001) Nr. 7, S. 7.

[Tradecosmos 2001]
Tradecosmos, eCl@ss: Basis für Ihren Erfolg, Tradecosmos, http://www.tradecosmos.com/Produkte/eclass.htm, 25.7.2001.

[UDDI.org 2000a]
UDDI.org, UDDI Overview 9/6/2000, Ariba, Inc., International Business Machines Corporation, Microsoft Corporation, 2000a.

[UDDI.org 2000b]
UDDI.org, UDDI Technical White Paper, September 6, 2000, Ariba, Inc., International Business Machines Corporation, Microsoft Corporation, 2000b.

[von Ammon 2001]
von Ammon, R., Persönliche Kommunikation über die Zukunftsaussichten von EDI und XML, Bea Systems GmbH, Aschheim-Dornach, 2001.

[W3C 2000]
W3C, Simple Object Access Protocol (SOAP) 1.1 - W3C Note 08 May 2000, W3C, http://www.w3.org/TR/SOAP/, 20.9.2001.

[W3C 2001]
W3C, SOAP Version 1.2 - W3C Working Draft 9 July 2001, W3C, http://www.w3.org /TR/2001/WD-soap12-20010709/, 20.9.2001.

[W3C 2001b]
W3C, Web Service Description Language (WSDL) 1.1, W3C, http://www.w3.org /TR/2001/WSDL/, 5.12.2001.

[Weitzel et al. 2001]
Weitzel, T., Harder, T., Buxmann, P., Electronic Business und EDI mit XML, dpunkt.verlag, Heidelberg, 2001.

[Wolf et al. 1999]
Wolf, T., Decker, S., Abecker, A., Unterstützung des Wissensmanagements durch Informations- und Kommunikationstechnologie, in: Electronic Business Engineering, Physica-Verlag, Heidelberg, 1999, S. 745-766.

© HSG / IWI / CCBN / 18