

Universität Kaiserslautern

AG Datenbanken und Informationssysteme

Seminar

**DB-Aspekte des E-Commerce,
Schwerpunkt: Techniken**

Die Bedeutung von XML für E-Commerce

Sommersemester 2001

Marco Müller

Inhaltsverzeichnis:

0 VORWORT	3
1. XML – DIE ENTWICKLUNG	3
1.1 CHARAKTERISIERUNG VON XML – NORMEN.....	4
1.2 EINSATZGEBIETE VON XML.....	6
1.3 XML – DIE FALLEN.....	7
1.4 LÖSUNGSANSÄTZE.....	8
1.4.1 BizTalk.....	8
1.3.2 Eigener goldener Standard.....	9
2. EC – HISTORISCHE ENTWICKLUNG, GRUNDLAGEN	9
2.1 KLASSIFIKATIONSMÖGLICHKEITEN VON EC-ANWENDUNGEN.....	10
2.1.1 Klassifikation nach Art der Beteiligten.....	10
Customer to Customer (C2C).....	10
Business to Customer (B2C).....	10
Business to Business (B2B).....	10
Business to Administration (B2A).....	10
Intra Business (IB).....	11
2.1.2 Klassifikation nach den Transaktionsphasen.....	11
2.1.3 Klassifikation nach der Architektur von EC-Systemen.....	12
2.2 TECHNISCH JURISTISCHE INFRASTRUKTUR.....	13
2.3 RELEVANTE STANDARDS.....	13
2.4 EC-ANWENDUNGSDIENSTE.....	14
2.5 VORTEILE VON XML FÜR E-COMMERCE.....	14
2.5.1 Business.....	15
2.5.2 Customer.....	15
3. EDI – ELECTRONIC DATA INTERCHANGE	16
3.1 UN/EDIFACT.....	16
3.2 AUFBAU EINER EDI-NACHRICHT.....	17
3.3 SUBSETS.....	18
3.4 VORAUSSETZUNGEN FÜR DEN EINSATZ VON EDI.....	18
4. VERGLEICH VON XML UND EDI	19
4.1 XML vs. EDI.....	19
4.2 XML/EDI.....	19
5. AUSBLICK	21
6. LITERATURREFERENZEN UND ONLINE-RESSOURCEN:	21

0 Vorwort

XML – Extensible Markup Language. Jeder hat den Begriff XML mindestens schon einmal gehört, wenn nicht schon sogar im Rahmen eines Praktikums oder ähnlichem damit gearbeitet. In diesem Teil der Seminarreihe soll darauf eingegangen werden, wie XML in E-Commerce Applikationen integriert bzw. sinnvoll genutzt werden kann. Dazu wird im ersten Teil der Begriff XML eingeführt und die Einsatzgebiete und Probleme von XML erörtert. Der zweite Teil beschäftigt sich mit E-Commerce. Klassifikationsmöglichkeiten und relevante Standards werden aufgezeigt, danach wird auf den Nutzen von XML für E-Commerce Anwendungen eingegangen. Im dritten Teil wird der elektronische Datenaustausch (EDI) charakterisiert. Im vierten Teil werden Vor- und Nachteile von EDI erläutert. Des Weiteren werden XML und EDI verglichen, danach wird ein Ansatz vorgestellt, in dem versucht wird, XML und EDI zusammenzufassen, um die Vorteile beider Welten zu verbinden.

1. XML – Die Entwicklung

Der wachsende Anteil an E-Commerce (EC) Anwendungen bringt einen Wandel in der Informationsvermittlung mit sich. Um EC-Geschäfte zu tätigen, brauchen die beteiligten Firmen eine gemeinsame Sprache, um Informationen auszutauschen. EDI und SGML waren als erster Ansatz zu komplex und zu kostenintensiv in der Einführung für kleine Unternehmen. HTML als Seitenbeschreibungssprache war für diesen Zweck nicht geeignet, da diese Sprache in erster Linie zur Formatierung der Informationen gedacht war. Man brauchte eine Sprache, die die Bedeutung der Information ersichtlich macht. XML als Teilmenge von SGML kann die Struktur der Daten und die Bedeutung von Informationen beschreiben, um somit computergestützten Werkzeugen die direkte Benutzung der Information zu ermöglichen. XML wurde in den letzten Jahren von vielen IT-Lieferanten und Benutzergruppen begrüßt. Die Standardisierung von XML war eine der Hauptentwicklungen der IT in den letzten beiden Jahren. Rivalen der Industrie wie z.B. IBM, Microsoft, Sun und Oracle unterstützen den XML 1.0 Standard, entwickeln auf diesem Standard aufbauende Produkte und arbeiten zusammen, um weitere Standards zu entwickeln. XML wurde nun mehr und mehr die Standardplattform, um Geschäfte via EC zu erledigen. XML ist aber nur das Fundament, auf dem Ergebnisse aufzubauen sind - eine genaue Beschreibung wie - fehlt jedoch. In diesem Mangel liegen einerseits die Stärken, andererseits aber auch die Schwächen von XML im EC, die im weiteren Verlauf erörtert werden sollen.

1.1 Charakterisierung von XML – Normen

XML – Extensible Markup Language

XML ist eine vereinfachte Form von SGML (Standard Generalized Markup Language, ISO 8879 [SGML]), entstanden im Jahr 1986. SGML ist sehr generisch und aus diesem Grund sehr allgemein aufgebaut. Sie wird in großen industriellen Dokumenten verwendet, Beispiele hierfür sind Veröffentlichungsberichte der Industrie, Dokumente des Weltraumtechnik, Finanzberichte usw. Der hohe Implementierungsaufwand von SGML hatte zur Folge, daß nur wenige große Firmen die Funktionalität der Sprache in vollem Umfang

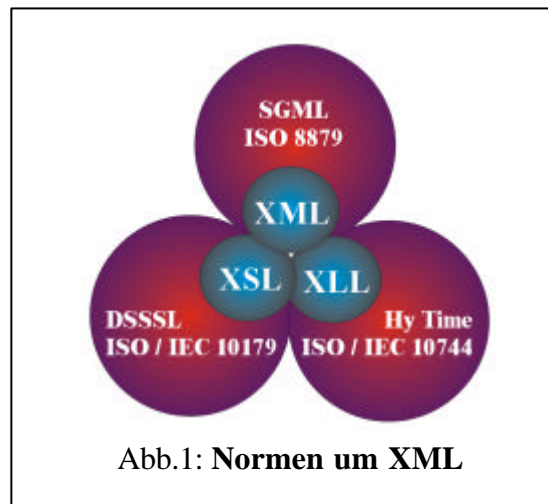


Abb.1: Normen um XML

nutzten. Für kleinere Unternehmen rentierte sich der Einsatz nicht. Jedoch waren die Ideen und die Nützlichkeit dieses Sprachansatzes so von Bedeutung, so daß man ihn auch für die große Zahl kleiner Unternehmen verfügbar machen wollte. XML entstand, eine Sprache, die einfach zu benutzen und zu implementieren war. [SEANMcG]

XML selbst besteht aus einer Menge von Regeln, um individuelle Markierungszeichen (Markup-Tags, im folgenden nur mit Tags bezeichnet) zu definieren, die in erster Linie benutzt werden um ein Dokument in mehrere Teile aufzuteilen und anschließend diese auch identifizieren zu können. XML ist demnach eine erweiterbare Beschreibungssprache. HTML gibt eine feste Menge von Tags vor, in XML ist es möglich, eigene Tags zu definieren. Im Gegensatz zu HTML wird die Struktur der Daten von deren Darstellung getrennt. Daraus resultiert ein flexiblerer Umgang mit den Inhalten der Daten und deren Strukturierung. Diese Trennung erlaubt es, auf relativ einfache Weise etwa mehrere Layoutdefinitionen auf Dokumente anzuwenden. Beispielsweise wäre es denkbar für die Mitarbeiter eines Unternehmens andere Darstellungen eines Dokuments anzufertigen, einzelne Elemente auszublenden und andere stark hervorzuheben. Um solche Formatierungen effizient durchführen zu können, werden separate Dokumente (Stylesheets) erstellt, in denen nur Darstellungsinformationen über das betreffende Dokument vorhanden sind. Der Inhalt von Dokumenten ist auf diese Weise leicht modifizierbar, ohne daß die Formatierungsinformationen verändert werden müssen und umgekehrt. Im Folgenden werden kurz die Begriffe wellformed, valid, DTD und XML Schema erläutert.

- Von **wellformed** (wohlgeformt) spricht man, wenn ein XML Dokument der XML Syntax genügt. Beispielsweise muß für jedes Open-Tag ein Close-Tag existieren, (<item> </item>). Die Tags können geschachtelt werden, dabei dürfen sie sich nicht überlappen, (<a> ist erlaubt, aber nicht <a>). Alternativ zur vorherigen Definition wird ein Dokument als wellformed bezeichnet, wenn ein XML-Dokument eine Baumstruktur aufweist.
- Unter einer **DTD** (Document Type Definition) versteht man eine formale Grammatik zur Beschreibung einer XML Anwendung. Durch eine Grammatik kann eine bestimmte

Sprache definiert werden. In ihr werden die zulässigen Bezeichnungen der Feldinhalte (Tags), sowie deren Anordnung, Kardinalität und Verschachtelung definiert. Des Weiteren kann spezifiziert werden, welche Attribute optional, zwingend erforderlich oder konstant sind, oder auch zu welchen Elementen diese Attribute gehören und welche Werte jeweils zulässig sind. Inhaltliche Beschreibungen der Tags oder auch Hinweise auf die optische Präsentation werden in einer DTD nicht spezifiziert. Ein XML Dokument kann, muß aber nicht, von einer DTD begleitet werden. Auf Basis von DTD können sog. Parser eine Syntaxprüfung (Sprachanalyse) vornehmen. Kommunikationspartner, die Informationen im XML Format austauschen wollen, sollten sich auf eine gemeinsame DTD einigen.

- Von **valid** spricht man, wenn ein XML Dokument wohlgeformt und zusätzlich der in der DTD spezifizierten Struktur entspricht.
- **XML Schema** hat die gleichen Aufgaben wie die vorher erwähnte DTD, jedoch geht der Funktionsumfang noch weit über den von DTD hinaus. Im Gegensatz zu DTD weisen sie keine eigene Struktur auf, sondern sind mit der XML-Syntax verträglich. Um XML Schemata einzulesen, können die üblichen XML-Werkzeuge verwendet werden, spezielle Lösungen werden überflüssig da XML-Schema im XML Format vorliegen. XML wurde für den Datenaustausch über das Internet entworfen, dabei werden die Informationen in einem Format kodiert, das von Maschinen verarbeitet und auch von Menschen gelesen werden kann. Mit wachsender Beliebtheit kamen auch viele neue Benutzergruppen, mit deren speziellen Bedürfnissen, hinzu. Die standardmäßig integrierten Datentypen wie z.B. PCDATA reichten nicht mehr aus, um den Anwendungen zu genügen. Bei Datenbankanwendungen bspw. müssen detaillierte Informationen über die in der Datenbank gespeicherten Werte ausgetauscht werden. Bei anderen Anwendungen wie z.B. Veröffentlichungen ist es notwendig, die Eigenschaften von Überschriften, News oder auch Querverweisen näher zu beschreiben. XML Schema versucht nun den Werten der Elemente (Tags) noch speziellere Typen zuzuordnen, als das in XML normalerweise der Fall ist (also z.B. INTEGER, REAL statt dem generischen PCDATA). XML Schema beinhaltet eine große Auswahl von Datentypen, es ist auch möglich eigene, komplexe Datentypen mittels Aggregation und Vererbung zu definieren, wie es bei objektorientierten Programmiersprachen der Fall ist.

XSL – eXtensible Stylesheet Language

Wie schon vorher erwähnt ist hierbei der Grundgedanke die separate Erfassung der Darstellungsinformationen, in einem von dem Inhalt getrennten Dokument. XSL gliedert sich in drei Teile:

- XSL Transformation (XSLT), mit dessen Hilfe Webserver XML Dokumente vor der Darstellung filtern, umformen und sortieren,
- XPath Language um bei der Transformation auf Teile des XML Dokuments referenzieren zu können und
- einem Vokabular, um die Formatierungsinformationen selbst zu beschreiben.

Parallelen könnten zu Cascading Style Sheets (CSS), die in HTML benutzt werden, gezogen werden. CSS legt die Formatierungseigenschaften Schriftgröße, Ausrichtung von Abschnitten usw. fest. So könnte man z.B. für HTML Dokumente festlegen, daß H1-Elemente mit

Schriftgröße 30, Schriftart Times New Roman, Fett und Unterstrichen dargestellt werden sollen anstatt mit der Default-Einstellung des Browsers. XSL ist mächtiger als CSS und in erster Linie für die Darstellung von XML-Dokumenten entworfen worden. In XSL wird spezifiziert, wie welches Tag des XML Dokuments dargestellt werden soll. XSL Dokumente benutzen die XML Notation, CSS ist eine eigene Sprache. Im Gegensatz zu CSS, daß den Textinhalt immer vollständig auf dem Ausgabemedium anzeigt, kommt hinzu, daß die Anordnung der Elemente vertauscht werden kann, weiterhin ist es auch möglich einzelne Bereiche des Dokuments auszublenden. Die Formatierungen der Tags beruht nicht nur auf dem Namen, sondern auch auf dem Kontext in den es eingebettet ist, d.h. Formatierungen in Abhängigkeit der Attribute und der Stelle, an welcher sich das Tag im Dokument befindet. XSL ist eine Teilmenge von der International Standard Style Language, bekannt unter dem Namen DSSSL (Document Style and Semantics Specification Language, ISO/IEC 10179). Weitere Informationen findet man im Internet unter [XSL].

XLL – eXtensible Linking Language

Analog zu XSL, wird auch die Verkettung der einzelnen XML Dokumente gelöst. XLL ist die Bezeichnung für ein Framework, das versucht, Hyperlinks in XML Dokumente zu integrieren. Mit der Sprache XLink können einfache, unidirektionale Links bis hin zu komplexen Verkettungsstrukturen erstellt werden. Es können Verbindungen zwischen mehr als zwei Ressourcen angelegt und auch Metadaten mit einem Link versehen werden. Ressourcen können dabei Dateien, Bilder, Dokumente, Programme oder auch Anfrageergebnisse von Datenbanken sein. Weiterhin sind bidirektionale Links vorgesehen, die in beide Richtungen referenzieren können, so daß von dem Ziel wieder zur Quelle gesprungen werden kann. Der Pflegeaufwand der Links wird verringert, da diese in einer separaten Datei gespeichert werden. [XLink]

1.2 Einsatzgebiete von XML

Die Einsatzgebiete von XML reichen nach [Edi-Change] von der Verwendung als Auszeichnungssprache für Dokumente über die Beschreibung von Metadaten bis hin zum Datenaustausch im Internet. XML weist eine Nähe zu objektorientierten Programmiersprachen auf. XML Dokumente beschreiben eine Objektstruktur – jedoch nur die Daten und nicht die wie in objektorientierten Sprachen üblichen Methoden.

- Bei der Erstellung von Webseiten spielt XML eine bedeutende Rolle. Es werden einheitliche Ausgabeformate für die Daten geschaffen, die dann in den für die unterschiedlichen Browser optimierten Seitenversionen genutzt werden. Daraus resultiert, daß vor Ort keine spezielle Clientsoftware benötigt wird, sondern nur ein Internetzugang. Demzufolge reduziert sich dann auch der nötige Support auf Standardprobleme der jeweiligen Betriebssysteme und des Webzugriffs. Die Anwendung erzeugt lediglich strukturierte Daten. Unabhängig, wie die Gestaltung aussehen soll – der zu erzeugende Code bleibt gleich. Die XML Struktur wird im Browser mit Hilfe von Stylesheets dargestellt, die die notwendigen Layout-Informationen liefern.

- In HTML werden die Darstellungsinformationen zusammen mit den Inhalten in ein Dokument fest einkodiert. Bei der Benutzung von XML zur Strukturierung des eigentlichen Dokumentinhalts entstehen weitreichende Vorteile wenn man einen für XML geeigneten Browser verwenden kann.
 - (1) Der Inhalt kann verändert bzw. umgeordnet werden, Berechnungen können durchgeführt werden, um zusätzlichen Inhalt (bspw. Statistiken) zu erzeugen. Unter Zuhilfenahme von Java, JavaScript u.ä. können diese Berechnungen auf der Clientseite ausgeführt werden.
 - (2) Der gleiche Inhalt kann für verschiedene Benutzer unterschiedlich dargestellt werden.
- Mit XML werden Suchanfragen präzisiert. Mit der Hilfe von sog. „intelligenten Software-Agenten“, können in Verbindung mit DTD Tags interpretiert werden. Es werden nur relevante und redundanzfreie Seiten ausgegeben.
- Das Einsatzspektrum für elektronische Austauschbeziehungen wird wesentlich erweitert, da XML sowohl Mensch-Maschine-Kommunikation als auch Maschine-Maschine-Kommunikation unterstützt. Mit den früheren Lösungen (SGML, EDI) wurden nur Business-to-Business Lösungen unterstützt, XML hingegen kann auch für den Datentransfer zwischen Endkonsumenten herangezogen werden.
- Es können unterschiedliche Datenarten (VITA) integriert werden, da XML die Dokumente inhaltlich beschreibt und Beziehungen zwischen diesen definieren kann.
- Wie schon in Abschnitt 1.2 erwähnt, ist es mit XSL möglich, unterschiedliche Sichten auf Daten zu bilden. Durch die variable Informationsdarstellung von XML ist es möglich, je nach Ausgabemedium und Benutzergruppen das Layout der Daten zu definieren.
- Zuletzt können Daten strukturierter und intelligenter verwaltet werden. Dazu wird eine XML-fähige Datenbank benötigt, in der die Informationen aufgebrochen, indiziert und gespeichert werden. Beim Data-Mining können Informationen schnell und flexibel gewonnen werden.

Einige Beispiele für XML-Applikationen sind :

- OFX – Open Financial Exchange, Austausch von Daten zwischen Banken
- OSD – Open Software Description, Beschreibung von Software Modulen
- PGML – Austausch von Vektorgrafiken
- MATHML – Austausch mathematischer Formeln

1.3 XML – Die Fallen

Durch die verbreitete Benutzung von XML erhöht sich die Komplexität, da jedes Unternehmen ihre eigene Struktur der Daten festlegen kann – zugeschnitten auf den eigenen Informationsbedarf. Anfangs entstanden für die einzelnen Industriebereiche eine Reihe von XML-Strukturen, wie z.B.: FIN-XML für Geldgeschäfte oder C-XML für verschiedene kaufmännische Bereiche. Jedes Unternehmen kann eigene Schnittstellen definieren, um die unternehmensspezifische Struktur auf die jeweiligen XML Formate anzupassen. Dies ist ein

beträchtliches Unterfangen, aber machbar. Unglücklicherweise ist nicht nur ein XML Standard pro Industriesektor entstanden, sogar Zusammenschlüsse einzelner Unternehmen haben ihre eigenen Standards entwickelt. Es wird auf Dauer nicht möglich sein, zu mehreren Standards eigene Schnittstellen innerhalb des Unternehmens anzubieten. [XMLPP] Gründe:

- Ein Unternehmen wird nicht nur in einem Marksektor tätig sein;
- Ein solcher Standard wird i.a. den Bedürfnissen eines Unternehmens nicht entsprechen;
- Geschäftspartner unterstützen andere Standards;

1.4 Lösungsansätze

Im Folgenden werden Ansätze betrachtet, wie sich Unternehmen vor der wachsenden Sprachenvielfalt von XML isolieren können.

1.4.1 BizTalk

Das BizTalk Framework, gefördert durch Microsoft und Partner, will es den Unternehmen ermöglichen, XML Nachrichtenformate der unterschiedlichen Anbieter auf die eigenen Bedürfnisse anzupassen und diejenigen herauszusuchen, die am ehesten den unternehmensspezifischen Bedürfnissen und Anwendungen genügen. Das BizTalk Framework arbeitet dazu in 3 Schritten. Zuerst wird es eine sog. „kanonische Form“ generiert, in der alle anwendungsspezifischen Nachrichtenformate von XML definiert werden können. Zum zweiten stellt es ein sog. Repository bereit, an dem BizTalk konforme XML Nachrichtenformate öffentlich validiert, eingereicht, erhalten und frei benutzt werden können [BizTalk]. Die Übersetzungen der Nachrichtenformate werden im dritten Schritt XSL-basiert durchgeführt. Im Folgenden wird anhand Abb.2 die Funktionsweise dieses Ansatzes erläutert.

Die eigene Firma benutzt den BizTalk konformen Standard A, und richtet die Schnittstellen darauf aus. Eine anderes Unternehmen benutzt den BizTalk konformen Standard B. Indem nun XSL-Transformationen (erhältlich von dem BizTalk repository) zwischen den beiden Standards durchgeführt werden, kann man nun Nachrichten im Standard A versenden, die der Empfänger übersetzen kann. Wenn genug Standards in BizTalk aufgenommen werden, kann man zu jedem Geschäftspartner Nachrichten senden, der diesen Standard auch benutzt.

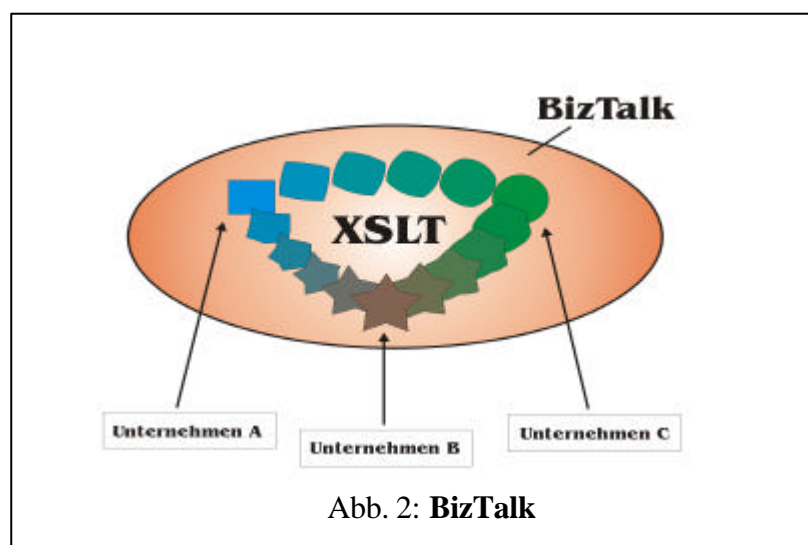


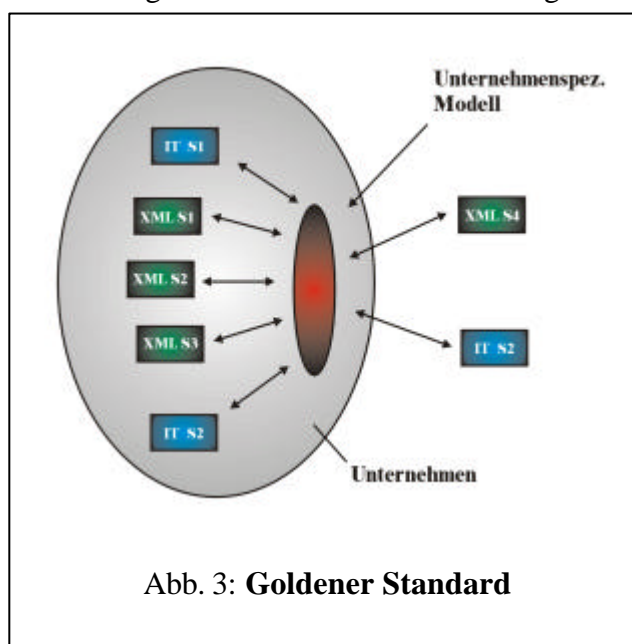
Abb. 2: BizTalk

Dieser Ansatz wirft aber auch Fragen und Probleme auf. Kann ein Unternehmen sich darauf verlassen, daß Microsoft und Partner alle Standards berücksichtigen die das Unternehmen braucht um mit anderen Nachrichten auszutauschen? Kann angenommen werden, daß der

eigene Standard korrekt übersetzt wird. BizTalk muß jede der einzelnen XML-Strukturen verstehen um die Transformation durchführen zu können. Mit BizTalk sind $N*(N-1)$ Transformationen nötig, um die einzelnen XML-Schema ineinander umzuwandeln. Man spricht hier von einem N^2 Problem. Es existieren heute schon mehr als 100 XML basierte Nachrichtenstandards. Es wäre ein gewaltiges Unterfangen einen neuen Nachrichtenstandard in alle anderen transformieren zu wollen bzw. alle Transformationen zwischen den Schemata zu warten. Wenn man eine effiziente Lösung sucht, dann liegt sie innerhalb des eigenen Unternehmens.

1.3.2 Eigener goldener Standard

Der Grundgedanke bei diesem Ansatz ist, das sich kein anderes Unternehmen im selben Business befindet, wie das Eigene. Auf dieser Grundlage wird ein Modell zusammengestellt, das auf den Informationsbedarf des Unternehmens zugeschnitten ist. Wie in Abb.3 gezeigt, werden auf dieses logische Modell die Ergebnisse der eigenen Anwendungssysteme abgebildet und Übersetzungsalgorithmen für die in der Unternehmung benutzten XML Formate erstellt. Um mit anderen Unternehmen Daten auszutauschen, wird jedes externe Nachrichtenformat in das eigene Modell übersetzt. Dadurch wird das N^2 -Problem gelöst, da jeder Übersetzungsvorgang zweistufig durchgeführt wird. Die Unternehmung erhält einen entscheidenden Vorteil: Sie wird von dem wachsendem Daten-Wirrwarr isoliert, kann sich aber bei Bedarf schnell an neue Modelle anpassen, denn dieser Vorgang ist in der Zeit des EC der Schlüssel zum Erfolg.



2. EC – Historische Entwicklung, Grundlagen

Die automatische Abwicklung von Handelstransaktionen mit Hilfe von Kommunikationsnetzen wird als EC bezeichnet. Seit Anfang der 70er Jahre wurde beruhend auf dieser Definition EC mit der Grundlage des elektronischen Datenaustausches (EDI) betrieben. EC ist kann sich zur Zeit großer Beliebtheit erfreuen. Das läßt sich an der großen Vielfalt der Anwendungsmöglichkeiten beobachten, die sich durch das Zusammentreffen des WWW, preiswerter Kommunikationstechnologien, bzw. -verbindungen und nicht zuletzt durch erschwingliche Hard- & Software gebildet haben. Des weiteren kommen Standardisierungen von Datenaustausch- bzw. Kommunikationsprotokollen (ICE, SOAP) und Sicherheitsmechanismen hinzu, die für die Abwicklung von EC Geschäften notwendig sind. Beispiele hierfür sind das Internet selbst, Verfahren zur Authentisierung und Vertrauensbildung, Verschlüsselungsalgorithmen, digitale Signaturen, Zertifizierung und nicht

zuletzt elektronische Bezahlverfahren. Verteilte Systeme stellen einen wichtigen Ausgangspunkt für Kommunikations- und Kooperationsprotokolle dar, weil der elektronische Handel von Natur aus die Grenzen einzelner Organisationen überschreitet. Im weiteren wird versucht den Begriff des EC näher zu charakterisieren und auf dieser Grundlage die Einsatzgebiete des EC herauszuarbeiten. [PXML]

2.1 Klassifikationsmöglichkeiten von EC-Anwendungen

Es stehen mehrere Möglichkeiten zur Klassifikation zur Auswahl. Üblich ist es, Handelsbeziehungen des EC nach der Art der daran beteiligten Geschäftspartner zu klassifizieren. Weitere Möglichkeiten zur Klassifikation bieten sich nach den Phasen, die eine Handelstransaktion durchläuft, oder auch nach dem Transaktionsvolumen einer Transaktion.

2.1.1 Klassifikation nach Art der Beteiligten

Customer to Customer (C2C)

Diese Ausprägung ist im Internet auch häufig anzutreffen. Die EC-Geschäfte werden zwischen Privatpersonen abgewickelt. eBay als bekanntes Beispiel bietet zeitlich begrenzte Versteigerungen an, bei denen Endkunden selbst Angebote aufgeben und auch mitbieten können.

Business to Customer (B2C)

Zusammen mit B2B ist diese Form des EC wohl die am häufigste anzutreffende Form des EC. Dieser Bereich bezieht sich auf den Online-Handel zwischen Unternehmen und Privatpersonen. Die Händler verfolgen mit ihren Aktivitäten hierbei die Strategie, die Waren direkt an die Endkunden abzusetzen. Kennzeichen für diesen Bereich sind zum einen die Spontaneität der Geschäfte, bei der sich ein Kunde kurzfristig zum Kauf entscheidet und zum anderen die „kleinen“ Transaktionsvolumina. Neben der Produktauswahl und der Auslieferung ist es heute nichts Neues die Produkte auch online zu bezahlen.

Business to Business (B2B)

Dieser Zweig des EC beschäftigt sich mit den Geschäftsprozessen von Unternehmen untereinander. In der Regel besteht schon eine längerfristige Geschäftsbeziehung der an der Transaktion beteiligten Betrieben. Hier steht der Aufbau flexibler Kooperationstechniken zwischen den jeweils beteiligten IT-Systemen im Zentrum der Betrachtung. Als vorherrschendes Muster stehen hier die flexible Organisation von Regeln, Rollen, Abläufen und Kommunikationstechnologien im Vordergrund. Üblicherweise werden bei den in dieser Sparte anzutreffenden Geschäfte „größere“ Transaktionsvolumina beobachtet.

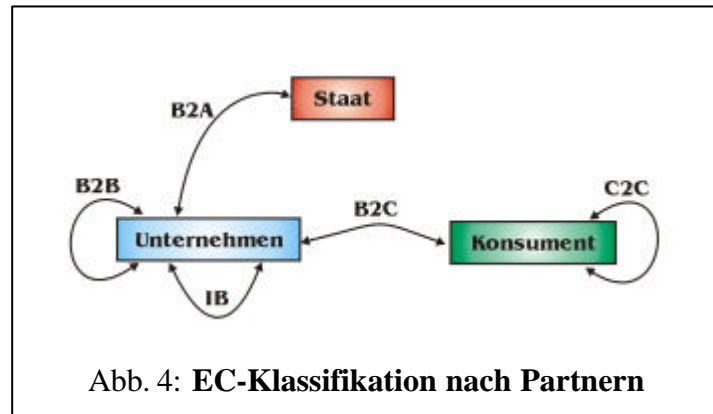
Business to Administration (B2A)

In diesen Bereich kann man beispielsweise des öffentliche Beschaffungswesen hinzuzählen. Mit Hilfe von Auktionssystemen erreicht man hierbei eine hohes Maß an Transparenz mit der zwangsläufig auch eine höhere Transaktionseffizienz erreicht wird.

Intra Business (IB)

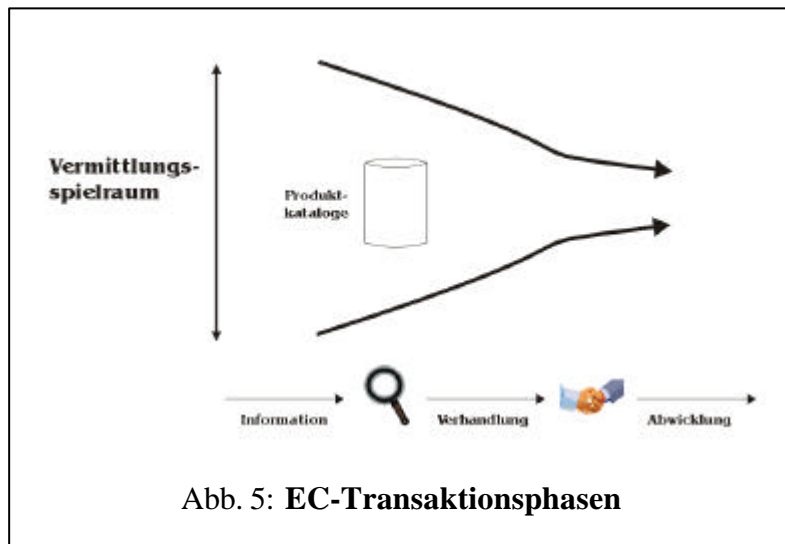
Hier steht nur ein Unternehmen in der Betrachtung, bei der nur Transaktionen innerhalb dieser von Bedeutung sind. Beispielsweise wickelt eine Bank einen Teil der Geldgeschäfte mit ihren Filialen ab.

Diese Art der Klassifizierung soll in Abb. 4 veranschaulicht werden.



2.1.2 Klassifikation nach den Transaktionsphasen

Bei dieser Klassifikation werden die grundlegenden Phasen einer Handelstransaktion unterschieden. In der **Informationsphase** verschaffen sich die Kunden eine Produkt- und Marktübersicht. Sie engen ihre Wahl soweit ein, bis nur die gewünschten Produkte übrig bleiben (siehe Abb.5). Unterstützt wird dieser Vorgang durch elektronische Produktkataloge bzw. Preisagenten. In der darauffolgenden **Verhandlungsphase** werden Preiskonditionen und Produkteigenschaften ausgehandelt. Der Umfang dieser Phase reicht vom einfachen Akzeptieren bis hin zum Einsatz von komplexen Verhandlungsformen. Ersterem genügt der Einsatz von definierten Kommunikationsprotokollen, für letzteren ist eine Unterstützung durch Gruppenkommunikations- bis hin zu Workflowsystemen möglich bzw. notwendig. Zuletzt werden in der **Abwicklungs- oder Ausführungsphase** die entsprechenden Leistungen (Zahlungen, Dienstleistungen oder auch Rechte) ausgetauscht. Vielfach werden zum Übergang zwischen den Phasen Werkzeuge eingesetzt. So findet man zwischen Phase eins und zwei oft so genannte Broker. Deren Einsatzgebiete erstrecken sich von Datenbankanwendungen mit einfachen Anfragen über komplementäre Angebote bis hin zum Vermitteln zwischen Teilnehmern. Zwischen Phase zwei und drei kann der Übergang z.B. von elektronischen Notariatsdiensten unterstützt werden, die den Vertragsabschluß durch Verifikation von Signaturen und Archivieren des Vertrages vornehmen [MTL99].



2.1.3 Klassifikation nach der Architektur von EC-Systemen

In dieser Betrachtungsweise lassen sich 3 verschiedene technologische und organisatorische Ebenen identifizieren.

Die **Kommunikationsebene** ist für den standardisierten Austausch von Nachrichten unterschiedlicher Art verantwortlich. Beispielsweise sind im B2C-Bereich das TCP/IP und HTTP-Protokoll anzutreffen, mit deren Hilfe auf Webserver zugegriffen und somit in Online Shops gestöbert werden kann. Wohingegen es im B2B-Bereich vielmehr auf einen zuverlässigen Austausch von Nachrichten ankommt, finden hier so genannte Value-added-Networking Services (VAN) ihre Anwendung. Des Weiteren wird dieser Ebene das Internet-Inter-ORB-Protokoll (IIOP) von CORBA (Common Object Request Broker Architecture) zugeordnet.

Die darüber angeordnete Ebene entspricht der **Darstellungsebene** im ISO/OSI-Referenzmodell (Schicht 6). Diese Schicht beschäftigt sich mit der Kodierung / Dekodierung von Nachrichten (Transfersyntax) und mit der Standardisierung von Nachrichtenformaten. Als Beispiele seien hier die HTML-Syntax, XML-Dokumententypen und EDI-Nachrichtentypen genannt.

Auf der nächsten Ebene befinden sich kommunizierende Anwendungssysteme, die aus softwarearchitektureller Sicht nicht mehr zu unterscheiden sind (CORBA, WWW-Anwendungen, verteiltes Java usw.). Innerhalb und über dieser Ebene lassen sich EC-Systeme nur schwer allgemeingültig fassen. Deswegen ist ab hier eine **organisatorische Sichtweise** notwendig. Häufig lassen sich hier die EC-Komponenten als Dienstbringer (Server) und Dienstnehmer (Client) gegeneinander abgrenzen. An dieser Stelle spielen Frameworks eine wichtige Rolle, bei denen ein allgemeingültiges Entwurfsmuster eingesetzt wird und softwaretechnisch Schablonen vorgegeben werden, welche die Ereignisse, Attribute und Methoden von den Anwendungsobjekten vorschreiben. Durch diesen Ansatz können die Anwendungen flexibel in eine generische Infrastruktur eingebettet werden. Als Beispiele können einerseits "JWallet" von Sun Micro Systems zur Integration von Zahlungsinstrumenten und andererseits das "San-Francisco Framework" von IBM zur einheitlichen Spezifikation von Geschäftsprotokollen genannt werden.

Das fundamentale Problem bei EC-Architekturen liegt in der technologischen Schnellebigkeit der beteiligten Funktionen. Daher muß eine Architektur stark von konkreten Anwendungsfällen abstrahieren, denn nur so kann die Software dem Wandel am Markt gerecht werden. Die vorher erwähnten Frameworks erweisen sich dabei als wichtige Grundlage.

2.2 Technisch juristische Infrastruktur

Da EC räumlich nicht eingegrenzt werden kann, spielt die Notwendigkeit einer internationalen harmonisierten Regelung eine bedeutende Rolle. Dabei müssen sehr unterschiedliche Bereiche wie zum Beispiel Fragen des Steuerrechts, des Datenschutzgesetzes, Verbraucherschutzes, Vertragsrechts sowie der einheitlichen Zulassung und Anerkennung erforderlicher kryptografischer Verfahren geklärt werden, worauf hier aber wegen des begrenzten Rahmens nicht weiter eingegangen werden kann.

2.3 Relevante Standards

Aufgrund des internationalen Charakters von EC-Geschäften werden Standards auf unterschiedlichen architekturellen Ebenen zwangsläufig unentbehrlich. Unabhängig von den vorher schon erwähnten Standards der Kommunikationsebene sind Normen vor allem auf den anwendungsnahen Ebenen und in den verschiedenartigsten Fachgebieten erforderlich. Im weiteren werden diese nur kurz erläutert da auf einige noch genauer in den folgenden Vorträgen der Seminarreihe eingegangen wird.

- **Kryptographie:** Die Kryptographie beschäftigt sich mit der Verschlüsselung von Nachrichten, damit die Information von Unbefugten nicht gelesen werden kann. Die notwendigen Algorithmen sind jedem an der Transaktion Beteiligten bekannt, so daß als einziges Geheimnis nur die verwendeten Schlüssel bleiben. Beispiele hierfür sind : DES, RSA zur symmetrischen und asymmetrischen Verschlüsselung sowie MD5 und SHA zur Sicherung der Integrität und schließlich RSA und DAS für digitale Signaturen.
- **Vertrauensinfrastruktur:** Vom technischen Hintergrund aus, können durch Zertifikate zwischen den Beteiligten ein Mindestmaß an Vertrauen gewonnen werden. Als Beispiel ist hier der 509-Standard der CCITT/ISO für Verzeichnisdienste entwickelt worden, er schreibt vor, wie Zertifikate zu strukturieren sind, sowie Protokolle und Verfahren zur Verwaltung dieser Informationen.
- **Zahlungsverfahren:** In diesem Bereich sind viele Standards entwickelt worden, die jedoch zumeist Industrieprodukte oder Ergebnisse von Industriekonsortien sind. Als Paradebeispiel sei hierfür Secure Electronic Transaction (SET) genannt, welches von Visa, MasterCard, IBM und Microsoft definiert wurde.
- **Elektronischer Datenaustausch:** Hier wäre der EDI- Standard zu erwähnen, auf den aber im dritten Teil noch näher eingegangen wird.
- **Softwareschnittstellen und Frameworks:** Auf der Schicht der Anwendungssoftware von Geschäftspartnern ist es notwendig, Softwarekomponenten und deren Schnittstellen zu standardisieren, um die Kommunikation über Unternehmensgrenzen hinweg

durchführen zu können. Dies wird durch das Prinzip der vorher genannten Frameworks unterstützt.

- **Dokumentenformate:** Hierbei spielt XML als Basis für spezielle Sprachen eine bedeutende Rolle. So beruht RDF (Resource Description Framework) zum Beispiel auf einer Erweiterung von XML, mit deren Hilfe Suchmaschinen und elektronische Bibliotheken Metainformationen über Dokumente austauschen. Dabei hilft RDF die Syntax der Metainformation zu definieren.
- **Meßstandards:** Betreiber von EC-Geschäften möchten gerne über ihre Erfolge informiert werden. So zum Beispiel hat die IVW (Informationsgemeinschaft zur Feststellung der Verbreitung von Werbeträgern e.V.) einen Meßstandard „AdClick“ entwickelt, zur einheitlichen Zählung der Besuche einer Seite im Internet. Mittels dieser Norm erreicht man eine bessere Kontrolle von Werbemaßnahmen und Besuchervolumen im Internet.

2.4 EC-Anwendungsdienste

In diesem Abschnitt soll ein kurzer Überblick von EC-Anwendungen gegeben werden. Als erste Anwendung seien hier die **Online-Kataloge** genannt. Diese bieten meistens navigierenden Zugriff und Auswahl von Produkten nach Suchkriterien. Sie werden auf Skriptbasis in das WWW integriert und mit zusätzlichen Funktionen wie z.B. elektronischen Einkaufskörben ausgestattet. Eine spezielle Form von Online-Katalogen sind sog. Marktplatzsysteme, hauptsächlich für den B2C-Bereich gedacht. Diese werden für Stellenangebote von Unternehmen genutzt, die vom Kunden Ad-Hoc abgerufen werden können.

Eine weitere Form von EC-Anwendungen sind sog. **Broker** bzw. **Makler**. Die Aufgabe eines Brokers besteht darin, dem Kunden während der Informationsphase geeignete Geschäftspartner zu finden. Diese Aufgabe wird mit Hilfe von Profilen über die zur Auswahl stehenden Teilnehmer erreicht. Agenten werden im Internet als Suchwerkzeuge eingesetzt. Der Agent – realisiert als Skript – prüft dazu eine vorgegebene Liste von Anbietern nach vorgegebenen Kriterien. Es werden zwei Formen dieser Agenten unterschieden. Zum einen existieren sog. „Intelligente Agenten“, denen zusätzliche Eigenschaften wie z.B. der Fähigkeit zum Schlußfolgern oder auch dem proaktiven Handeln zugesprochen werden. Zum anderen gibt es sog. „Mobile Agenten“ die nach eigener Entscheidung das Netz durchsuchen um ihre Arbeit zu erledigen. Zur Effizienzsteigerung von Geschäftsabschlüssen kommen Auktionsdienste zum Einsatz. Beispiele hierfür sind das Info-Flyway der Lufthansa für B2C oder eBay für den C2C Bereich. Des weiteren seien noch genannt: Elektronische Vertragssysteme und Workflow-Management-Systeme.

2.5 Vorteile von XML für E-Commerce

Nach dem E-Commerce klassifiziert und Anwendungsdiensten näher betrachtet wurden, ist es in diesem Abschnitt das Ziel, die Vorteile von XML für E-Commerce herauszuarbeiten. Dabei wird zuerst auf den Business- danach auf den Customer-Bereich näher eingegangen.

2.5.1 Business

(1) XML ist plattformunabhängig und anwendungsneutral, beruht auf standardisierten Technologien, wie z.B. SGML für die Syntax, URI für Referenzierung, EBNF für die Grammatik und UNICODE für die Kodierung der Zeichen. XML kann somit die weitreichenden Vorteile von Standardisierungen der Basistechnologien für sich verbuchen.

(2) Layoutdefinitionen für verschiedenste Medien (Bildschirm, Palm, Papier) sind möglich. Dadurch resultiert auch eine bessere Wartbarkeit und Veränderbarkeit der jeweils voneinander getrennten Dokumente.

(3) Herstellerspezifische bzw. binäre Formate sind nur von Bedeutung, wenn es Systeme gibt, die diese Formate unterstützen. Um der Schnellebigkeit des Informationszeitalters gerecht zu werden, müssen immer mehr Systeme ausgetauscht werden, woraus sich Probleme mit den Datenbeständen der Unternehmen ergeben. XML-Dokumente bestehen aus Plain-text, das auch zukünftig noch von jedem System verarbeitet werden kann.

(4) Der Datenaustausch zwischen Unternehmen stellt wegen der Vielfalt der Anwendungssoftware eine beachtliche Hürde dar. Mit der Hilfe von XML müssen kooperierende Firmen nur ein gemeinsames Vokabular aushandeln. Es ergeben sich für Unternehmen keine Einschränkungen, denn sie müssen keine Annahmen über die Systeme (Plattformen, Betriebssysteme, Programmiersprachen) des Gegenüber machen, um mit diesem Daten auszutauschen.

(5) XML bietet als Metasprache ein Framework an, um ein auf den eigenen Bedarf zugeschnittenes Vokabular zu definieren. Im Gegensatz zu SGML kann dies ohne jahrelange Erfahrung auf einfachem Wege geschehen.

(6) Der größte Vorteil für das Geschäftswesen ist wohl darin zu sehen, daß sich auf globaler Ebene potentielle Kunden ansprechen lassen, ohne physikalisch vor Ort zu sein. Dazu muß die in der betreffenden Region anzutreffende Sprache unterstützt werden, was aber aufgrund der UNICODE-Unterstützung für XML heute wohl kaum mehr ein Problem darstellt.

2.5.2 Customer

(1) Neben den schon „alltäglich“ gewordenen Zugangsmöglichkeiten wie Home-PC und Notebook kommen vermehrt auch Palm und Mobil-Telefone für den Web Zugriff in Mode. XML ist in der Hinsicht skalierbar, indem es für jedes Endgerät Unterstützung via XSL bietet, ohne daß Anpassungen auf der Clientseite notwendig werden.

(2) Eine Unterstützung für die Suche im Netz wäre denkbar. Hierbei müßte die Suchmaschine zusätzliche Informationen über die Semantik der Tags erlangen. Dies ist jedoch ein schwieriges Unterfangen, wenn man zu der großen Anzahl der Tags noch die Vielzahl der Bedeutungen dieser betrachtet. Hierfür wären wiederum Standards notwendig.

(3) Der Inhalt von XML Dokumenten könnte in Verbindung mit sog. „Smart Agents“ auf die Präferenzen des Benutzers zugeschnitten werden.

3. EDI – Electronic Data Interchange

Mit EDI wird jede Art des strukturierten Austausches von Daten zwischen Anwendungsprogrammen entfernter Kommunikationspartner über Netze unter Verwendung einer vorher vereinbarten Norm bezeichnet.

Die mit der Hilfe von EDI übertragenen Daten sind nach bestimmten Regeln formatiert und nach einer Reihenfolge geordnet. Mit EDI werden Daten direkt zwischen Anwendungsprogrammen übertragen. Das bedeutet, daß eine erneute Eingabe von Informationen über diese Daten auf der Empfängerseite nicht notwendig wird. Das setzt voraus, daß die Kommunikationspartner ein bestimmtes Format vereinbaren oder sich auf ein Standardformat einigen. Anwendungsprogramme, zwischen denen der Datenaustausch durchgeführt wird sind z.B. Warenwirtschaftssysteme, Lagerbestandsverwaltungen u.ä. Analog zu den Problemen mit den unzähligen XML Sprachen (siehe Kapitel 1.5-1.6) liegt auch hier ein technisches Grundproblem beim Einsatz von EDI vor. Es besteht darin, daß die Kommunikationspartner im Regelfall unterschiedliche Anwendungsprogramme auf verschiedenen Systemplattformen mit jeweils spezifischen internen Datenstrukturen benutzen. Aus diesem Grund können die Daten nicht direkt zwischen den Anwendungsprogrammen übertragen werden. Sie müssen individuell an den die eigenen Strukturen angepaßt werden, damit sie weiterverarbeitet werden können. Der gravierende Nachteil hierbei ist, daß bei jedem neu hinzukommenden Kommunikationspartner mit einer jeweils anderen Datenstruktur individuelle Anpassungen erforderlich werden. Aus diesem Grund versuchte man schon früh ein standardisiertes Datenaustauschformat zu verwirklichen, so daß dann für jeden Kommunikationspartner nur noch eine Anpassung notwendig wurde. Zuerst versuchte man Datenformate zu entwickeln, die in speziellen Branchen am besten geeignet für den jeweiligen Datenaustausch waren. So entstanden z.B.

- **SEDAS** (Standardregelungen Einheitlicher Datenaustausch-Systeme) für die deutsche Konsumgüterindustrie,
- **VDA** (Verband der Deutschen Automobilindustrie) für die deutsche Automobilindustrie,
- **ODETTE** (Organization for Data Exchange by Teletransmission in Europe) für die europäische Automobilindustrie,
- **RINET** (Reinsurance and Insurance Network) für die europäische Versicherungswirtschaft,
- **SWIFT** (Society for Worldwide Interbank Financial Telecommunication) für internationale Bankenwirtschaft

3.1 UN/EDIFACT

Da Unternehmen auch mit zahlreichen Firmen anderer Branchen kommunizieren, liegt das gleiche Problem, nur auf einer höheren Ebene vor. Das hat zu Überlegungen geführt, einen branchenübergreifenden und internationalen Standard zu entwickeln. Mit dieser Aufgabenstellung beschäftigt sich seit den 60er Jahren die Wirtschaftskommission der Vereinten Nationen für Europa (UN/ECE – United Nations Economic Commission for Europe). 1987 wurde als Ergebnis UN/EDIFACT (United Nations Electronic Data Interchange for

Administration, Commerce and Transport) als branchenneutrale EDI-Lösung veröffentlicht. Als Norm hat sich EDIFACT sowohl international (ISO-NORM) als auch in der BRD durchgesetzt (DIN-Norm). Im engeren Sinn versteht man darunter die EDIFACT-Syntax Regeln nach ISO 9735. Wie alle EDI Standards wird hierdurch der Austausch von Geschäftsinformationen auf elektronischem Wege zwischen Herstellern, Groß- und Einzelhändlern, Handwerkern, Im- und Exporteuren usw. erleichtert. Gegenüber den branchenüblichen Standards finden sich die Einsatzgebiete von UN/EDIFACT in Unternehmen, die ein großes Transaktionsvolumen aufweisen und/oder geschäftlich mit internationalen Partnern Daten austauschen.

3.2 Aufbau einer EDI-Nachricht

Die Normungsgremien haben verschiedene Regelungen vereinbart, nach diesen der Aufbau einer EDI – Nachricht folgt. Diese Regelungen beziehen sich auf :

- die zu verwendeten Nachrichtenformate,
- die Datenelemente und ihre Zusammensetzung zu sogenannten Datenelementgruppen, Segmenten, Nachrichten, Nachrichtengruppen und Übertragungsdateien
- die Aufteilung in Muß- und Kann-Elemente,
- die Trennzeichen, mit denen die Nachrichteninhalte abgegrenzt werden.

Eine EDIFACT Nachricht wird als fortlaufende Zeichenkette übertragen, wie der Abb. 6 entnommen werden kann.

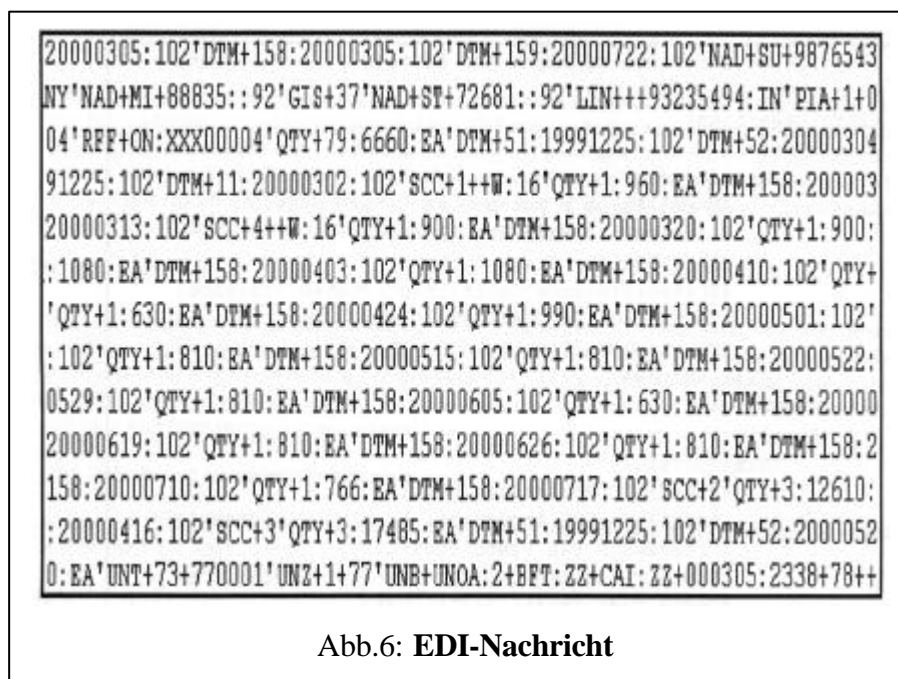


Abb.6: EDI-Nachricht

3.3 Subsets

Damit viele Anwendungsfälle abgedeckt werden können, sind die genormten Nachrichtentypen sehr komplex. In der offengehaltenen, sehr generischen Auslegung des Standards liegt auch die Problematik von EDIFACT. Um die Branchenunabhängigkeit zu verwirklichen muß eine große Menge von Datenelementen bereitgehalten, gepflegt und übermittelt werden. Um den Aufwand zu reduzieren entstanden sog. „Subsets“ von EDIFACT. Dabei war bzw. ist das Ziel, die für eine bestimmte Anwendergruppe relevanten Datenelemente auszuwählen und den Feldern eine bestimmte Bedeutung zuzuweisen, wobei stets die Konformität zu EDIFACT gewährleistet bleibt. Diese Subsets beziehen sich i. d. R. auf bestimmte Branchen. Beispiele hierfür sind:

- CEFIC (chem. Industrie)
- EANCOM (Konsumgüterindustrie)
- EDIFICE (Elektroindustrie)
- ELFE (Elektr. Fernmelderechnungen der dt. Telekom)

Weil die Definition von Subsets der angestrebten Branchenunabhängigkeit entgegenwirkt, versucht die DEDIG (Deutsche EDI-Gesellschaft) den Subset-Wildwuchs zu begrenzen, indem bspw. ähnliche Subsets zusammengelegt werden. Weiterhin können neue Subsets der DEDIG gemeldet werden, sie werden anschließend auf Normverträglichkeit geprüft und ggf. registriert. Die Hoffnungen gehen in die Richtung, daß die Zahl der Subsets zurückgehen wird und der elektronische Datenaustausch auf der Grundlage von EDIFACT ganz ohne Subsets auskommen wird.

3.4 Voraussetzungen für den Einsatz von EDI

Bei EDI handelt es sich um den Datenaustausch zwischen Anwendungsprogrammen. Als erste Voraussetzung müssen demnach auch solche Programme eingesetzt und daß diese über geeignete Schnittstellen zum Datenimport und -export bieten. Zum zweiten sind noch weitere technische Komponenten für den Einsatz notwendig. Selektionsprogramme dienen dazu, die relevanten Daten aus dem Anwendungsprogramm herauszufiltern, um sie einem Konverter



Abb.7: Voraussetzungen für den EDI-Einsatz

zur Verfügung zu stellen, der dann das interne Format der Anwendung in ein standardisiertes Datenaustauschformat umwandelt und umgekehrt. Abb.7 veranschaulicht den Datenaustausch mittels EDI.

4. Vergleich von XML und EDI

EDI war eine lange Zeit – vor XML – die einzige Möglichkeit, EC Geschäfte zu tätigen. Bei der Benutzung von EDI können jahrzehntelange Erfahrungen im Umgang mit Problemen des EC im B2B bzw. Ideen im Standardisierungsvorgang weiter genutzt werden. Ein weiterer positiver Aspekt ist eine sehr gute Dokumentation.

Es ist jedoch nicht alles Gold was glänzt, EDI hat auch mehrere Nachteile. Zum Einen existiert eine Vielzahl von Konstrukten, die nur schwierig einzuprägen und zu benutzen sind. Des weiteren kommt noch eine gute Portion Unleserlichkeit der Nachrichten hinzu, siehe Abb. 6. Zum Anderen sind die Programmierrichtlinien zu ungenau gehalten, wodurch eine große Freiheit resultiert, wie Segmente benutzt werden können, die Syntax ist überladen, d.h. es gibt viele Wege um Nachrichten zu kodieren. Überschwenglich lassen Personen aus der EDI-Branche verlauten, daß nur 20% der Möglichkeiten eingesetzt werden – das Problem ist jedoch, daß niemand dieselben 20% benutzt. Aufgrund dessen ist es nicht sonderlich einfach, ein Validationsprozeß durchzuführen, dies schlägt sich in Integrationskosten nieder.

4.1 XML vs. EDI

Wegen der hohen Integrationskosten ist EDI der kostenintensivere aber auch der effizientere Ansatz, um Informationen in der Geschäftswelt auszutauschen. XML hingegen, ist ein offener Ansatz über das Internet, der weitaus flexibler ist. XML ist billiger in der Einführung, jedoch weniger ausgereift als EDI. Die Wahl von XML oder EDI ist sehr branchenspezifisch. XML wird nach [MAXELON] über kurz oder lang der Standard für B2B Kommunikation werden. In den Branchen, in der sich ein XML Standard durchgesetzt hat bzw. akzeptiert wurde, ist es ratsam diesen auch zu benutzen. Wenn sich jedoch ein XML Standard noch nicht durchsetzen konnte, ist der EDI Ansatz zu bevorzugen, da bei der Benutzung von XML aufgrund der Anpassungskosten noch weitaus höhere Kosten entstehen können, da in dieser Branche erst noch der Standard gefunden werden muß. Aktuelle Informationen hierzu können bei [XMLORG] gefunden werden.

4.2 XML/EDI

Es wird versucht die Vorteile von XML und EDI zu verbinden. EDI-Systeme verwenden eindeutige Segment-Identifizierer, diese können durch XML-Token ersetzt werden. Jedoch ist es nicht praktikabel alle EDI Konstrukte in XML abzubilden. Es ist vielmehr wichtig zu wissen, welche von diesen sich erfolgreich aufgrund ihrer Nützlichkeit durchgesetzt haben und jenen, die in so gut wie keiner Anwendung benötigt bzw. benutzt werden. Durch diesen Ansatz werden XML und EDI miteinander verbunden. In XML eingebettete EDI-Daten können über das Internet übertragen werden. XML/EDI wird aber noch durch drei weitere Komponenten ergänzt, die es erlauben, EC zwischen Unternehmen dynamisch zu gestalten. Die Werkzeuge der einzelnen Komponenten unterstützen sich gegenseitig. EDI ist nun nicht mehr durch seine fixierte Struktur eingeschränkt und kann vielseitig angepaßt werden. XML/EDI ist eine Verschmelzung von fünf komplementären Technologien. Die Stärke dieses

Ansatzes liegt in den Kombinationsmöglichkeiten [EDI]. Die Komponenten von XML/EDI werden nun näher betrachtet, vgl. Abb. 8:

- In **XML** als Rahmentechnologie werden die Technologien eingebettet. Aufbauend auf der XML Syntax (Tags und Frameworks) können die anderen Komponenten übers Netz übertragen werden. Segmentbezeichner von EDI werden durch XML-Tags ersetzt bzw. erweitert.
- **EDI** als Urvater des EC bietet die Möglichkeit, Daten in ein (einfaches) Format zu transformieren, um diese dann über das Netz zu senden. Der Ansatz von XML/EDI bietet eine 100%-ige Rückwärts-Kompatibilität, um existierende EDI-Systeme auch weiterhin zu benutzen. Sowohl auf vorhandenen Investitionen in EDI Systemen als auch auf Kenntnissen kann weiter aufgebaut werden.
- **Templates** halten die Komponenten zusammen. Ohne solche Schablonen wäre es in XML nicht möglich, alle notwendigen Details auszudrücken. Templates ähneln Prozeßbeschreibungssprachen, sie beschreiben, was mit den Daten passiert. Auf Templates wird referenziert, oder sie werden in einem XML Dokument in einem Abschnitt als spezielle Menge von Token übertragen. Ergänzt werden Templates von DTD, damit durch die Definition von Struktur und Inhalt eine Interoperabilität von Transaktionen erreicht wird.
- **Repositories** sind öffentlich nutzbare Verzeichnisse im Internet (Shared Internet Dictionaries). Sie ermöglichen dem Anwender die Bedeutung und Definition von EDI-Elementen nachzuschlagen.
- **Agenten** interpretieren Templates, interagieren mit dem Anwendungssystemen bzw. der Transaktion, um neue Templates für spezielle Aufgaben zu generieren. Agenten können auch Templates den Transaktionen hinzufügen indem sie auf globale Repositories zugreifen und auf vorhandene Templates referenzieren.

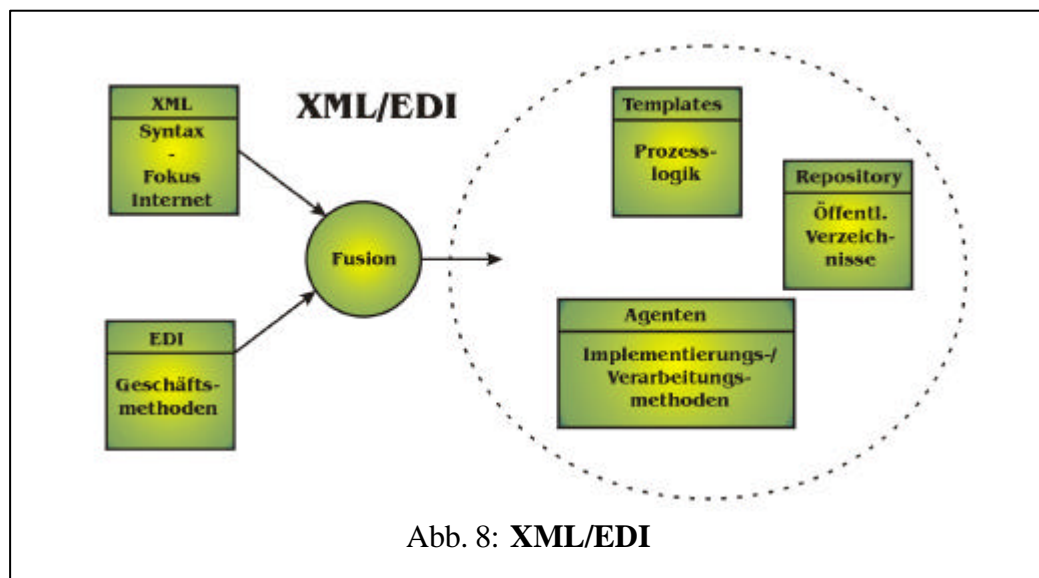


Abb. 8: XML/EDI

Das XML/EDI Framework stellt einen dynamischen Prozeß zum Senden von Nachrichten dar, der nahezu uneingeschränkt erweitert werden kann. XML/EDI baut auf dem gesammelten Wissen der letzten 20 Jahren auf.

XML seinerseits gibt die Möglichkeit Daten als Objekte ineinander einzubetten, miteinander zu verketten und schließlich zu versenden. XML erleichtert auch die Suche, so daß mehr relevante Ergebnisse geliefert werden. Des weiteren ist XML auf offenen Standards aufgebaut, Möglichkeiten für selbstbeschreibende Transaktionen sind gegeben. Klassische Austauschverfahren werden vielmehr ergänzt, da XML nicht in Konkurrenz zu XML/EDI steht. Die Abstimmung von Dokumentinhalten zwischen den Austauschpartnern wird im Gegensatz zu klassischen EDI-Lösungen bei diesem Ansatz wesentlich vereinfacht, da DTDs in zentralen Repositories aufbewahrt werden.

5. Ausblick

Wie so oft gibt es nicht die Lösung um auf elektronischem Wege Daten zwischen Unternehmen auszutauschen. XML ist heutzutage in aller Munde und wird sehr gelobt. In der Erweiterbarkeit liegen einerseits die Stärken aber auch die Schwächen von XML. Es muß ein Mittelweg zwischen der Anzahl der XML-Schema und der Komplexität dieser gefunden werden. Die Lesbarkeit und die einfache Benutzung von XML lassen mehr und mehr auch Endkonsumenten aktiv am EC teilnehmen. Im B2B-Bereich wurde der Datenaustausch durch XML im Vergleich zu den vorhandenen Lösungen SGML und EDI für die Anwender in der Benutzung vereinfacht.

6. Literaturreferenzen und Online-Ressourcen:

- [SGML]: Overview of SGML Resources : <http://www.w3.org/MarkUp/SGML/>
- [SEANMcG]: XML By Example, Building E-Commerce Applications,
Sean McGrath, 1998
- [XSL]: Extensible Stylesheet Language : <http://www.w3.org/Style/XSL/>
- [XLINK]: XML Linking Language (Xlink):
<http://www.w3.org/TR/1998/WD-xlink-19980303>
- [XMLPP]: XML E-Business Standards: Promises and Pitfalls, 5.1. 2000:
<http://www.xml.com/pub/a/2000/01/ebusiness/>
- [PXML]: Perspectives of XML in E-Commerce:
<http://tech.irt.org/articles/js215/XML.com>
- [EDI-CHANGE]: edi-change – Die neuen Edi-Wege, Magazin für Electronic Commerce
ISSN 0947-8493
- [BIZTALK]: BizTalk Framwork 2.0 Draft:
<http://msdn.microsoft.com/xml/articles/biztalk/biztalkfww2draft.asp>
- [MTL99]: Merz, Tu, Lamersdorf, Informatik Spektrum, Band 22-Heft 5, 22.10.99
- [EDI]: XML-EDI Frameworks:
<http://www.geocities.com/WallStreet/Floor/5815/xmlexec.htm>
- [XMLORG]: <http://www.xml.org>
- [MAXELON]: XML vs. EDI: <http://www.maxelon.co.uk/technology/xml.html>