Grundstudium

Konverter für XML-basierte Geschäftsdokumente

Prof. Dr.-Ing. Frank-Dieter Dorloff / Dipl.-Wirt.-Inform. Jörg Leukel / Dipl.-Inform. Volker Schmitz, Essen

Im zwischenbetrieblichen Datenaustausch werden oft Geschäftsdokumente mit verschiedenen Formaten verarbeitet, obwohl sie durch die gemeinsame Formatbasis XML standardisiert sind. Flexible Konverter ermöglichen eine verlustfreie und automatische Umwandlung.

1. Verarbeitung XML-basierter Geschäftsdokumente

Dokumente importieren, erzeugen und umwandeln

Mit der zunehmenden Nutzung des XML-Standards für den zwischenbetrieblichen Datenaustausch stehen viele Unternehmen vor der Aufgabe, XML-basierte Geschäftsdokumente unterschiedlichster Ausprägung zu verarbeiten und mit Lieferanten und Kunden in definierten Formaten auszutauschen. Dies erfordert unter anderem, dass die betrieblichen Informationssysteme in der Lage sind,

1. eingehende XML-Dokumente zu lesen und deren Daten für die Weiterverarbeitung zu importieren,
2. ausgehende XML-Dokumente zu erzeugen und mit Daten aus betrieblichen Datenbeständen zu befüllen und
3. bei Bedarf XML-Dokumente direkt in andere XML-Dokumente umzuwandeln.

Die Verarbeitung XML-basierter Geschäftsdokumente lässt sich vereinfacht als ein **Formatkonvertierungsproblem** darstellen. Die Konvertierung überführt ein Dokument von einem Format in ein anderes. Bezogen auf die genannten Anforderungen lassen sich drei Fälle unterscheiden. Für den Import und Export ist das jeweilige Schema des importierenden bzw. des exportierenden Informationssystems heranzuziehen (siehe Tab. 1), wohingegen bei der direkten Konvertierung das Quell- und Zielformat durch die den Dokumenten zu Grunde liegenden Schemata beschrieben werden.

Fall	Quellformat	Zielformat
Datenimport	Schema des zu verarbeitenden Dokuments	Schema des importierenden Informationssystems
Datenexport	Schema des exportierenden Informationssystems	Schema des zu erzeugenden Dokuments
Direkte Konvertierung	Schema des zu verarbeitenden Dokuments	Schema des zu erzeugenden Dokuments

Tab. 1: Konvertierung von Geschäftsdokumenten

Die Entwicklung von Konvertern für XML-basierte Geschäftsdokumente beruht auf der Analyse der **Schemata von Quell- und Zielformat**. Dazu werden Beziehungen zwischen inhaltlich korrespondierenden Datenelementen aufgedeckt und die Art der Beziehung definiert. Dieser Vorgang wird allgemein als **Mapping** bezeichnet, das im Ergebnis zu einzelnen Mapping-Definitionen führt. Eine Mapping-Definition beschreibt in einer strukturierten Form den Zusammenhang zwischen einem oder mehreren Datenelementen des Quell- und Zielformats.

Alternativen für Konvertersysteme

Die Rolle des Informationssystems, das eingangsseitig XML-basierte Geschäftsdokumente importiert und diese ausgangsseitig in veränderter Form wiederum in XML-basierte Geschäftsdokumente überführt, entspricht der eines Intermediärformats. Dadurch ent-

steht eine **Sternstruktur**, die bei zunehmender Anzahl von Quell- und Zielformaten die Anzahl der zu erstellenden Mappings (Konverter) minimiert. Im Gegensatz dazu erfordert die direkte Konvertierung zwischen jeweils zwei Formaten bereits bei drei Formaten eine höhere Anzahl von Konvertern, sodass eine vermaschte Netzstruktur entsteht. Alternativ lässt sich analog zu den Grundtypen von Netzwerktopologien die Konvertierungsaufgabe auch durch eine **Ringstruktur** lösen, die ein Dokument solange in einer vordefinierten Richtung an die Ringknoten weitergibt, bis das Zielformat erreicht ist. Voraussetzung dazu ist, dass beim Durchlauf im Ring keine Informationsverluste auftreten (vgl. Wüstner/Hotzel/Buxmann 2002).

Bestimmungsgrößen für die Konverterstruktur sind die Anzahl der Formate differenziert nach Quell- und Zielformaten, die Kosten für die Erstellung jedes Konverters, die Kosten je Konvertierung und die Anzahl der auszuführenden Konvertierungen je Periode. Darüber hinaus sind auch Varianten der Stern-, Netz- und Ringstrukturen zu berücksichtigen. In einer Sternstruktur ist das Mapping zwischen zu verarbeitendem Format und Intermediärformat nur dann in beiden Richtungen zu definieren, wenn das zu verarbeitende Format sowohl Quell- als auch Zielformat ist. Analog dazu wird die Netzstruktur vollständig vermascht sein, wenn alle Formate zugleich Quell- und Zielformat sind. Die Ringstruktur lässt sich dadurch verändern, dass die Konvertierung nicht mehr nur in einer Richtung, sondern auch in der umgekehrten Richtung ermöglicht wird. Bei einer relativ hohen Knotenanzahl können abweichend von dem Ring zusätzliche Verbindungen eingefügt werden, sodass im Ergebnis wieder eine Netzstruktur entsteht.

Frage 1: Welche Konverterstrukturen sind in Anlehnung an Netzwerktopologien möglich?

Darstellung ändern, Informationsgehalt erhalten

Unabhängig von der Frage, welche Konverterstruktur zu wählen ist, lässt sich die Dokumentkonvertierung auf gerichtete **Datentransformationen** zurückführen, die an den Kanten der Konverterstruktur ansetzen und die unidirektionale Konvertierung zwischen zwei Formaten zum Gegenstand haben. Unter einer Datentransformation wird die Umwandlung von Daten, die in einem Quelldatenformat vorliegen, in ein Zieldatenformat verstanden. Die Umwandlung sollte den Informationsgehalt der Daten nicht verändern oder verringern, sondern lediglich die Darstellung der Informationen dem jeweiligen Zieldatenformat anpassen. Bei der Entwicklung und der Definition solcher Transformationen sind eine Reihe von Problemen zu strukturieren und zu beschreiben, um daraus die Anforderungen an eine Formatkonvertierung ableiten zu können.

2. Arten des Mapping

In Abhängigkeit von der Anzahl der Datenelemente, die im Quell- und Zielformat an einer Mapping-Definition beteiligt sind, lassen sich die **vier Grundtypen** 1:1-, 1:N-, N:1- und N:M-Mapping bilden (vgl. Dorloff/Leukel/Schmitz 2004):

Grundtypen des Mapping basieren auf Kardinalität

- Beim 1:1-Mapping besteht eine Beziehung zwischen genau einem Datenelement des Quellformats zu genau einem Datenelement des Zielformats. Die zugehörige Datentransformation bildet den Informationsgehalt des Quell- auf das Zieldatenelement ab.
- Beim 1:N-Mapping besteht eine Beziehung zwischen genau einem Datenelement des Quellformats zu mehreren Datenelementen des Zielformats. Die zugehörige Datentransformation bildet den Informationsgehalt des Quelldatenelementes durch **Verteilung und Hinzufügung** auf die Zieldatenelemente ab. Dieser Fall tritt vor allem dann auf, wenn das Zielformat einen höheren Detaillierungsgrad aufweist.
- Beim N:1-Mapping besteht eine Beziehung zwischen mehreren Datenelementen des Quellformats zu genau einem Datenelement des Zielformats. Die zugehörige Datentransformation muss den Informationsgehalt der Quelldatenelemente durch **Konkatenation, Aggregation und Selektion** auf das Zieldatenelement abbilden. Komplementär zum 1:N-Mapping tritt dieser Fall vor allem auf, wenn das Quellformat einen höheren Detaillierungsgrad als das Zielformat besitzt.
- Beim N:M-Mapping besteht eine Beziehung zwischen mehreren Datenelementen des Quellformats zu mehreren Datenelementen des Zielformats. Die zugehörige Datentransformation bildet den Informationsgehalt der Quelldatenelemente durch eine **komplexe Operation**, die sich aus Teiloperationen vom Typ Verteilung, Hinzufügung, Konkatenation, Selektion und Aggregation zusammensetzt, auf die Zieldatenelemente ab. Dies ist der Fall, wenn die beiden Formate (in Teilbereichen) eine stark abweichende Dokumentstruktur aufweisen.

Frage 2: Wie sind Operationen vom Typ Verteilung, Hinzufügung, Konkatenation, Aggregation und Selektion zu interpretieren?

3. Weitere Anforderungen an die Konvertierung

Mapping-Kardinalitäten sind nicht ausreichend

Mit Hilfe der vier Mapping-Typen können bereits viele Zusammenhänge zwischen Quell- und Zielformat beschrieben werden. Darüber hinaus entstehen aufgrund der Komplexität von Geschäftsdokumenten und des Umfangs ihrer Schemata weitere Anforderungen an die Konvertierung. So sind partitionierte Geschäftsdokumente, unterschiedliche Kodierungen von Datenwerten und unterschiedliche Informationsgehalte zu berücksichtigen. Insbesondere die korrekte Umsetzung unterschiedlicher Informationsgehalte ist wichtig für die erfolgreiche Konvertierung von XML-Geschäftsdokumenten. Sie ist dann erreicht, wenn die transformierten Geschäftsdokumente syntaktisch korrekt sind und auch inhaltlich den Geschäftsprozessregeln entsprechen.

3.1. Partitionierung von Geschäftsdokumenten

Ein N:M-Mapping ist im Allgemeinen dann notwendig, wenn sich die Schemata von Quell- und Zielformat grundlegend unterscheiden. Dies gilt insbesondere für die **Verarbeitung partitionierter, CSV-basierter Formate**, die beispielsweise Katalogdaten auf mehrere, unterschiedlich strukturierte CSV-Dateien verteilen und Produkte, Preise und Merkmale voneinander trennen. Die Ursache liegt in den im Vergleich zu XML-Dokumenten nur sehr geringen Fähigkeiten des CSV-Formats, komplexe Datenstrukturen — insbesondere 1:N- und N:M-Assoziationen — abzubilden. Die Partitionierung erfordert, dass beim Mapping der Zusammenhang der einzelnen Teilformate erkannt und berücksichtigt wird, d.h., die in den Teilformaten enthaltenen Referenzen auf Identifikatoren der jeweiligen Objekttypen (insbesondere Produktidentifikatoren) werden je nach Beziehungskardinalität für die Definition des Mapping benötigt. Ein Beispiel soll dies verdeutlichen (s. Abb. 1).

```
Quellformat 1:
        <Produkt>
          <ID>100</ID>
          <Bezeichnung>Ringschlüssel 8x10</Bezeichnung>
          ...
        </Produkt>

Quellformat 2:
        <Produktmerkmal>
          <ProduktID>100</ProduktID>
          <Bezeichnung>Schlüsselweite 1</Bezeichnung>
                <Wert>8</Wert>
        </Produktmerkmal>
        <Produktmerkmal>
          <ProduktID>100</ProduktID>
          <Bezeichnung>Schlüsselweite 2</Bezeichnung>
                <Wert>10</Wert>
        </Produktmerkmal>

Zielformat BMEcat 1.2:
        <ARTICLE mode="new">
          <SUPPLIER_AID>100</SUPPLIER_AID>
          <ARTICLE_DETAILS>
            <DESCRIPTION_SHORT>Ringschlüssel 8x10</DESCRIPTION_SHORT>
            ...
          </ARTICLE_DETAILS>
          <ARTICLE_FEATURES>
            <FEATURE>
              <FNAME>Schlüsselweite 1</FNAME>
              <FVALUE>8</FVALUE>
            </FEATURE>
            <FEATURE>
              <FNAME>Schlüsselweite 2</FNAME>
              <FVALUE>10</FVALUE>
            </FEATURE>
          </ARTICLE_FEATURES>
          ...
        </ARTICLE>
```

Abb. 1: Beispiel XML-Daten für partitioniertes Quellformat

CSV-Format repräsentiert durch XML-Format

Als Quellformat liegt ein partitioniertes CSV-Format vor, das zur Beschreibung von Produkten eines elektronisches Kataloges die Produktmerkmale von den weiteren Produktdaten separiert und daher in einer zweiten Datei ablegt (1:N-Assozation zwischen Produkten und Merkmalen). Im Beispiel sind beide CSV-Dateien bereits in eine XML-Repräsentation überführt worden, die die Spaltenbezeichner des CSV-Formats für die Benen-

nung der XML-Tags übernimmt. Als Zielformat dient hier BMEcat (vgl. Schmitz/Kelkar/Pastoors 2001). Der umgekehrte Fall, also die Konvertierung eines XML-Geschäftsdokuments in mehrere, partitionierte Dokumente kann ebenfalls auftreten.

3.2. Unterschiedliche Datentypen

Sind bei einer Datentransformation Datenwerte zu verändern, so beruhen diese Modifikationen auf unterschiedlichen Datentypen der korrespondierenden Datenelemente in Quell- und Zielformat. Im einfachsten Fall des 1:1-Mapping sind beide Datenelemente gleich bezeichnet und unterscheiden sich ausschließlich in der **Kodierung der Datenwerte**.

Datentypen sind zu konvertieren

Kodierungsunterschiede werden vorwiegend durch **Zeichenkettenoperationen** und bei numerischen Werten durch **Berechnungsvorschriften** überbrückt. Sie reichen von einfachen, allgemeingültigen Ausdrücken bis zu komplexen Regeln, die in Abhängigkeit von den Datenwerten verschiedene Operationen anwenden oder sich aus einer Folge von Operationen zusammensetzen. Folglich sind die Mapping-Definitionen um solche Operationen zu ergänzen, d.h., die Zuordnung korrespondierender Datenelemente allein ist nicht ausreichend. Vielmehr sind die Operationen zu formalisieren, zum Beispiel durch Formeln oder in Anlehnung an prozedurale Programmiersprachen durch Pseudo-Code.

Neben der Art und Weise der formalisierten Beschreibung sind gerade für XML-basierte Geschäftsdokumente die verwendeten Datentypen und ihr Standardisierungsgrad von Bedeutung. Erstens werden beim bevorzugten Austausch von standardisierten XML-Geschäftsdokumenten die Datentypen für jedes Datenelement durch den jeweiligen Standard bereits vorgegeben und sind bei der Erstellung von Dokumenten einzuhalten. Zweitens greifen diese Dokumentstandards selbst auf andere, häufig international etablierte Standarddatentypen zurück. Dies gilt einerseits für die **Basisdatentypen für Ganzzahlen, Fließkommazahlen, Zeichenketten, Boolesche Werte und Zeitangaben**. Andererseits zeigt sich die Wiederverwendung vorhandener Standards bei den so genannten Aufzählungsdatentypen oder Enumerationen.

Viele Enumerationen sind standardisiert

Aufzählungsdatentypen gehören nicht zu der Gruppe der Basisdatentypen, da sie eine individuell festzulegende Menge von zulässigen Zeichenketten enthalten. Sie werden unter anderem für die einheitliche Kodierung von Sprachen, Ländern, Regionen, Währungen und Einheiten verwendet. Einen Überblick über wichtige, internationale Datentypstandards in diesem Bereich gibt die Tab. 2. Allen diesen Standards ist gemein, dass sie nicht spezifisch für E-Business-Zwecke, sondern generell für die Geschäftskommunikation entwickelt worden sind.

Standard	Objekte	Code	Beispiele	
ISO 639-2	Sprachen	2 Zeichen	de en ru	German English Russian
ISO 3166-1	Länder	2 Zeichen	DE GB US	Germany United Kingdom United States
ISO 3166-2	Regionen	2/3 Zeichen	DE-BW DK-025 FR-75	Federal State of Baden-Württemberg Roskilde Metropolitan Department Paris
ISO 4217	Währungen	3 Zeichen	GBP EUR USD	Pound Sterling Euro US Dollar
UN/LOCODE 2002-2	Lokationen	2/3 Zeichen	DE AWR DE ROR DE SXF	Achterwehr Ringkanal Duisburg-Ruhrort Berlin-Schönefeld Apt
UN/ECE Trade Facilitation Recommendation 20	Einheiten	2 bis 3 Zeichen	C62 MTR VLT	One, Piece, Unit Meter Volt

Tab. 2: International standardisierte Aufzählungsdatentypen

Neben standardisierten Enumerationen sind häufig weitere Aufzählungsdatentypen notwendig. Sie dienen dazu, **Sachverhalte kurz und prägnant zu beschreiben**, ohne die Semantik der Beschreibung vollständig über zugehörige Datenelemente explizit angeben zu müssen. Die Aufgabe dieser Enumerationen ist also die Charakterisierung oder Typisierung von Sachverhalten durch festgelegte Aussagen. Die Semantik der zulässigen Aufzählungswerte wird in dem Geschäftsdokumentstandard festgelegt. Im Geschäftsdokument selbst genügt anschließend die Angabe eines dieser Werte, um die Bedeutung für den Dokumentempfänger eindeutig zu bestimmen.

Mapping bei unterschiedlichen Datentypen

Weichen im Quell- und Zielformat die Datentypen derart voneinander ab, dass sie unterschiedliche Aufzählungen erlauben, so sind diese durch geeignete Datentransformationen umzusetzen. Im proprietären Quellformat seien zum Beispiel die Landesangaben nicht gemäß ISO 3166-1 kodiert, sondern als Klartext angegeben („Deutschland" vs. „DE"). Dann ist ein Mapping zwischen den Aufzählungswerten des Quelldatentyps zu den Aufzählungswerten des Zieldatentyps zu erstellen. Ein solches Mapping wird auch durch so genannte **Mapping- oder Referenztabellen** formalisiert, die die jeweiligen Zuordnungen enthalten. Neben den direkten Zuordnungen, die dem 1:1-Mapping entsprechen, sind auch die weiteren Fälle zu berücksichtigen, wenn beispielsweise im Zielformat weniger oder anders segmentierte Preistypen als im BMEcat-Standard vorgegeben sind. Folglich lassen sich die vier Grundtypen, die auf die Anzahl der beteiligten Elemente am Mapping abstellen, auf das Mapping von Aufzählungsdatentypen übertragen. Anstelle von Datenelementen werden jedoch Datenwerte in Beziehung gesetzt.

Frage 3: Gegeben sei ein Aufzählungsdatentyp für Farben mit den Werten {kobaltblau, ultramarinblau, grüngelb, lila, rot}. Wie könnte ein zweiter Aufzählungsdatentyp aussehen, auf den die Grundtypen des Mapping übertragen werden können?

3.3. Unterschiedlicher Informationsgehalt

Ist die Konvertierung überhaupt möglich?

Die bisherigen Ausführungen sind davon ausgegangen, dass sich Quell- und Zielformat nur strukturell unterscheiden, der durch die jeweiligen Schemata abgebildete Informationsgehalt jedoch weitgehend identisch ist. Zumindest sind beide Schemata so angelegt, dass prinzipiell für alle Datenelemente Mapping-Definitionen entwickelt werden können. Von der Erfüllung dieser Voraussetzungen kann in der Praxis nicht immer ausgegangen werden. Im Gegenteil ist für die meisten proprietären und standardisierten Austauschformate für Geschäftsdokumente kennzeichnend, dass sie **bestimmte Sachverhalte nicht abdecken** und **anwendungs- oder branchenspezifische Sachverhalte berücksichtigen**, die sich wiederum in anderen Austauschformaten nicht wieder finden. Dies gilt für die zahlreichen XML-basierten Standardformate und insbesondere für standardisierte Katalogaustauschformate (vgl. Beul et al. 2003).

Wenn sich Quell- und Zielformat in ihrem Informationsgehalt unterscheiden, dann können bei der Konvertierung von Dokumenten relevante **Informationsverluste und Informationsdefizite** auftreten. Im ungünstigsten Fall wird die automatische Konvertierung sogar verhindert oder ist nur teilweise möglich. Daher ist es notwendig, ergänzend zu der Mapping-Definition den Informationsgehalt der Formate zu untersuchen und vergleichend gegenüberzustellen. Da der Informationsgehalt durch Datenelemente ausgedrückt wird, ist zu fragen, ob korrespondierende Datenelemente vorhanden sind und ob es sich um Pflicht- oder optionale Datenelemente handelt. Gemäß diesem Differenzierungsmerkmal entstehen die in Tab. 3 dargestellten Fälle.

		Datenelemente im Zielformat		
		Pflicht	**Optional**	**Nicht vorhanden**
Datenelemente im Quellformat	**Pflicht**	Mapping	Mapping	Informationsverlust
	Optional	Informationsdefizit	Mapping	Informationsverlust
	Nicht vorhanden	Informationsdefizit	—	—

Tab. 3: Typisierung von Datentransformationen nach dem Informationsgehalt

Aus dieser Typisierung lassen sich drei Aussagen ableiten:

1. Die Fälle, in denen im Quellformat enthaltene Informationen ebenfalls Bestandteil des Zielformats sind, werden vollständig durch Mapping-Definitionen abgedeckt.
2. Ein Informationsverlust entsteht, wenn Informationen des Quellformats nicht durch das Zielformat wiedergegeben werden können. Besonders gravierend ist der Verlust dann, wenn es sich um Pflichtinformationen bzw. **Pflichtdatenelemente** handelt. Hier ist eine **Konvertierung nicht mehr sinnvoll**, da bereits Basisinformationen verloren gehen. Weniger schwer wiegt der Verlust bei optionalen Informationen, da wenigstens die Pflichtbestandteile des Quellformats verlustfrei transformiert werden können. Die Konvertierung muss dabei akzeptieren, dass das Zielformat weniger mächtig als das Quellformat ist.
3. Ein Informationsdefizit liegt vor, wenn Pflichtelemente des Zielformats nur optional oder überhaupt nicht im Quellformat vorhanden sind. Um dennoch eine Dokument-

konvertierung zu ermöglichen, ist es notwendig, die fehlenden Informationen in den Transformationsprozess zu integrieren. Dazu ist in diesen Prozess einzugreifen, indem die benötigten Informationen manuell hinzugefügt werden.

Informationsverluste bei der Konvertierung von XML-Geschäftsdokumenten können **verschiedene Ursachen** haben, die sich an dem Zielformat festmachen lassen.

Offensichtlicher Informationsverlust

— **Fehlende Datenelemente**: Im Zielformat findet sich kein korrespondierendes Datenelement, das den Informationsgehalt des Quelldatenelementes wiedergeben könnte. Zum Beispiel unterscheiden sich die Schemata für Produktpreise in den Katalogformaten BMEcat und cXML erheblich (vgl. Ariba 2004). Während in cXML nur Endpreise angegeben werden können, umfasst das BMEcat-Preismodell auch Mengenstaffeln, einen Rabattfaktor, die Kennzeichnung als Tagespreis und liefergebietsspezifische Preisdifferenzierungen. Alle diese Informationen gehen bei einer Konvertierung von BMEcat-Katalogdokumenten zu cXML verloren.

Spezifisch für das XML-Datenmodell

— **Geringere Datenelementkardinalität**: Im XML-Datenmodell besitzen alle Datenelemente eine definierte Kardinalität, die ausdrückt, mit welcher Häufigkeit das Datenelement in Dokumenten auftreten darf. Die Min-Kardinalität bestimmt die Mindestanzahl, die Max-Kardinalität die höchste Anzahl (z.B. bedeutet 0..1, das Element kann einmal oder keinmal auftreten; 1..N bedeutet, das Element muss mindestens einmal und kann mehrfach auftreten). Im BMEcat- vs. cXML-Beispiel tritt dieser Informationsverlust wiederum bei den Produktpreisen auf: Während in cXML jedes Produkt nur einen Preis besitzt — das Datenelement UnitPrice besitzt die Kardinalität 1..1 — können mit dem BMEcat-Format mehrere Preise je Produkt übertragen werden (Datenelement ARTICLE_PRICE mit Kardinalität 1..N). Hier zeigt sich exemplarisch, wie die Berücksichtigung des Informationsgehaltes die Mapping-Definition erweitert. Das N:1-Mapping zwischen ARTICLE_PRICE und UnitPrice lässt sich korrekt definieren, es tritt jedoch möglicherweise ein Informationsverlust auf, und zwar dann, wenn in dem BMEcat-Katalogdokument Produkte mit mehreren Preisen enthalten sind.

Alle Datentypen und Wertebereich sind zu überprüfen

— **Unzureichender Wertebereich**: Informationsverluste können auch aufgrund unterschiedlicher Wertebereiche korrespondierender Datenelemente entstehen. In diesem Fall ist der Wertebereich im Zielformat unzureichend, d.h., der Wertebereich im Quellformat ist größer. In Abhängigkeit von der Art des Datentyps zeigt sich dies in Form von
  - Längenbeschränkungen für Zeichenketten, die bei der Konvertierung möglicherweise abzuschneiden sind,
  - geringer Genauigkeit von numerischen Werten (Vor- und Nachkommastellen),
  - geringer Genauigkeit von Zeitangaben, beispielsweise für Lieferzeiten, die im Quellformat bis auf Stunden genau angegeben werden können, im Zielformat jedoch nur in ganzen Tagen, und
  - unzureichenden Aufzählungsdatentypen; beispielsweise werden Bestelleinheiten im Quellformat gemäß dem umfassenden Standard UN/ECE Trade Facilitation Recommendation 20 kodiert, während das Zielformat nur eine stark eingeschränkte Untermenge dieses Standards verwendet.

Informationsdefizite sind kritisch

Informationsverluste können unter Umständen die Konvertierung von XML-Geschäftsdokumenten als nicht sinnvoll erscheinen lassen und damit verhindern, da wesentliche, für den Geschäftsprozess notwendige Informationen verloren gehen. In diesem Fall ist das Zielformat nicht geeignet, die gestellten inhaltlichen Anforderungen zu erfüllen. Demgegenüber erschweren oder verhindern Informationsdefizite die Konvertierung aus syntaktischen Gründen: Die im Quelldokument enthaltenen Informationen sind nicht ausreichend, um automatisiert ein gültiges Zieldokument zu erstellen.

Die **Gültigkeit eines XML-Dokuments** ist dann gegeben, wenn das Dokument seiner formalen Spezifikation genügt. Die formale Spezifikation beschreibt dabei den Aufbau gültiger Dokumente vergleichbar mit dem Schema einer relationalen Datenbank. Dazu zählen die Definition von Datentypen und Datenelementen, die Verwendung dieser Elemente zur Beschreibung der hierarchischen Dokumentstruktur, die Hinzufügung von Integritätsbedingungen und je nach eingesetzter Schemasprache weitere Eigenschaften des Dokumenttyps. Die wichtigsten Schemasprachen für XML-Dokumente sind **DTD (Document Type Definition)** und **XSDL (Extensible Schema Definition Language)**. Letztere ist nicht zuletzt aufgrund ihrer Mächtigkeit zur Standardsprache für die Spezifikation von XML-Geschäftsdokumenten geworden (vgl. Schmitz/Leukel/Dorloff 2003).

Manueller Eingriff — Aufwand vertretbar?

Im Kontext der Datentransformationen muss sichergestellt werden, dass die erstellten Dokumente mindestens alle Pflichtdatenelemente des Zielformats aufweisen, da ansonsten gegen eine zentrale Bedingung der Gültigkeit verstoßen wird. Lässt sich nun für

ein Pflichtdatenelement kein korrespondierendes Element im Quellformat identifizieren, so ist eine valide Konvertierung nicht ohne weiteres möglich. Die Lösung kann darin bestehen, die fehlenden Informationen nicht formal aus dem Quelldokument abzuleiten, sondern vor der Ausführung der Datentransformation manuell hinzuzufügen. Im Beispiel der Katalogdaten ist diese Hinzufügung gerade dann Erfolg versprechend, wenn sie nur einmal zu Beginn der Konvertierung und nicht für jeden Produktdatensatz vorzunehmen ist. Im letzteren Fall wäre der manuelle Aufwand bei Hunderten oder Tausenden von Produkten wirtschaftlich kaum zu vertreten.

Frage 4: Welche weiteren Faktoren könnten die Konvertierung verhindern oder erschweren?

4. Entwicklung von Datentransformationen

Das Erarbeiten von Mapping-Definitionen und das Feststellen von Informationsverlusten und -defiziten basiert auf der **Analyse der formalen und deskriptiven Formatspezifikationen**, soweit diese verfügbar und hinreichend sind. Zunächst muss für jedes Datenelement überprüft werden, welche Informationen es enthält und ob diese im Zieldokument relevant sind. Die semantische Zuordnung erfordert daher eine genaue Kenntnis der Dokumenttypen. Zu vielen Formaten kann die zugehörige Spezifikation bei der Klärung von Elementinhalten helfen. Die Bedeutung ist jedoch oft nur mit fachlichem Wissen zweifelsfrei feststellbar, d.h., es ist profundes **Wissen über die jeweilige Domäne** notwendig (z.B. E-Procurement, Logistik).

Entwicklung planen und dokumentieren

Mapping-Definitionen sind bei umfangreichen Dokumenttypen detailliert zu dokumentieren und nach den beschriebenen Kriterien zu klassifizieren. Dazu bieten sich der Aufbau und die schrittweise Vervollständigung von Mapping-Tabellen an, welche die Dokumentstruktur von Quell- und Zielformat gegenüberstellen. Zu den Inhalten dieser Beschreibung gehören bezüglich der Formate die Elementbezeichner nach ihrem Auftreten in der hierarchischen Dokumentstruktur, die Elementkardinalitäten und die Datentypen. Die Gegenüberstellung von Quell- und Zielformat erfolgt durch das Zuordnen korrespondierender Datenelemente, die Angabe der Mapping-Kardinalität und die Kennzeichnung, ob Informationsverluste bzw. -defizite auftreten. Je nach Ansatz und Bedeutung dieser geplanten Entwicklung von Datentransformationen kann so eine fundierte und vollständige Beschreibung entstehen, die anschließend auch von Nicht-Domänenexperten mittels geeigneter **Datentransformationswerkzeuge** (z.B. Microsoft BizTalk, Seeburger Business Integration Server) überführt oder in Skripte umgesetzt werden kann.

Die Ausführung von Datentransformationen für XML-Dokumente geschieht häufig auf der Grundlage von Skripten, die in der standardisierten Sprache **XSLT (Extensible Stylesheet Language Transformations)** erstellt sind. Gerade bei aufwändigen Transformationen, komplexen Dokumentstrukturen und einer Vielzahl von Formaten, die zudem in unterschiedlichen Versionen vorliegen, bietet es sich an, den Entwicklungsprozess an **Prinzipien der Softwareentwicklung** auszurichten. Im Speziellen ist die direkte Codierung in XSLT ohne eine ausreichende inhaltliche Dokumentation zu vermeiden. Üblicherweise wird dazu die konzeptionelle Entwicklung, wie skizziert, von der Codierung getrennt.

Werkzeuge entwickeln Mapping-Vorschläge

Ein alternatives Konzept zur manuellen Entwicklung von Datentransformationen ist die **Automatisierung**, die in Zusammenhang mit dem Schema-Matching in Datenbanken entstanden ist (vgl. Rahm/Bernstein 2001). Übertragen auf XML-Geschäftsdokumente lassen sich somit Verfahren einsetzen, die die formalen Spezifikationen, also die DTD- bzw. XSD-Dateien der Quell- und Zielformate analysieren, im Ergebnis korrespondierende Datenelemente zueinander in Beziehung setzen und XSLT-Anweisungen für die Konvertierung generieren. Der Erfolg solcher Verfahren hängt jedoch stark von der Qualität der formalen Spezifikationen, der Komplexität der Domäne und den inhaltlichen Überschneidungen von Quell- und Zielformat ab. Aus diesem Grund erfordert auch das automatisierte Schema-Matching die Hinzuziehung von Domänenexperten. Diese bringen ihr Wissen in den Automatisierungsprozess ein, indem z.B. eine Taxonomie von Datenelementen erstellt wird, Synonyme für Datenelemente benannt und Nebenbedingungen für Datenwerte definiert werden. Diese Schritte gehören zu den **Vorarbeiten des Schema-Matching**, die im Fall der Automatisierung noch um die Konfigurationsparameter der Automatisierungsverfahren zu ergänzen sind. Das erzeugte Schema-Matching ist eine Menge von Datentransformationen, die anschließend zu überprüfen und in der Regel iterativ zu modifizieren und zu ergänzen sind. Damit ist die Entwicklung von Datentransfor-

mationen ein Prozess, der die Schritte **Vorbereitung, Definition, Überprüfung und Modifikation** umfasst. Die Überprüfung bezieht sich dabei (1) auf die syntaktische Gültigkeit („Entspricht das Matching der zu Grunde liegenden Transformationssprache?"), (2) die Vollständigkeit („Werden alle Datenelemente des Quell- und Zielformats berücksichtigt?") und (3) die Korrektheit („Sind die Transformationen inhaltlich korrekt?").

Literaturempfehlungen:

Ariba, Inc.: cXML 1.2.009. Online: http://xml.cxml.org (Stand: 17.01.2004).

Beul, M./Bittscheidt, C./Leukel, J./Spies, T.: Behandlung von Informationsdefiziten und -verlusten bei der Transformation von XML-Geschäftsdaten. In: Proceedings der 5. Paderborner Frühjahrstagung „Innovationen im E-Business", Paderborn, 2003, S. 159 - 168.

Dorloff, F.-D./Leukel, J./Schmitz, V.: Datentransformation bei XML-basierten Geschäftsdokumenten. In: WISU , 33. Jg. (2004), S. 87 - 94.

Leukel, J./Schmitz, V./Dorloff, F.-D.: Coordination and Exchange of XML Catalog Data in B2B. In: Proceedings of the 5th International Conference on Electronic Commerce Research (ICECR-5), Montreal 2002.

Rahm, E./Bernstein, P.A.: A Survey of Approaches to Automatic Schema Matching. In: The VLDB Journal, Vol. 10 (2001), S. 334 - 350.

Schmitz, V./Kelkar, O./Pastoors, T.: Spezifikation BMEcat Version 1.2, 2001. Online: http://www.bmecat.org (Anmeldung erforderlich, Stand: 17.01.2004).

Schmitz, V./Leukel, J. Dorloff, F.-D.: Does B2B Data Exchange Tap the Full Potential of XML Schema Languages. In: Proceedings of the 16th Bled Electronic Commerce Conference, Bled 2003, S. 172 - 182.

Wüstner, E./Hotzel, T./Buxmann, P.: Converting Business Documents: A Classification of Problems and Solutions using XML/XSLT. In: Proceedings of the 4th IEEE International Workshop on Advanced Issues of E-Commerce and Web-based Information Systems (WECWIS 2002), Newport Beach 2002, S. 61 - 68.

Die Fragen werden im WISU-Repetitorium beantwortet.

Zieldatenelemente mit Werten belegt, die nicht explizit im Quelldokument enthalten sind. Die Konkatenation fügt die Werte mehrerer Quelldatenelemente zu einem Zieldatenelement zusammen, während die Selektionsoperation eine Auswahl vornimmt. Schließlich zeigt der Typ Aggregation an, dass für die Ermittlung der Zieldatenwerte eine komplexere Konstruktions- oder Berechnungsvorschrift anzuwenden ist.

Frage 3: Gegeben sei ein Aufzählungsdatentyp für Farben mit den Werten {kobaltblau, ultramarinblau, grüngelb, lila, rot}. Wie könnte ein zweiter Aufzählungsdatentyp aussehen, auf den die Grundtypen des Mapping übertragen werden können?

Die gegebenen Werte „kobaltblau" und „ultramarinblau" lassen sich auf den Zielwert „blau" abbilden, so dass ein N:1-Mapping vorliegt. Der umgekehrte Fall des 1:N-Mapping zeigt sich bei der Zuordnung des gegebenen Wertes „grüngelb" auf die Zielwerte „grün" und „gelb". Die Übernahme des gegebenen Wertes „rot" in den zweiten Aufzählungsdatentyp deckt den 1:1-Fall ab. Das gleiche gilt auch das Mapping von „lila" zu dem abweichenden Wert „violett" der zweiten Enumeration. Im Gegensatz zu dem Mapping zwischen Datenelementen ist der N:M-Fall für das Mapping zwischen Enumerationswerten nicht relevant.

Frage 4: Welche weiteren Faktoren könnten die Konvertierung verhindern oder erschweren?

Neben den beschriebenen Schemaunterschieden und ihrer Berücksichtigung bei der Entwicklung von Datentransformationen ist insbesondere die Ausführungsphase von Bedeutung. Die Verarbeitung komplexer, umfangreicher XML-Dokumente ist ein ressourcenintensiver Vorgang, der entsprechende Laufzeiten nach sich zieht und Kosten für die Bereitstellung einer Konvertierungsinfrastruktur verursacht (z.B. Serversysteme, Anzahl Prozessoren, Arbeitsspeicher). Weiterhin sind die Anforderungen an die zeitliche Dauer der Konvertierung zu überprüfen. Während die Verarbeitung von Stammdaten (z.B. elektronischer Produktkatalog) in der Regel weniger zeitkritisch ist, erfordert die Verarbeitung von Transaktionsdaten, zum Beispiel auf elektronischen Marktplätzen und in Just-in-Time-Systemen, sehr viel kürzere und häufig garantierte Verarbeitungszeiten.

Wirtschaftsinformatik/Grundstudium

Fragen und Antworten 1 - 4 zu „Konverter für XML-basierte Geschäftsdokumente" von Prof. Dr.-Ing. F.-D. Dorloff/Dipl.-Wirt.-Inform. J. Leukel/Dipl.-Inform. V. Schmitz. WISU 3/04, S. 341 - 348.

Frage 1: Welche Konverterstrukturen sind in Anlehnung an Netzwerktopologien möglich?

Die Netzwerktopologien Stern, Netz und Ring lassen sich wie folgt für die Beschreibung von Konverterstrukturen nutzen: Bei einer Sternstruktur übernimmt das im Zentrum stehende Intermediärformat eine integrierende Funktion, da die Konvertierung nicht direkt zwischen den Einzelformaten, sondern zunächst von dem Quellformat zu dem Intermediärformat und anschließend zu dem Zielformat erfolgt. Bei einer Netzstruktur bilden die Formate die gleichberechtigten Knoten des Netzes, in welchem über gerichtete Kanten die direkte Konvertierung ermöglicht wird. Bei einer Ringstruktur sind die Formate (Knoten) in einer festen Reihenfolge seriell angeordnet und zu konvertierende Dokumente werden solange zu dem nachfolgenden Format konvertiert, bis das Zielformat erreicht worden ist. Dagegen lassen sich Konverter nicht gemäß der Netzwerktopologie Bus organisieren.

Frage 2: Wie sind Operationen vom Typ Verteilung, Hinzufügung, Konkatenation, Aggregation und Selektion zu interpretieren?

Die Typen charakterisieren den Zusammenhang der an einem Mapping beteiligten Datenelemente genauer, indem sie beschreiben, wodurch die Werte der Zieldatenelemente entstehen. Bei einer Verteilung wird der Informationsgehalt eines Quelldatenelements auf mehrere Zieldatenelemente verteilt. Bei einer Hinzufügung werden

Wirtschaftsinformatik/Hauptstudium

Fragen und Antworten 1 - 4 zu „Model Driven Architecture" von Prof. Dr. R. Thome/Dipl.-Kfm. M. Böhn/Dipl.-Kfm. A. Hagn. WISU 3/04, S. 348 - 357.

Frage 1: Welche Parallelen bestehen zwischen der Programmerstellung und -ausführung mit Java und dem mehrstufigen Modell der Model Driven Architecture?

Die vor der Programmentwicklung durchzuführende Anforderungsanalyse entspricht der Fachkonzeption im CIM. Zur Modellierung der abzubildenden Problemstellung werden neutrale Diagramme, beispielsweise UML-Diagramme, verwendet. Die Entwicklung Java-basierter Applikationen basiert grundlegend auf plattformunabhängigem Quell-Code. Dieser wird mit Hilfe eines Java-Compilers in einen ebenfalls plattformunabhängigen Byte-Code transformiert. Erst die jeweilige plattformabhängige Virtual Machine besitzt alle relevanten Transformations-Vorschriften, um den vorliegenden Byte-Code in den zugehörigen Maschinen-Code umzusetzen. Die Trennung der unabhängigen Compilierung von der maschinenabhängigen Interpretierung ist eine der wichtigsten Eigenschaften von Java, da einmal geschriebener Source-Code universell auf allen Plattformen einsetzbar ist, für die eine Virtual Machine vorliegt („write once, run many"). Hier wurde das Konzept der Trennung von Programmlogik und plattformspezifischer Ausführung bereits umgesetzt.

Frage 2: Welcher Zusammenhang besteht zwischen den Modellierungssprachen MOF, UML und CWM?

Mit Hilfe der Unified Modeling Language (UML) werden insbesondere Anwendungsfälle, Klassen, Zustände und Aktivitäten des zu