Studiengang:	Informatik
Prüfer:	Prof. Dr.-Ing. habil. Bernhard Mitschang
Betreuer:	Dr.-Ing. Oliver Kelkar (Fraunhofer IAO) Dr. rer. nat. Holger Schwarz
begonnen am:	15.05.2003
beendet am:	14.11.2003
CR-Klassifikation:	C.2.4, D.1.7, D.2.12

Studienarbeit Nr. 1907

Integrationskonzept für zwischenbetriebliche Geschäftstransaktionen mit SAP R/3 auf Basis offener XML-Standards

Sven Sprandel

Institut für Parallele und Verteilte Systeme
Universitätsstr. 38
70569 Stuttgart

1 Inhaltsverzeichnis

1 Inhaltsverzeichnis ... 3

2 Zusammenfassung ... 5

3 Einleitung ... 6

4 Zielsetzung und Vorgehensweise der Arbeit ... 7

5 Stand der Technik ... 8

5.1 XML ... 8
5.1.1 Überblick ... 8
5.1.2 XSLT ... 8
5.1.3 XPath ... 9
5.1.4 XML Schema ... 9
5.2 openTRANS ... 10
5.3 Anwendungsystem SAP R/3 ... 11
5.3.1 SAP Business Framework ... 12
5.3.2 IDoc ... 17
5.3.3 SAP Business Connector ... 18

6 Anforderungen ... 24

6.1 System zu System Kommunikation ... 24
6.2 Standardkomponenten ... 25
6.3 Sicherheit ... 26
6.3.1 Verfahren ... 27
6.3.2 Bewertung ... 27

7 Umsetzung asynchroner Nachrichtenaustausch ... 29

7.1 Beschreibung der openTRANS-Integration ... 29
7.2 Systemkonzept ... 29
7.3 Realisierung mittels BAPIS ... 31
7.3.1 Senden einer Nachricht ... 32
7.3.2 Anlegen der Bestellung im SAP-System ... 32
7.3.3 Erzeugung der Bestellbestätigung ... 36
7.4 Realisierung mittels XSLT ... 38
7.4.1 Senden einer Nachricht ... 38
7.4.2 Transformation der Bestellung ... 38
7.4.3 Transformation der Bestellbestätigung ... 41
7.5 Bewertung ... 43

8 Umsetzung synchroner Nachrichtenaustausch ... 45

8.1 Beschreibung eines Preisanfrage-Systems ... 45
8.2 Systemkonzept ... 45
8.3 Realisierung ... 46
8.3.1 Senden einer Nachricht ... 46
8.3.2 Preisermittlung ... 46
8.4 Bewertung ... 47

9 Ergebnisse und Ausblick ... 48

10 Bibliografie ... 49

11 Anhang A1 openTRANS-BAPI-Zuordungen 51
12 Anhang A2 Zuordnungen IDOC-openTRANS 51
13 Anhang B1 openTRANS-IDoc XSLT Schablonen 52
14 Anhang B2 IDoc-openTRANS XSLT-Schablonen 55
15 Anhang C1 openTRANS-Zuordungen 58
16 Anhang D1 IDoc Nachricht ORDER02 (ASCII) 59
17 Anhang D2 IDoc Nachricht ORDERS (XML) 60
18 Anhang E1 openTRANS-Beispielnachrichten 62
18.1 ORDER (Bestellung) 62
18.2 ORDERRESPONSE (Bestellbestätigung) 64
18.3 RFQ (Angebotsanforderung) 66
18.4 QUOTATION (Angebot) 68

2 Zusammenfassung

Unternehmen sehen sich auf nationalen wie internationalen Märkten einem stärker wachsenden Druck im Hinblick auf Kostenreduktion, Prozessverkürzung und Flexibilität gegenüber.

Besonders durch stetig kürzer werdende Produktlebenszyklen stehen alle Bereiche eines Unternehmens der Notwendigkeit gegenüber, Prozesse in sehr enger Koordination miteinander durchzuführen. Diese Entwicklung wirkt sich auf sämtliche in der Wertschöpfungskette eines Produktes beteiligten Unternehmen aus.

Unternehmen benötigen leistungsfähige Methoden zur technischen Integration von Prozessen, auch über die eigenen Unternehmensgrenzen hinweg.

Das in der vorliegenden Arbeit entwickelte Integrationskonzept für zwischenbetriebliche Geschäftstransaktionen ermöglicht die vollständig automatisierte System zu System Kommunikation auf Basis offener XML-Standards zur Abwicklung von Bestellprozessen.

Die prototypische Realisierung der Integration von Geschäftstransaktionen auf Basis eines asynchronen und synchronem Nachrichtenaustauschs mit einem SAP R/3 System sowie eine abschließende Bewertung der verwendeten Technologie sind Teil der Arbeit.

3 Einleitung

Integration von zwischenbetrieblichen Geschäftstransaktionen heißt, die Systeme von Geschäftspartnern über offene Standards miteinander zu verbinden. Durch den durchgängigen Einsatz von E-Business-Lösungen in einer automatischen Kommunikation zwischen Systemen können, bedingt durch den Wegfall von manuellen Bearbeitungsschritten, hohe Kosteneinsparungen erzielt werden.

Für einen durchgängigen Einsatz sind Standards in der IT-Kommunikation unumgänglich. Die openTRANS-Initiative mit führenden deutschen und internationalen Unternehmen unter Leitung des Fraunhofer IAO, hat sich die Standardisierung von Geschäftsdokumenten zum Ziel gesetzt.

Um die volle Leistungsfähigkeit des von der Initiative entwickelten openTRANS-Standards zu entfalten, sind Konnektoren für verschiedenste Systeme notwendig, um eine Kommunikation zwischen Geschäftspartnern zu ermöglichen.

Das Warenwirtschafts- und Unternehmenssteuerungssystem SAP R/3 ist mit über 12.000 Installationen das in großen und mittelständischen Unternehmen am weitesten verbreitete Warenwirtschaftsystem der Welt.

Es besteht folglich ein Interesse an einer kostengünstigen, einfach zu implementierenden und auf Standard-Komponenten aufbauenden Lösung für die Integration des openTRANS-Standards in SAP R/3.

4 Zielsetzung und Vorgehensweise der Arbeit

Das Ziel der Arbeit ist die Entwicklung eines Integrationskonzepts für zwischenbetriebliche Geschäftstransaktionen mit SAP R/3 auf Basis offener XML-Standards. Das Ergebnis soll neben einer Anforderungsanalyse auch die prototypische Realisierung des Integrationskonzepts beinhalten.

In Folge dessen werden in Kapitel 5 zunächst die technischen Grundlagen des elektronischen Datenaustauschs aufgearbeitet und die technische Basis eines SAP-Systems beschrieben. Insbesondere wird der einem SAP-System beiliegende Business Connector im Hinblick auf Schnittstellen und Leistungsumfang als Integrationswerkzeug beleuchtet.

Anforderungen an eine System zu System Kommunikation, sowie Methoden und Aspekte der Integration unter der Gewährleistung von Sicherheitsaspekten werden im sechsten Kapitel dargelegt. Eine Bewertung der De-facto Sicherheitsstandards bildet in der späteren Realisierung die Entscheidungsgrundlage für die Auswahl der Sicherheitstechniken.

Das siebte Kapitel beschreibt ein in der Industrie häufig auftretendes Integrationsszenario für die asynchrone Übermittlung von Bestellinformationen und stellt zwei unterschiedliche technologische Varianten der Realisierung einer Integration vor, für die eine Bewertung erfolgt.

Im Anschluss wird eine Integration auf Basis eines synchronen Nachrichtenaustauschs anhand eines Online-Preisanfrage-Systems vorgestellt. Eine prototypische Realisierung und anschließende Bewertung finden sich im Kapitel 8.

5 Stand der Technik

Im diesem Kapitel werden für die Arbeit verwendeten Techniken und Formate vorgestellt.
Dies beinhaltet einen Einblick in die Technologie eines SAP R/3–Systems, wie auch eine Übersicht über vorhandene Integrations-Schnittstellen.

5.1 XML

5.1.1 Überblick

XML (Extended Markup Language) [W3C-XML1998] ist eine textbasierte Meta-Sprache, die sich für die Beschreibung, den Austausch, sowie für die Darstellung und die Manipulation von strukturierten Daten eignet.
Ein großer Teil des geschäftlichen Datenaustauschs, viele Datenformate und auch die in Kapitel 7 beschriebene Realisierung bauen auf XML auf. In Folge dessen sollen wesentliche Vorteile und Techniken von XML näher erläutert werden.

XML bietet für die Beschreibung von Daten ein einheitliches und standardisiertes Modell: vgl. [Kelkar2003]

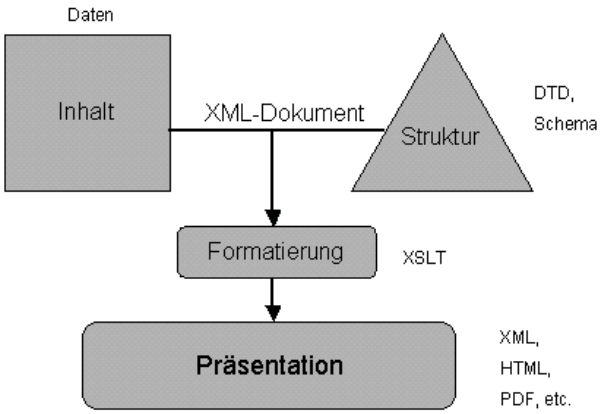

Abbildung 5-1 XML-Sprachkonzept

In Folge dessen wird die Trennung von Daten, deren Struktur und Ihrer Präsentationen möglich. DTD (Document-Typ-Definition) und Schemata ermöglichen eine Definition der formalen Grammatik, XSLT (Extensible Stylesheet Language for Transformations) realisieren die Präsentation.

5.1.2 XSLT

Die Extensible Stylesheet Language for Transformations (XSLT) ist neben XPath, das im folgenden Abschnitt beschieben wird, ein Teilbereich der Extensible Stylesheet Language (XSL) die vornehmlich zur Präsentation von XML-Dokumenten entwickelt wurde.

XSLT erlaubt die Transformation eines XML-Dokuments von einer Struktur in eine andere Struktur. Die für die Transformation benötigten Regeln werden in sog. XSLT-Stylesheets

definiert und sind selbst in einem XML-Dialekt formuliert. Ein XSLT-Prozessor kann aus einem XML-Dokument (Quelle) mit Hilfe der in einem Stylesheet hinterlegten Regeln ein neues XML-Dokuement (Ziel) erzeugen. Die Dokumente können dabei in Ihrer Struktur vollkommen unterschiedlich sein. Folgende Abbildung verdeutlicht den Ablauf einer Transformation:

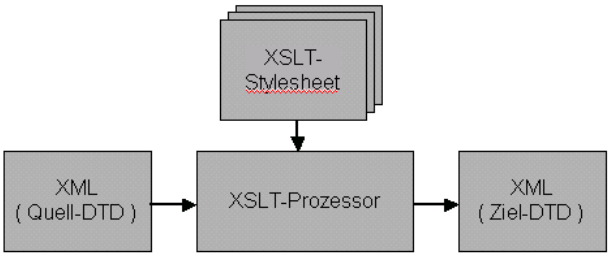

Abbildung 5-2 Funktionsweise der XSLT-Transformation

Die Sprache wird vor allem beim Datenaustausch für die Überwindung von unterschiedlichen XML-Formaten / Dialekten eingesetzt.

5.1.3 XPath

Bei der Entwicklung von Stylesheets kann sowohl die Struktur eines Dokuments geändert werden, als auch ihr Inhalt. Dazu bedient man sich sog. XPath-Konstrukte, die selektiven Zugriff auf Daten des Quelldokuments erlauben. Mit XPath lassen sich sehr leistungsfähige und mächtige Konstrukte realisieren. Das Zieldokument kann abhängig von den Inhalten des Quelldokuments geformt und erzeugt werden.

XPath wurde zusammen mit XSLT im November 1999 [W3C-XSLT1999] veröffentlicht. In XSLT-Dokumenten werden XPath-Ausdrücke für die Selektion von Daten verwendet.

5.1.4 XML Schema

Für die Validierung und die Beschreibung der Struktur von XML-Dokumenten existieren Document Type Definition (DTD) [W3C-DTD1999] und XML-Schema [W3C-Schema2001].

DTD besitzen Nachteile gegenüber XML-Schemata im Hinblick auf die Formulierung von Datentypen und deren Einschränkungen (z.B. keine Möglichkeit die maximal erlaubte Länge einer Zeichenkette zu definieren) In Folge dessen werden im weiteren nur XML Schemata betrachtet..

XML Schemata werden selbst in XML definiert und ermöglichen die genaue Definition einer Struktur sowie inhaltliche Vorgaben (z.B. Datentypen, Abhängigkeiten, Festwerte) denen ein XML-Dokument unterliegen soll.

Ein Schema lässt sich aufgrund der Eigenschaft, dass es in XML formuliert ist, selbst ebenfalls durch ein Schema auf Gültigkeit prüfen.

Im geschäftlichen Datenaustausch werden sowohl XML-Schema, wie auch DTD verwendet. Der Einsatz von Schemata bietet sich an, wenn die Struktur und die inhaltlichen

Einschränkungen eines Dokuments im Hinblick auf die Vorgaben einer formalen Grammatik geprüft werden sollen.

5.2 openTRANS

Das openTRANS-Format ging aus der im Dezember 1999 vom eBSC (eBusiness Standardisation Comitee) [Schmitz1999] gegründeten openTRANS-Initiative hervor. Leitgedanke und Ziel der Initiative ist die Entwicklung eines internationalen Standards zur elektronischen Übertragung von Geschäftsdaten und für die Abwicklung von Geschäftstransaktionen.

Durch die Verwendung von XML ist eine einfache und kostengünstige Lösung für den Austausch von Geschäftsdokumenten sichergestellt, die frei verfügbar und herstellerunabhängig ist. Die Entwicklung des openTRANS-Standards erfolgt ähnlich wie bei Katalogdatenstandard BMEcat [Schmitz2001a], [Schmitz2001b] anhand von grundsätzlichen Leitgedanken, die alle definierten Dokumente charakterisieren:

- Internationale Verwendbarkeit
- Unterteilung der Geschäftsdaten in mehrere Segmente (Kopfdaten, Geschäftspartnerinformationen, Artikeldaten)
- Unterstützung von Teilprozessen durch unterschiedliche Nachrichtentypen
- Separate Übertragung von Zusatzdaten möglich (z.B. Bilder)
- XML als Basistechnologie
- Einfache Erweiterbarkeit
- Unterscheidung von Pflichtfeldern und optionalen Angaben
- Verwendung von Datentypen und Feldlängen sowie Zusatzregeln
- Unabhängigkeit vom Übertragungsmedium (Internet, Diskette, CD-ROM, usw.)

openTRANS-Geschäftsdokumente sind kompatibel zu BMEcat-Katalogdokumenten. Beide Formate verwenden identische Feldbezeichnungen und Syntax. Diese Übereinstimmung erleichtert die einheitliche Nutzung beider Standards in Softwaresystemen.

Während BMEcat jedoch ausschließlich für den Austausch von Produktkatalogen vorgesehen ist, werden durch openTRANS die relevanten Geschäftsprozesse bei der Abwicklung von Bestellungen unterstützt. Damit ergänzen sich die beiden Formate und bieten so umfassende Möglichkeiten zur Automatisierung des Geschäftsverkehrs im Beschaffungsbereich. Die enge Verknüpfung mit dem weit verbreiteten BMEcat wird als großer Vorteil bei der Bewertung der zukünftigen Entwicklungschancen von openTRANS angesehen.

Für Ende 2003 wird mit der Version 2.0 die Vereinigung von BMEcat und openTRANS zu einem E-Commerce-Set angekündigt. Die Dokumentation beider Formate soll dann auf eine gemeinsame Element-Bibliothek zurückgreifen.

Die für diese Arbeit verwendete openTRANS-Version 1.0 liegt seit September 2001 vor und unterstützt den Datenaustausch für Bestellprozesse zwischen Kunden und Lieferanten durch folgende Nachrichtentypen:

- RFQ (Request For Quotation, Angebotsaufforderung)
- QUOTATION (Angebot)
- ORDER (Bestellung)

- ORDERCHANGE (Bestelländerung)
- ORDERRESPONSE (Auftragsbestätigung)
- DISPATCHNOTIFICATION (Lieferavis)
- RECEIPTACKNOWLEDGEMENT (Wareneingangsbestätigung)
- INVOICE (Rechnung)

Anhand der Nachrichtentypen kann der Prozess der Bestellabwicklung zwischen Kunden und Lieferanten wie folgt abgebildet werden: vgl. [Kelkar2003]

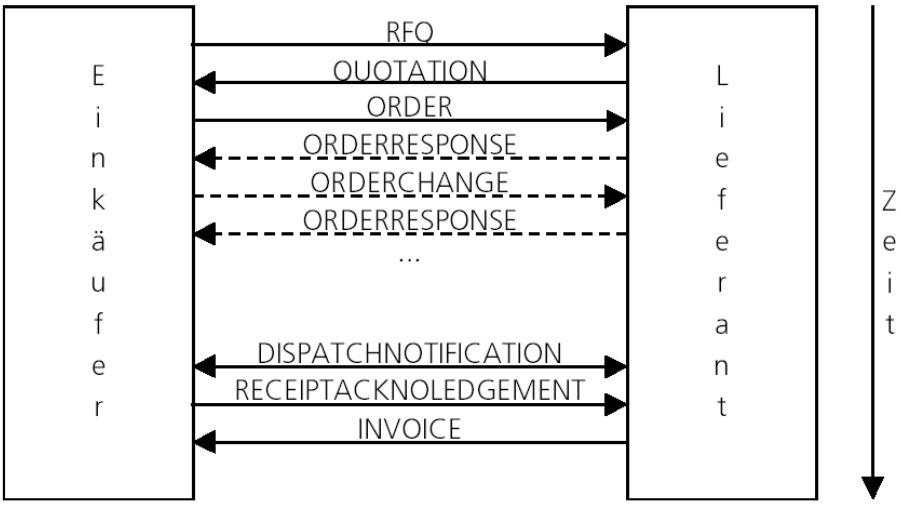

Abbildung 5-3 Nachrichtentypen in der Bestellabwicklung

Beispieldokumente für die in dieser Arbeit verwendeten Nachrichtentypen RFQ, QUOTATION, ORDER und ORDERRESPONSE finden sich im Anhang E1.

Über die derzeitige Verbreitung von openTRANS in der Praxis gibt es aufgrund der freien Lizenzbestimmungen keine verlässlichen Aussagen. Die openTRANS-Initiative geht von Implementierungen bei „mehreren Dutzend“ Organisationen aus, darunter sieben der größten deutschen Unternehmen.

5.3 Anwendungsystem SAP R/3

Das Produkt SAP R/3 stellt eine der umfassendsten und weitreichendsten Lösungen für die betriebswirtschaftliche Datenverarbeitung dar. Die modulare Aufteilung des Systems in einzelne betriebswirtschaftliche Bausteine wie etwa Logistik oder Personalwirtschaft ermöglicht einen flexiblen Einsatz in Abhängigkeit von Unternehmensbranche und individuellen Anforderungen. Die SAP AG ist der weltweit größte Hersteller von betriebswirtschaftlicher Software. Weltweit sind mehr als 12.000 SAP-Systeme im Einsatz. Vgl. [Walter2001]

Als betriebswirtschaftliche Software bildet R/3 in über 800 Geschäftsprozessen die Abläufe innerhalb eines Unternehmens durch teilweise auch eigenständig einsetzbare Module ab, deren Gesamtnutzen aber erst durch die Integration mehrerer Module vollständig zur Geltung kommt.

Erforderliche Anpassungen der Standardmodule an Kundenwünsche und kundenspezifische Gegebenheiten werden über ein in die Software integriertes Customizing-System ermöglicht.

Für die Erstellung von Ergänzungen der Standardfunktionen bzw. für Add-on-Entwicklung wird eine komplette Softwareentwicklungsumgebung angeboten, mit deren Hilfe auch individuelle Lösungen integriert und eingebunden werden können.

SAP R/3 kann auf verschiedenen Betriebssystemen (z. B. Unix / Windows) eingesetzt werden. Die Unternehmensdaten werden in zahlreichen Tabellen in einer Datenbank gespeichert. Als Datenbankmanagementsysteme (DBMS) kommen Fremdprodukte wie Oracle, Informix oder der Microsoft SQL-Server zum Einsatz.

SAP R/3 verwendet eine dreistufige Client-Server-Architektur. Dabei handelt es sich um eine Trennung in Präsentationsschicht, Anwendungsschicht und Datenbankserver.

Die Entwicklung innerhalb des SAP R/3-Systems erfolgt über die ABAP/4-Programmiersprache. Ein zentraler Bestandteil der gesamten Architektur ist das ABAP/4 Dictionary. Dabei handelt es sich um ein Verzeichnis über Felder, Tabellen und Beziehungen innerhalb des Systems.

Der konventionelle, interaktive Zugriff auf das SAP-System erfolgt über das SAPgui, welches den Benutzern ermöglicht, direkt mit dem SAP-System zu kommunizieren.

5.3.1 SAP Business Framework

Das SAP Business Framework ist die technische Basis (API) eines SAP-Systems. Es ermöglicht den einheitlichen Zugriff auf alle Daten des Systems und ist somit die zentrale Schnittstelle zwischen Anwendungen und Datenbank.

Das Framework erleichtert die durch eine über mehrere Version abwärtskompatible Schnittstellendefinition die Interaktion von R/3 mit Altsystemen, kundenspezifischen Anwendungen sowie Lösungen anderer Hersteller. In Folge dessen ist das Business Framework für eine Integration von zentraler Bedeutung.

Elemente des Business Frameworks sind:

5.3.1.1 Business Komponenten

Dies sind die installierten Module des Systems. Die SAP Business Komponenten sind eigenständig wartbar und setzen sich aus Business-Objekttypen zusammen. Beispielsweise sind der Business-Komponente Personalmanagement (Human Resources) unter anderem die Business-Objekttypen Angestellter (Employee) und Auftraggeber (Applicant) zugeordnet. Geschäftsprozesse werden entweder innerhalb einer Business-Komponente implementiert oder können sich über mehrere Komponenten erstrecken.

5.3.1.2 Business-Objekte

Business-Objekte sind die Grundlage der objektorientierten Strukturierung des SAP-Systems. Ein Business-Objekt ist das Abbild eines betriebswirtschaftlichen Sachverhaltes. Es umfasst sowohl die Funktionalität als auch die Daten dieses Sachverhaltes.

Business-Objekte kapseln die Daten und Geschäftsprozesse und verbergen auf diese Weise Struktur und Implementierungsdetails. Dazu wird ein mehrstufiges Schichtenmodell verwendet, das ein Objekt in eine Zugriffsschicht, eine Schnittstellenschicht eine Integritätsschicht und den eigentlichen Kern aufteilt.

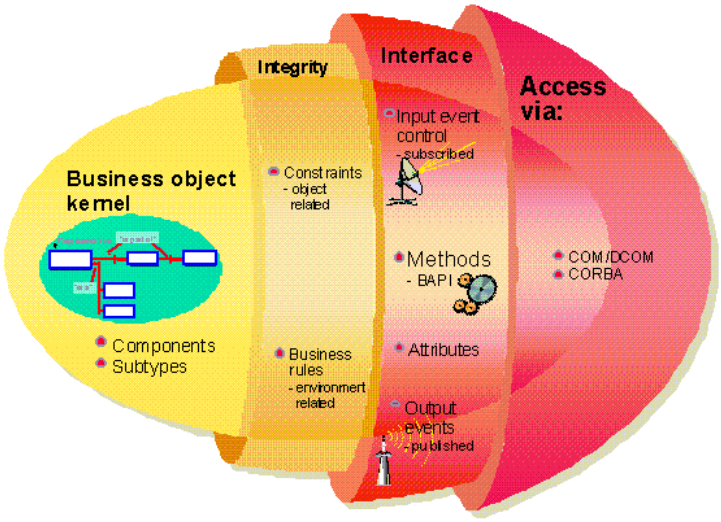

Abbildung 5-4 Schichtenmodell eines Business-Objekts

Die oberste Schicht, die *Zugriffsschicht* ermöglicht die Verwendung des Objekts von außen. Dabei kann auf ein Business Objekt mittels RFC, CORBA [CORBA1995] oder COM/DCOM zuggegriffen werden.
Die *Schnittstellenschicht* trennt die Daten des Objekts von der Anwendung, bzw. von der Technik, mit der auf sie zugegriffen wird. Die Schicht definiert die Attribute, BAPIs und Ereignisse, die das Objekt besitzt.
Während die *Integritätsschicht* die Konsistenzbedingungen gewährleistet enthält der *Kern* die eigentliche Implementierung der Methoden.
Business Objekte besitzen einen Typ, von dem jeweils eine Instanzen erzeugt werden kann. So ist der Kundenauftrag „62113“ eine Instanz vom Objekttyp „SalesOrder“.

5.3.1.3 Business Application Programming Interface

BAPIs bieten stabile und standardisierte Methoden auf Business-Objekte und werden in der SAP eigenen Programmiersprache ABAP/4 geschrieben. BAPIs bieten eine objektorientierte Sicht auf die betriebswirtschaftliche Funktionalität des SAP-Systems. Der Zugriff auf die Daten eines Business Objekts ist ausschließlich über BAPIs möglich.

Sie werden im Business Object Repository eines SAP-Systems hinterlegt. Dort können ggf. auch kundenspezifische Anpassungen vorgenommen werden und der Rollout für eine Gruppe von SAP-Systemen erfolgen (z.B. unternehmensweite Verteilung einer Anwendung)

Merkmale von BAPIs sind: vgl.[SAP-Bapi2001]

Merkmal	Bescheibung
Kapselung	Ein BAPI stellt Methoden an einem Business-Objekt dar. BAPIs sind unabhängig von der konkreten Realisierung der Objekte.
Kompatibilität	Nachdem ein BAPI von SAP freigegeben wurde, bleiben seine Schnittstellendefinitionen und Parameter langfristig stabil, d.h. Anwendungen werden nicht von Änderungen an zugrundeliegenden Daten beeinflusst. Sollten aufwärtskompatible Erweiterungen an einem BAPI vorgenommen werden, wird die Stabilität der vorhandenen Anwendungen nicht beeinträchtigt. Bei inkompatiblen Erweiterungen wird ein Zähler im Namen des BAPI erhöht. (SalesOrder.createfromdat1, SalesOrder.createfromdat2)
Erweiterbarkeit	Ein BAPI kann im Business Object Repository an kundenspezifische Anforderungen angepasst werden. Alle BAPIs liegen im Quellcode vor.
Transaktions-sicherheit	Der Aufruf eines BAPIs hinterlässt stets einen konsistenten Datenbankzustand. Es existieren Rollback und Commit-Methoden in jedem BAPI, die normalerweise nach dem Ausführen einer Methode an einem BAPI ausgeführt werden. Zusätzlich kann über das sog. ServiceBAPI auch eine Folge von BAPI-Aufrufen innerhalb einer Transkation ausgeführt werden.
Berechtigungen	Jede Benutzeraktion in einem SAP-System setzt bestimmte Berechtigungen voraus. Wenn ein Benutzer eine Anwendung ausführt, die eine Instanz eines Business-Objekt erzeugt hat, wird das BAPI von der Anwendung mit den Rechten des Benutzers ausgeführt. Ein BAPI definiert Berechtigungen für Benutzer anhand von Aktionen an einem Business-Objekt. Beispielsweise kann ein Benutzer im Warenlager am Business-Objekt Kundenauftrag "SalesOrder" nur die Aktion „getDetail" ausführen, und nicht die Aktion „create" order „delete". Es ist Ihm so möglich, einen Kundenauftrag einzusehen um die Konfektionierung der entsprechenden Waren vorzunehmen. Es kann jedoch keine Kundenaufträge anlegen oder löschen.
Offenheit	Auf BAPIs kann von allen gängigen Entwicklungsplattformen aus zugegriffen werden. Es existieren DCOM-, COM-, RPC-, CORBA- [CORBA1995] , JAVA- und XML-Connectoren.
Namen-konventionen	Ein BAPI wird durch den Namen des zugehörigen Business-Objekts gefolgt von dem Namen des BAPIs vollständig identifiziert. Der englische Funktionsname eines BAPI definiert die Aktion, die am Business-Objekt ausgeführt wird. (z.B SalesOrder.getDetail) Für viele Business-Objekte stehen eine Reihe von standardisierten Funktionen zur Verfügung, was sich durch einen einheitlichen Funktionsnamen ausdrückt. (z..B. getList,. getDetail, delete, create)

Ein Anwendungsprogramm, das ein SAP-Business-Objekt und seine Daten nutzen möchte, benötigt nur die Informationen zur Ausführung der Objektmethoden. Auf diese Weise kann ein Anwendungsprogramm mit den SAP-Business-Objekten arbeiten und ihre Methoden aktivieren, ohne die zugrundeliegenden Implementierungs- und Codierungsdetails des Objekts zu kennen zu müssen. Es genügt vielmehr, die Schnittstellendefinition der Methode zu kennen.

Diese Schnittstellendefinition eines BAPI kann bestehen aus:

- *Importparameter* mit Daten, die vom aufrufenden Programm an das BAPI übergeben werden sollen.
- *Exportparameter* mit Daten, die vom BAPI an das aufrufende Programm zurückgegeben werden sollen.
- *Tabellenparameter* für den Datenimport und -export.
- *Ausnahmen* für Fehlerbedingungen, die bei der Ausführung der BAPI-Methode auftreten können.

Weiterführende Informationen finden sich in [Janko1999]

5.3.1.4 Application Link Enabling

Der Integrationsdienst ALE (Application Link Enabling) ermöglicht die technische Integration von Geschäftsprozessen, die über mehrere SAP- oder Fremdsysteme hinweg abgewickelt werden.

Folgende Abbildung veranschaulicht die Architektur:

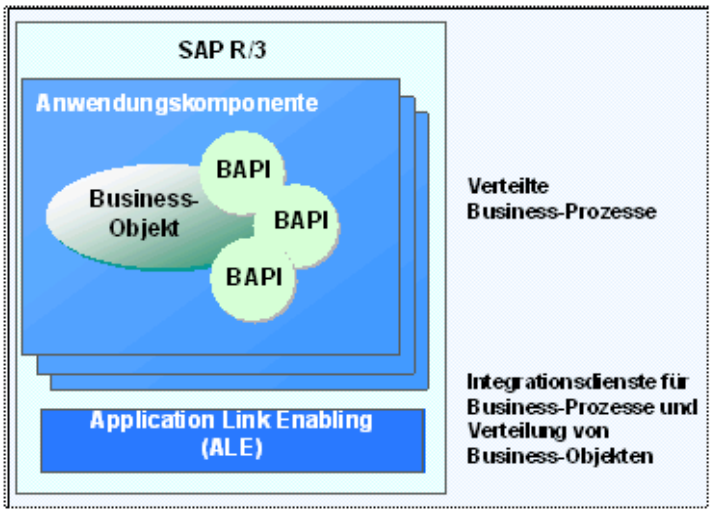

Abbildung 5-5 Application Link Enabling -Architektur

Grundlage hierfür ist eine systemgrenzenübergreifende Verteilung der betreffenden Business-Objekte (z.B. Materialien, Kundenaufträge etc.). Der Datenaustausch zwischen den Anwendungen erfolgt dabei über IDocs.

ALE leistet dabei:

- Die Gewährleistung der Kommunikation zwischen unterschiedlichen Release-Ständen der beteiligten SAP-Systeme

- Fortsetzung des Datenaustauschs nach Release-Wechsel ohne Modifikation
- Keine Abhängigkeit des technischen Formats einer Nachricht von deren Inhalt
- Erweiterungen, ggf. auch durch den Kunden, problemlos möglich
- Abkopplung der Applikation von der Kommunikation
- Unterstützung der Anbindung von Nicht-SAP-Anwendungen über Kommunikationsschnittstellen
- Berücksichtigung von R/2-R/3- und R/3-R/3-Szenarien

Die Anwendungsintegration erfolgt nicht über eine zentrale Datenbank, sondern über synchrone und asynchrone Kommunikation.

5.3.1.5 Kommunikationsdienste

SAP setzt seit Beginn seines Bestehens auf RFC (RemoteFunctionCall), was als eine SAP-Variante des RPC (RemoteProcedureCall) angesehen werden kann.

Neben dem RFC-Protokoll bietet SAP für die Integration von Fremdsystemen auch CORBA [CORBA1995] und Distibuted Component Object Model (DCOM) als Schnittstelle an.

Nahezu alle Bausteine eines SAP-Systems kommunizieren untereinander per RFC. Das RFC-Schnittstellensystem ermöglicht Funktionsaufrufe zwischen zwei SAP-Systemen oder zwischen einem SAP-System und einem außerhalb der Unternehmensgrenzen liegenden Nicht-SAP-System.

Die RFC-Schnittstelle wurde über die Releases des SAP-Systeme immer weiter ausgebaut und bietet mittlerweile verschiedene Arten von RFC-Aufrufen an:

RFC	Bescheibung
Synchron	Die erste Version des RFC war der synchrone RFC (sRFC). Dieser RFC-Typ führt den Aufruf einer Funktion auf der Grundlage der synchronen Kommunikation durch, d.h. die beteiligten Systeme müssen zum Zeitpunkt des Aufrufs beide ansprechbar sein.
Asynchron	Um die Probleme des sRFC, das beide Systeme aktiv sein müssen, zu beheben wurde der sRFC entwickelt. Es arbeitet mit lokalen Puffern.
Transaktional	Der transaktionale RFC (tRFC) ist eine asynchrone Kommunikationsmethode, die den gerufenen Funktionsbaustein im RFC-Server genau einmal ausführt. Das entfernte System muss in dem Moment, in dem das RFC-Client-Programm einen tRFC ausführt, nicht verfügbar sein. Die entfernte tRFC-Komponente speichert die gerufene RFC-Funktion zusammen mit den zugehörigen Daten in der Datenbank unter einer eindeutigen Transaktionskennung ab. Wird ein Aufruf gesendet während das Empfangssystem nicht verfügbar ist, bleibt der Aufruf in der lokalen Warteschlange stehen. Das rufende Dialogprogramm kann weiterlaufen ohne darauf zu warten, ob der Funktionsbaustein mit oder ohne Erfolg durchgeführt wurde. Wird das Empfangssystem nicht innerhalb eines bestimmten Zeitraums aktiviert, so wird der

	Aufruf als Hintergrund-Job eingeplant.
Queued	Um eine von der Anwendung vorgegebene Reihenfolge mehrerer logischer Arbeitseinheiten (LogicalWorkUnit) zu garantieren, (z.B. Kunde anlegen, Kundenauftrag für diesen Kunden erzeugen) wird der queued RFC (qRFC) verwendet. Dabei wird eine Transaktion erst dann übertragen, wenn sie keine Vorgänger (bezogen auf die in verschiedenen Anwendungsprogrammen definierte Reihenfolge) in den beteiligten Queues hat.

5.3.2 IDoc

IDoc (Intermediate Document) [SAP-IDoc1999] ist ein durch die SAP AG standardisiertes, auf Textdateien aufbauendes, Belegformat. IDoc erlaubt es, verschiedene Anwendungssysteme über eine nachrichtenbasierte Schnittstelle zu koppeln. IDoc wurde vornehmlich für die Kommunikation zwischen mehreren im Unternehmen verteilten SAP-Systemen entwickelt.

Jedes IDoc besteht aus einem Kontrollsatz, einer Reihe von mehreren Datensegmenten und einigen Statussätzen.

- Im **Kontrollsatz** des IDoc werden der IDoc-Typ, der Nachrichtentyp, der Sender und der Empfänger des IDoc definiert;
- jedes **Datensegment** enthält einen Standardheader, der aus einer fortlaufenden Segmentnummer und dem Segmenttyp besteht. Daran schließen sich die eigentlichen Daten des Segments an;
- die **Statussätze** beschreiben die bisherigen Verarbeitungsschritte des Idoc;

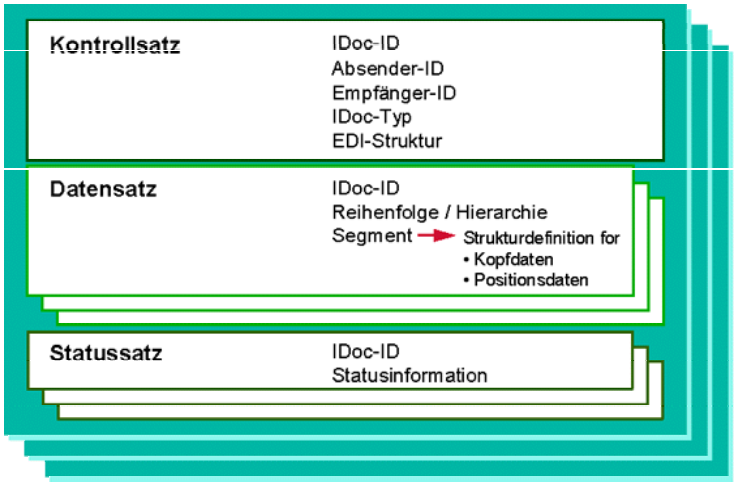

Abbildung 5-6 Aufbau eines IDoc-Dokuments

Ein Kunde kann ein IDoc erweitern und neue IDoc-Typen anlegen, um flexibel eigene oder anwendungsspezifische Daten in einem IDoc zu transportieren. In einem SAP-System sind standardmäßig bereits über hundert IDoc-Typen vordefiniert.

IDoc-Typen sind releaseunabhängigen Nachrichtentypen zugeordnet. So können auch Systeme unterschiedlicher Releasestände miteinander kommunizieren. Wenn durch ein neues SAP-Release Änderungen notwendig sind, wird ein Zähler im Funktionsnamen des IDoc

erhöht. (ORDERS01, ORDER02, ORDER03, …) Die Kommunikation auf Basis bereits freigegebener IDoc-Typen bleibt davon unberührt.

Wenn ein IDoc von einer Anwendung des SAP-Systems erzeugt wird, kann es von der Anwendung nicht selbst verschickt werden, sondern muss an die ALE (Application Linking Enabling)-Schicht übergeben werden.

Diese Schicht ist für den IDoc-Transport zuständig und ermittelt die Transportart aufgrund von für den Empfänger hinterlegten Konfigurationen. (vgl. auch [SAP-IDoc1999]) Ein IDoc kann auf folgende Arten versand werden:

- Dateischnittstelle
- RFC Schnittstelle
- CPI-C Schnittstelle
- Internet Schnittstelle http(s)

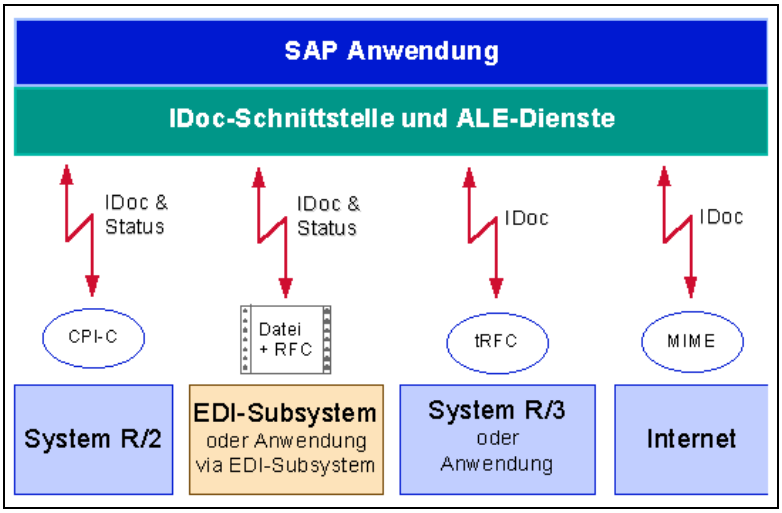

Abbildung 5-7 Der IDoc-Transport

Ein SAP-System verfügt in Folge dessen nicht über eine XML-Schnittstelle. Im Zuge der großen Verbreitung von XML wurde mit dem SAP Business Connector ein Standardprodukt für alle SAP-Benutzer bereitgestellt, das aus jedem IDoc eine äquivalente XML-Darstellung erzeugen kann.

5.3.3 SAP Business Connector

Der Business Connector ermöglicht bidirektionale synchrone wie asynchrone Kommunikation von und zu einem SAP-System. Er wird dabei vor ein SAP-System geschaltet und erweitert das SAP-System um zusätzliche Schnittstellen.

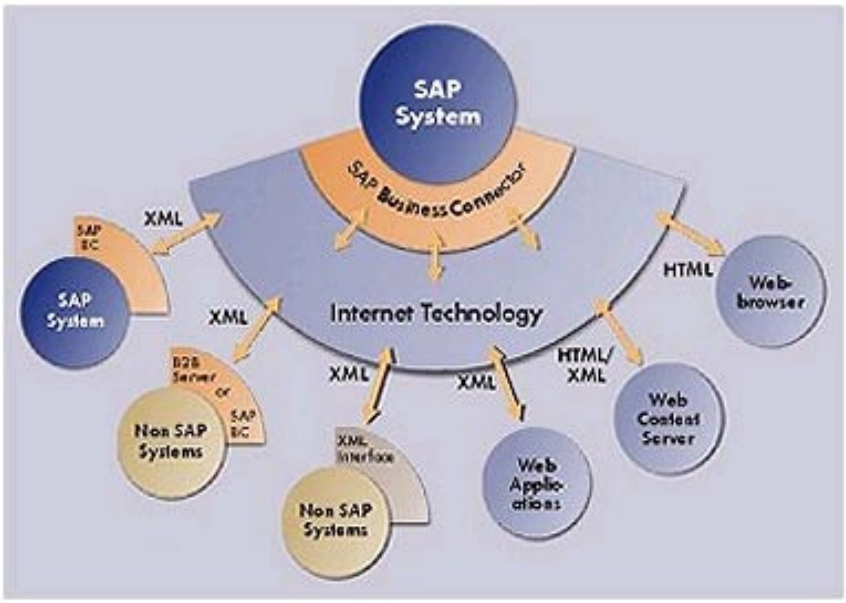

Abbildung 5-8 SAP Business Connector Architektur

Die Kernfunktionen des Business Connectors sind:

5.3.3.1 Erweiterte Schnittstellen

Ein SAP-System kann standardmäßig per RFC Dokumente versenden. Durch den Einsatz des Business Connectors wird das System fähig, auch andere Transportarten nutzen zu können. Der Business Connector bietet HTTP, HTTPS, Datei, FTP und Email-Versand wie auch -Empfang an.

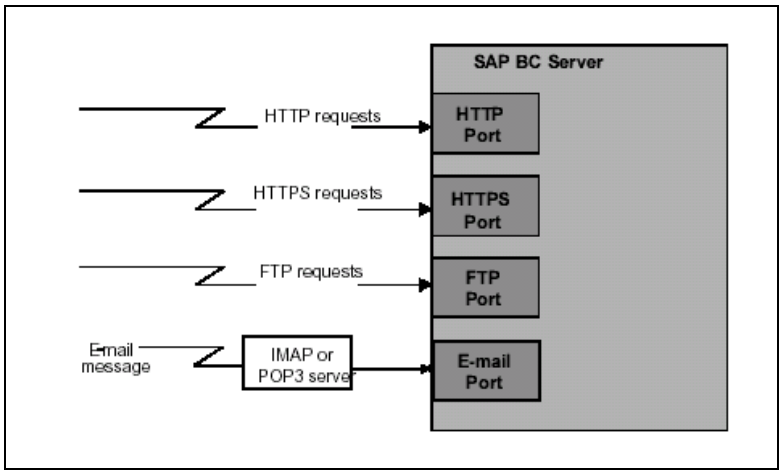

Abbildung 5-9 Schnittstellen des Business Connectors

Über diese Schnittstellen können nicht nur IDocs übermittelt werden, sondern auch beliebige andere Formate wie beispielsweise XML oder Text (CSV)-Dialekte. Dadurch wird der Dokumentenaustausch auch mit Nicht-SAP-Systemen möglich.

- **Umsetzung von Formaten**
  Um die SAP-internen Technologien auch nach außen hin offen zugänglich zu machen, bietet der Business Connector auch die Möglichkeit, IDocs, RFC- und

BAPI-Aufrufe XML-codiert zu tätigen. SAP bietet im dem Interface Repository [SAP-IFR2003] eine umfassende Referenzbibliothek für alle Formatdefinitionen.

- **IDocs**
  Zu jedem IDoc-Typ kann im Business Connector eine äquivalente XML-Darstellung erzeugt werden. Dies ist in beiden Richtungen möglich. (IDoc → XML, XML → IDoc) Das technologisch überholte textbasierende IDoc-Format wird dadurch kompatibel zur weit verbreiteten XML-Technologie.

- **RFC**
  Um RFC-Aufrufe auf einfachste Art und Weise zu ermöglichen, bietet der Business Connector für jeden RFC-Aufruf eine entsprechende XML-Aufruf-Notation an. Die Antwort eines RFC-Aufrufs wird ebenfalls in XML an den Aufrufer zurückgesendet. Durch diese Technik können Anwendungen durch das Senden einer XML-Nachricht RFC-Funktionen in einem SAP-System aufrufen.

- **BAPIs**
  BAPIs können über XML aufgerufen werden. Die Antwort wird dann vom Business Connector ebenfalls in XML zurückgesendet. Auch hier wird es Anwendungen ermöglicht durch das Senden einer XML-Nachricht einen BAPI-Aufruf im SAP-System zu erzeugen.

5.3.3.2 Dynamisches Routing

Jedes direkt an den Business Connector gesendete Dokument wird, bevor eine Verarbeitung ausgeführt wird, an den Partner Manager gesendet. Es übernimmt in Abhängigkeit von konfigurierbaren Regeln die weitere Verarbeitung des Dokuments.

Eine Ausnahme stellen die im Business Connector hinterlegten benutzerdefinierten Flows [SAP-BC2003] dar. Diese sind grafisch oder per Java-Sourcecode definierte Programme, ähnlich den bei Webserver häufig eingesetzten CGI-Skripten. Sie können ein Dokument direkt empfangen und verarbeiten. Dazu muss das Dokument über eine den Flow identifizierende URL an den Business Connector gesendet worden sein. Sollte der Flow, z.B. nach Konvertierung der Nachtricht, den Versand einer Nachricht an das SAP-System auslösen, wird diese Nachricht auch an den Partner Manager übergeben.

Folgende Abbildung zeigt die Routing-Funktion des Partner Managers:

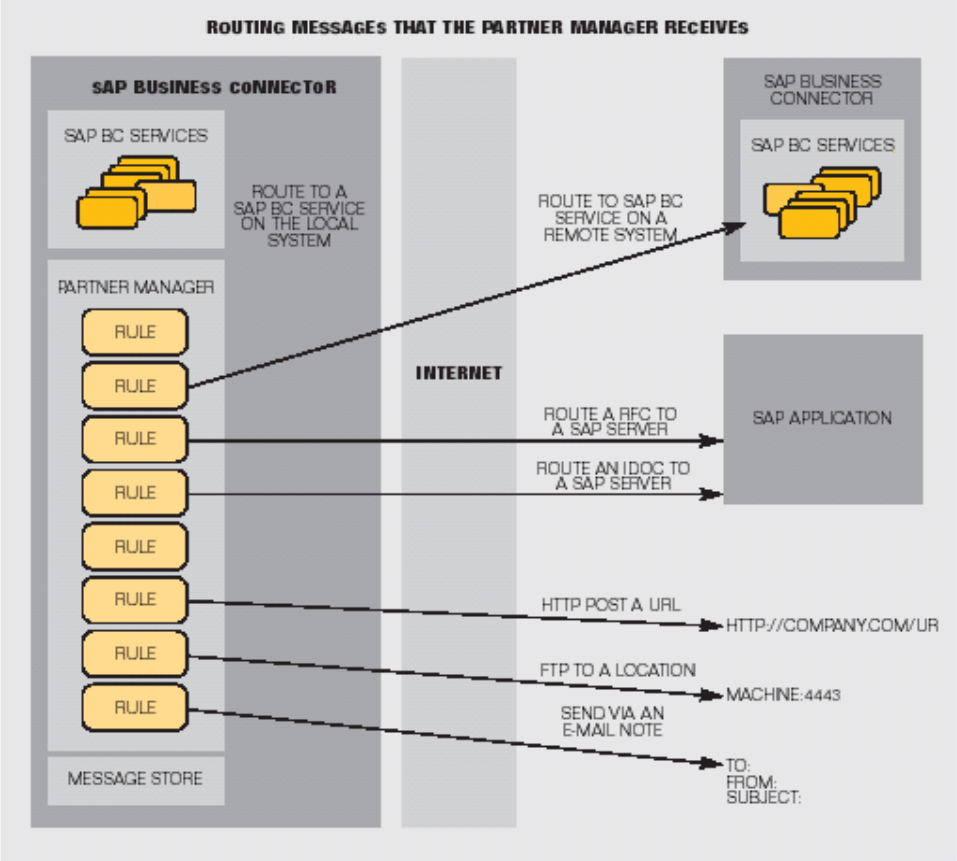

Abbildung 5-10 Routing-Funktion des Partner Managers

5.3.3.3 Routing-Regeln

Der Partner Manager entscheidet anhand von sog. Routing-Regeln, wie ein Dokument weiterverarbeitet wird. Wenn ein Dokument den Partner Manager erreicht, prüft dieser alle hinterlegten Regeln auf die Übereinstimmung von folgenden Kriterien.

Kriterium	Bescheibung
Sender	Der Sender wird aus der Nachricht extrahiert und mit den Sender-Angaben der hinterlegten Regeln verglichen. Bei einem IDoc kann der Sender aus dem Kontrollsatz herausgelesen werden. Im Falle eines XML-BAPI und eines XML-RFC wird beim Sender nur zwischen intern (vom SAP-System kommend) und extern (nicht vom SAP-System kommend) unterschieden. Direkte BAPI und RFC – Aufrufe (nicht per XML) passieren den Partner Manager nicht, da nur Nachrichten behandelt werden. Soll eine Regel unabhängig vom Sender ausgeführt werden, kann das Wildcard-Symbol „*“ für den Sender eingetragen werden. Dieses Symbol steht für alle möglichen Sender.
Empfänger	Der Empfänger wird wie der Sender aus dem Dokument extrahiert und abgeglichen. In einem IDoc ist der Empfänger im Kontrollsatz hinterlegt. Bei XML-BAPI und XML-RFC aufrufen findet nur eine Trennung

	zwischen Intern und Extern statt. Soll eine Regel unabhängig vom Empfänger ausgeführt werden, kann das Wildcard-Symbol „*“ für den Empfänger eingetragen werden. Dieses Symbol steht für alle möglichen Empfänger.
Nachrichtentyp	Der Business Connector kann von allen Ihm bekannten Dokumenten den Typ ermitteln. Bei einem IDoc wird der IDoc-Typ für das Routing herangezogen. Bei BAPI und RFC-Aufrufen wird der Name des BAPIs, bei RFC der Name des Funktionsbausteins für das Routing verwendet. Für den Nachrichtentyp lässt sich ebenfalls das Wildcard-Sybmol „*“ angeben.

Wenn für eine Nachricht eine Regel gefunden wurde, die in allen Kriterien mit den Inhalten der Nachricht übereinstimmt, wird der in der Regel definierte Transport der Nachricht ausgeführt.

Der Partner Manager protokolliert jede Nachricht und überwacht, ob der durch eine Regel ausgelöste Transport erfolgreich durchgeführt wurde. Es erstellt so ein Transaktionsprotokoll für jede Nachricht.

Jede Regel kann bei Ihrem Zutreffen vor und nach der Ausführung des spezifizierten Transportes die Ausführung eines Flow anstoßen. Dies ist beispielsweise sinnvoll, wenn eine Nachricht vor dem Transport per Email digital signiert werden soll, oder wenn nach dem Versand einer Nachricht ein Commit ausgelöst werden soll.

5.3.3.4 Transportarten

Als Aktion für eine Regel stehen verschiedene Transportarten zur Verfügung.

Transport	Bescheibung
B2B-Service	Führt einen Flow aus. Dabei wird dem Flow die Nachricht zur Verarbeitung übergeben.
FTP-Outbound Service	Sendet die Nachricht an eine FTP-Adresse. Parameter für Host, Passwort, User etc. werden direkt in der Regel konfiguriert.
Email-Outbound Service	Sendet die Nachricht per SMTP an eine Mail-Adresse. Parameter für Host, Email, Absender werden direkt in der Regel konfiguriert.
ALE (R/3 IDOC)	Sendet die Nachricht via tRFC an die Eingangsverarbeitung eine SAP-Systems.
BAPI	Führt den in der Nachricht im XML-Format übermittelten BAPI-Aufruf via tRFC aus.
RFC	Führt den in der Nachricht im XML-Format übermittelten RFC-Aufruf via tRFC aus.
XML	Sendet ein Dokument an eine URL.

Sollte ein Transport aufgrund eines Fehlers nicht durchgeführt werden können, wird die Nachricht im Transaktionsprotokoll als gescheitert markiert.

5.3.3.5 Berechtigungskonzept

Der Business Connector implementiert ein Berechtigungskonzept, bei dem auf verschiedenen Ebenen Rechte vergeben werden können:

Ebene	Bescheibung
Regeln	Jede Regel ist durch eine Zugriffsliste (ACL) beschränkt. Dabei muss der Sender einer Nachricht die erforderlichen Rechte besitzen, um die Regel ausführen zu können. Mit diesem Konzept kann verhindert werden, das unberechtigte Personen Nachrichten über den Business Connector senden können.
Flow	Jeder Flow ist mit einer Zugriffsliste beschränkt. Dadurch wird verhindert, das unberechtigte Personen Services und Funktionen am Business Connector ausführen.
Protokoll	Jedes zur Verfügung stehende Protokoll ist durch eine Zugriffsliste beschränkt. So kann festgelegt werden, das beispielsweise nur Zugriffe per HTTPs erfolgen dürfen.
SAP-System	Jeder BAPI / RFC-Aufruf wird vom SAP-System nochmals, isoliert vom Business Connector, auf Zugriffsberechtigung geprüft. Dabei fungiert der Business Connector als angemeldeter eigenständiger Benutzer des SAP-Systems. Durch selektives Setzen von Berechtigungen kann erreicht werden, das der Business Connector keinen (schreibenden) Zugriff auf kritische Funktionen bzw. -bereiche hat.

5.3.3.6 Grafische Programmierung von Flows

Der Developer Client [SAP-BC2003] des Business Connectors ist eine in Java implementierter Editor, der die grafische Programmierung von Flows erlaubt.

Mittels der Kombination von vordefinierten Modulen, sog. Flowsteps, wird ein Flow erzeugt. Mir den vordefinierten Flowsteps können u.a. Schleifen (LOOP, REPEAT), If-Abfragen (BRANCH) und Zuweisungen (MAP) realisiert werden.

Um auf ein SAP-System zugreifen zu können, existieren sog. Services, die Funktionsaufrufe realisieren.

Beispiele für Services:

Service	Bescheibung
pub.sap.transport.ALE:OutboundProcess	Sendet ein IDoc an ein SAP-System
pub.sap.transport.RFC:OutboundProcess	Führt einen RFC an einem SAP-System aus.
pub.sap.idoc:transformFlatToHierarchy	Überführt ein IDoc in seine äquivalente XML-Darstellung

Während ein Flow ausgeführt wird, speichert er alle lokalen Objekte, in einer sog. Pipeline (vgl. [SAP-BC2003] Kapitel 7). Über die Pipeline können Ergebnisse von Funktionsaufrufen zwischengespeichert werden, und als Eingabe für weitere

Funktionsaufrufen dienen. Sie entspricht der Halde eines Programms. Die Pipeline lässt sich auch lokal speichern und laden.

6 Anforderungen

In folgenden werden für die Arbeit relevante Anforderungen dargestellt.

6.1 System zu System Kommunikation

Unter der System zu System Kommunikation wird der automatisierte Austausch von Daten zwischen Rechnersystemen verstanden.

Die vollautomatisierte System zu System Kommunikation ist ein vor allem im eBusiness angestrebtes Ziel, durch das manuelle Vorgänge auf ein Minimum reduziert werden und die Potentiale der elektronischen Datenverarbeitung vollständig ausgenutzt werden.

Für die Gewährleistung einer reibungslosen Kommunikation zwischen Anwendungen wurden folgende Aspekte erarbeitet:

- **Prozesse**
  Automatische Kommunikation kann nur innerhalb der von beiden Geschäftspartnern definierten Prozesse erfolgen. In Folge dessen muss vor einer automatischen Kommunikationsbeziehungen eine Prozessdefinition vereinbart werden. (vgl. [ebXML2001])
  Die Prozessdefinition berücksichtigt alle auftretenden Fehlerfälle und Aktionen für den Fall eines Fehlers.

  Sollte in einem Bestellsystem beispielsweise eine Bestellung aufgrund eines Lieferenpasses nicht oder nur verzögert ausgeführt werden können, muss der Besteller informiert werden. Sollte der Prozess keine Nachricht für diesen Fall vorhalten, ist wiederum manuell einzugreifen.

- **Standardisierbarkeit**
  Eine wesentliche Vorraussetzung für automatische System zu System Kommunikation ist die Standardisierbarkeit von Prozessen. Viele Prozesse sind in Unternehmen zu umfangreich, als das man Sie standardisiert abwickeln könnte. Der Bau eines neuen Geschäftsgebäudes ist beispielsweise ein sehr umfangreicher Prozess, der nur bedingt über eine automatisierte System zu System Kommunikation abgewickelt werden könnte. (vgl. ebXML2001])
  Häufig werden daher einfache und oft auftretende Prozesse, wie beispielsweise die Bestellung eines Handwerkzeugs, automatisiert.

- **Formalisierbarkeit**
  Trotz leistungsfähiger Modellierungsmöglichkeiten sind Prozesse häufig zu komplex um formalisiert werden zu können. Viele Prozesse eigenen sich daher nur bedingt für eine automatisierte Kommunikation.

- **Sicherheit**

Wenn Systeme selbstständig kommunizieren, müssen Methoden für die Sicherung dieser Kommunikation bereitgestellt werden. Diese sind im Abschnitt 6.3 detailliert erläutert.

- **Asynchron / Synchron**
  Wenn Prozesse in Folge einer Automatisierung neu definiert oder umstrukturiert werden, kann durch die Einführung einer System zu System Kommunikation ein synchroner Prozess zu einem asynchronen Prozess werden. Dies kann beispielsweise der Fall sein, wenn aus technischen Gründen Nachrichten zwischengepuffert werden müssen. Ebenso kann ein asynchroner Prozess zu einem synchronen werden.

Wenn alle beteiligten Systeme einen einheitlichen Standard verwenden, ergeben sich durch den Einsatz einer automatische System zu System Kommunikation sich verschiedene Vorteile:

- **Geschwindigkeit**
  Die Minimierung von manuellen Vorgängen erwirkt eine wesentliche Reduzierung der Prozesslaufzeit.

- **Kostenreduktion**
  In Folge des Wegfalls von manuellen Vorgängen ergeben sich Kosteneinsparungen.

- **Fehlerminimierung**
  Durch automatische Generierung und Erfassung von Nachrichten können Fehler minimiert werden.

Probleme, die im Hinblick auf eine automatische System zu System Kommunikation auftreten können, sind Fehler, bei denen eine Anwendung auf eine Fehlersituation stößt, und diese mangels einer nicht definierten Aktion im Prozess nicht weiterkommunizieren kann, sowie die vielfach problematische Einigung auf einen gemeinsamen Standard. Darüber hinaus eignen sich auf Grund mangelnder Formalisierbarkeit nicht alle Prozesse für eine automatische System zu System Kommunikation.

6.2 Standardkomponenten

Standardkomponenten sind vorgefertigte, im Funktionsumfang für einen großen Anwenderkreis konzipierte Software-Komponenten, die ohne individuelle Anpassungen einsetzbar sind. Standardkomponenten verwenden, um eine gute Integrationsfähigkeit zu gewährleisten, eine offene Architektur und internationale Standard. [Kaftan2000]

Standardkomponenten verfügen neben der eigentlichen Anwendung auch über Systemdokumenation, Schulungsangeboten und Unterstützung bei Installation und Wartung.

Durch den Einsatz von Standard-Komponenten ergeben sich folgende Vorteile:

Vorteil	Bescheibung
Geringer Anschaffungspreis	In Folge des breiten Anwenderkreisen, der die Entwicklung finanziert, besitzen Standard-Komponenten einen geringeren Anschaffungspreis als vergleichbare Individuallösungen.
Permanente Weiterentwicklung	Ein Anwender kann eine Standard-Komponente auch zu einem späteren Zeitpunkt aktualisieren, das die Weiterentwicklung permanent und unabhängig von einem einzelnen Anender durchgeführt wird.
Hohe Funktionalität	Standard-Komponenten bieten Funktionen für einen breites Anwenderkreis und sind daher mit mehr Funktionalität ausgestattet, als ein durchschnittlicher Anwender benötigt.

Im Vergleich zu Individuallösungen besitzen Standard-Komponenten folgende Nachteile:

Nachteil	Bescheibung
Herstellerabhängigkeit	Die Wartung und Weiterentwicklung einer Standard-Komponente kann nur vom Hersteller der Komponente durchgeführt werden.
Mangelnder Einfluss	Hersteller von Standard-Komponenten richten die Entwicklung nach den Anforderungen der Mehrheit der Benutzer aus. Eine Individuallösung kann speziell auf die Anforderungen eines Benutzers ausgerichtet werden.

Gemäß der Aufgabenstellung werden Standard-Komponenten für die Realisierung verwendet.

6.3 Sicherheit

Frühere Massenmedien wie z.B. das Fernsehen, haben vier wesentliche Eigenschaften, die sie als Risikolose-Systeme einstufen: (vgl. [Schultz1997])

- Es sind unidirektionale Systeme
- Es sind Systeme die sich von einem ausgehend an viele richten
- Die Identität des Senders ist eindeutig und nicht verdeckt
- Die übertragenen Informationen haben rein passiven Charakter

Das Internet hat genau die gegenteiligen Eigenschaften:

- Es ist ein bidirektionales System
- Es ist ein System, das sich von vielen an viele richtet
- Die Identität des Informationsversenders ist programmierbar und kann verdeckt werden
- Die übertragenen Informationen können aktive Programme sein

Bei der Übertragung von digitalen Geschäftsdokumenten über das Internet ist es daher für die Geschäftspartner von zentraler Bedeutung, das Sicherheitsaspekte gewahrt werden. Dazu zählen unter anderem:

- **Vertraulichkeit** (*confidentiality)*
  Bei einer Datenübermittlung muss gewährleistet sein, das die Daten nicht von Dritten gelesen werden können. Dies trifft insbesondere bei sensiblen Daten wie Angebotspreisen, Rabatten etc. zu.

- **Authentizität** (*authentication)*
  Beide Geschäftspartner sind bestrebt, die Identität des jeweils Anderen zweifelsfrei festzustellen. Für einen Bestellvorgang muss beispielweise garantiert sein, das der angegebene Besteller auch der tatsächliche Besteller ist.

- **Integrität** *(integrity)*
  Es muss unmöglich sein, das Daten während der Übertragung von Dritten verändert werden. Der Empfänger einer Nachricht muss daher die Möglichkeit besitzen, festzustellen, ob eine Nachricht manipuliert wurde.

- **Zugriffskontrolle** *(access control)*
  In der Regel möchte ein Unternehmen vor der Annahme elektronischer Dokumente entsprechende Rahmenvereinbarungen treffen. Es benötigt daher Vorkehrungen, die sicherstellen, das nur berechtigte Unternehmen digitale Prozesse anstoßen können.

Die einzelnen Aspekte werden im folgenden an Markt gängigen Methoden zur Gewährung von Sicherheit kategorisiert und eingestuft.

6.3.1 Verfahren

In der Regel werden Geschäftsdokumente über Regionen-, Länder-, und Sprachgrenzen hinweg ausgetauscht. Es sind daher bei der Auswahl der Verfahren nur De-Facto-Industriestandards [Risak2003] aufgeführt, wie eine große internationale Verbreitung besitzen. Die folgende Tabelle bewertet die Verfahren hinsichtlich der in Abschnitt 6.3.1 aufgeführten Sicherheitsaspekte:

Verfahren	Vertraulichkeit	Authentizität	Integrität	Zugriffskontrolle
HTTPS	X	X –1	X	
S/MIME	X	X –1	X	-
HTTP-Auth.	-	X –2	-	X
Client-Zert.	-	X –2	-	-
Dig.Signatur		X –2	X	

1 – nur die des Empfängers
2 – nur die des Senders

6.3.2 Bewertung

Es zeigt sich, das kein De-Facto-Standard existiert, der alle Sicherheitsanforderungen abdeckt.

Durch eine Paarung von HTTPS, dem die Authentizität des Senders fehlt, mit einer HTTP-Authentifizierung kann in der Praxis ein einfach zu installierendes und allen Anforderungen gerecht werdendes Sicherheitssystem aufgesetzt werden. Ebenso wäre HTTPS in Verbindung mit Client-Zertifikaten möglich.

Bei der Übertragung von Emails muss die Prüfung einer Berechtigung im Mailserver oder bei der Verarbeitung der Emails erfolgen. Aufgrund des Mail-Prinzips, das an der Versendung von Briefen und deren Zustellung orientiert ist, kann bei eingehenden Emails keine Berechtigungsprüfung durchgeführt werden. In Folge dessen muss die verarbeitende Instanz prüfen, ob eine Nachricht die Berechtigung besitzt, verarbeitet werden zu dürfen.

Für die Praxis ergibt sich daher eine Verwendung von HTTPS mit Authentifizierung oder Client-Zertifikaten. Unterstützung für diese Verfahren findet sich in den meisten Standard-Produkten, was für eine hohe Praktikabilität steht.

7 Umsetzung asynchroner Nachrichtenaustausch

Dieses Kapitel beschreibt ein in der Industrie häufig auftretendes Integrationsszenario für die asynchrone Übermittlung von Bestellinformationen und stellt zwei unterschiedliche technologische Varianten der Realisierung einer Integration vor, für die eine Bewertung erfolgt.

7.1 Beschreibung der openTRANS-Integration

EBusiness-Standards werden meist zuerst von großen Firmen zum Einsatz gebracht, da sich aufgrund der Vielzahl ausgetauschter Nachrichten durch eine Vereinheitlichung die größten Einsparpotentiale ergeben.

Es wird daher ein beispielhaftes Integrations-Szenario beschrieben, in welchem ein Bestelldatenaustausch zwischen Einkäufer und Lieferante stattfindet.

Im Bestelldatenaustausch sendet das einkaufende Unternehmen eine Bestellnachricht über das Internet an das System des Lieferanten. Das System des Lieferanten verarbeitet die Bestellung selbstständig, prüft Vollständigkeit und Korrektheit der Daten und übermittelt anschließend eine Bestellbestätigung an das System des Einkäufers.

Der gesamte Prozess läuft asynchron ab. Als Transportart soll HTTPS in Verbindung mit einer HTTP-Authentifizierung verwendet werden. Der asynchrone Versand der Bestätigung ist in Abhängigkeit der Fähigkeiten des Einkäufersystems konfigurierbar. Hierbei stehten unterschiedliche Transportarten zur Verfügung: HTTPS, Email und FTP, wobei für jeden Kunden entsprechende Transportparameter einstellbar sind.

Das System des Lieferanten, für das im Folgenden eine Implementierung vorgestellt wird, ist ein SAP R/3 System (vgl. Kapitel 5.3.3) mit vorgeschaltetem SAP BusinessConnector. (vgl. Kapitel 5.3) Dabei ist der Business Connector die über das Internet erreichbare zentrale Schnittstelle des Lieferanten nach außen. Er kommuniziert im vorgestellten Szenario mit dem System des Einkäufers und wandelt Nachrichten in eine für das nachgeschaltete SAP-System lesbare Form um. Als eBusiness-Standard wird openTRANS (vgl. Kapitel 5.2) für die eingehende und ausgehende Kommunikation verwendet.

7.2 Systemkonzept

Der prinzipielle Aufbau wird durch Abbildung 7.1 verdeutlicht. Dargestellt ist die Übermittlung einer Bestellnachricht über das Internet. Die Bestellung wird per HTTPS-Aufruf an ein dafür vorgesehenes Programm des Business Connectors übergeben.

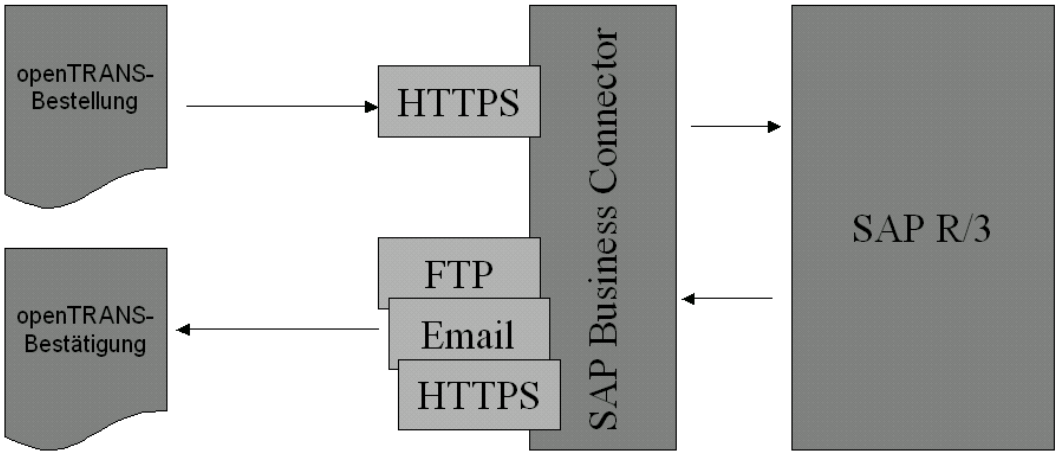

Abbildung 7-1 Systemkonzept

Die Realisierung lässt sich auf zwei Varianten durchführen: In der Realisierungsvariante (a) wird die Bestellung durch den Aufruf von BAPIs im SAP-System angelegt; In Variante (b) wird die Bestellung durch eine XSLT-Transformation in ein entsprechendes IDoc umgesetzt und durch anschließende Übergabe des IDoc an die SAP-Eingangsverarbeitung im System angelegt.

Nachdem die Bestellung im SAP-System angelegt ist, laufen verschiedne Prozesse in SAP ab:

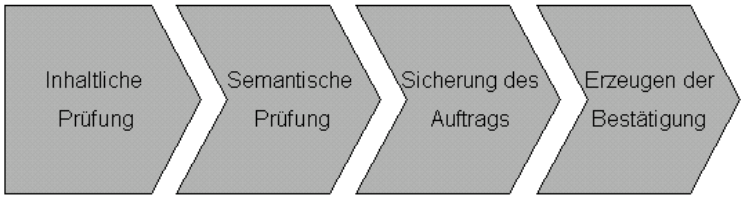

Abbildung 7-2 Validierungsprozess für eine Bestellung

Die Bestelldaten werden geprüft. Dies umfasst die inhaltliche Prüfung (z.B. erlaubte Bestelleinheiten, Artikelnummern vorhanden etc.) wie auch die semantische Prüfung (z.B. Artikel für Kunde angelegt, Lieferfristen, ggf. Teillieferungen etc.). In Folge dessen werden, sofern im System konfiguriert, auch benötigte Rohmaterialien und Produktionsmittel reserviert bzw. sogar beim Unterschreiten festgelegter Untergrenzen Bestellungen bei vorgeschalteten Zulieferern ausgelöst. Diese Prozesse enden, sofern die Bestellung geliefert werden kann, in der Erzeugung eines Bestellbestätigung, die wiederum über die Ausgangsverarbeitung des SAP-Systems an den angeschlossenen BusinessConnector übermittelt wird.

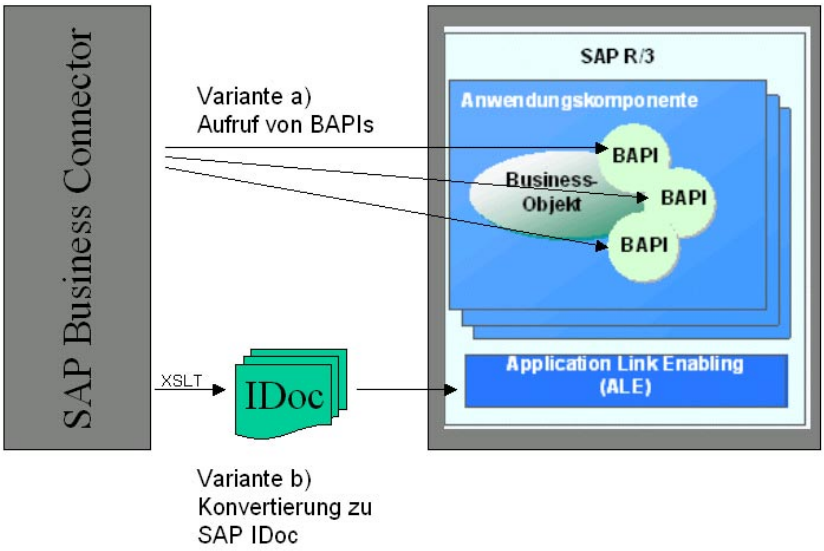

Abbildung 7-3 Realisierungsvarianten

Abhängig von der Realisierungsalternative wird im BusinessConnector (a) die Bestätigung direkt in ein entsprechendes openTRANS-Dokument umgewandelt, oder (b) durch ein XSLT-Stylesheet vom SAP-IDoc-Format nach openTRANS konvertiert.

Im Idealfall, sofern eine vollständige bzw. „bestellbare“ Bestellung übermittelt wurde, läuft der gesamt Prozess als direkte Kette von nacheinander folgenden Schritten ab.

Ist die Bestellung aus einem oder mehreren Gründen nicht bestellbar, hält der Prozess an, und erfordert manuelles Eingreifen. Gründe hierfür wären Fehler in der Semantik (z.B. Kabel nicht in Metern sondern in Litern bestellt, oder die Bestellung eines nicht existierenden Produkts ~ unvollständige oder fehlerhafte Artikelnummer) aber auch inhaltliche Fehler (z.B. falsche Preise) Eine solche Bestellung wird vom System als ungültig oder unvollständig markiert und mit einem Signal in der Bestellübersicht versehen.

Ein Mitarbeiter des Lieferanten im Auftrageingang kann durch Rücksprache mit dem einkaufenden Unternehmen die Fehler manuell korrigieren, oder die Bestellung aus dem System löschen. Wenn die Bestellung manuell berichtigt wird, läuft der Prozess weiter, und die Bestellbestätigung wird erzeugt und versand. Der Einkäufer erhält also in jedem Fall, sofern eine Lieferung stattfinden wird, eine Bestellbestätigung. Auch denkbar ist die automatische Löschung einer ungültigen Bestellung. Das einkaufende Unternehmen kann in diesem Fall eine fehlerhafte Bestellung am Ausbleiben der Bestellbestätigung erkennen.

7.3 Realisierung mittels BAPIS

Im Folgenden wird die Umsetzung des Szenarios mittels der BAPI-Technologie beschrieben. Kern dieser Realisierung ist die Erzeugung der Bestellung im SAP-System mittels eines BAPI-Aufrufs.

7.3.1 Senden einer Nachricht

Das einkaufende Untenehmen erzeugt eine Bestellung im openTRANS-Format (vgl. 5.2) und sendet diese per HTTPS-Aufruf an das System des Lieferanten. Dazu vereinbaren Einkäufer und Lieferant eine URL-Adresse, an welche die Nachricht gesendet wird.

Für die Evaluierung des Prototypen wurde eine Webseite eingerichtet, über die Nachrichten abgeschickt werden können.

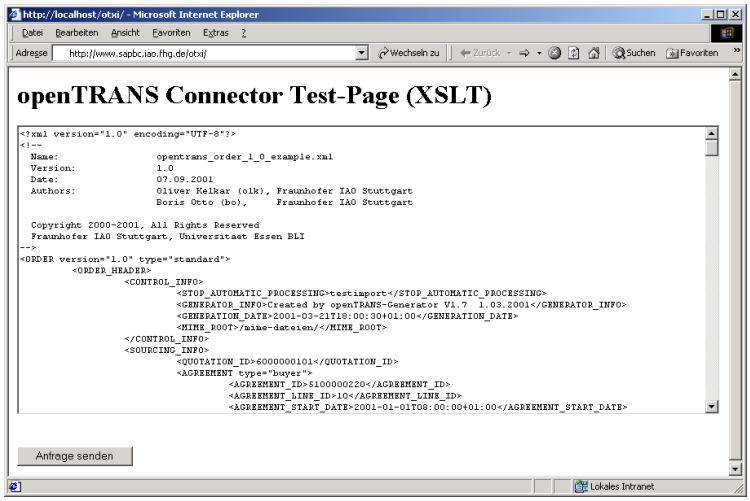

Abbildung 7-4 Simulation des Bestellvorgangs

Die Nachricht wird an den Business Connector geschickt, der ein im Rahmen dieser Arbeit entwickeltes Skript für die Verarbeitung vorhält. Die Nachricht wird als Parameter an dieses Skript übergeben.

7.3.2 Anlegen der Bestellung im SAP-System

Das für das Anlegen der Bestellung entwickelte Skript besteht aus drei Blöcken, die im Folgeden erläutert werden.

1. Empfang und Einlesen des Parameters als openTRANS-Dokument

Da der BusinessConnector beim Aufruf durch eine Webseite nur Parameter vom Typ „String“ entgegen nehmen kann, muss die Nachricht zuerst in einen Objekt vom Typ „Document“ konvertiert werden. Die verwendete Funktion „pub.web:stringToDocument“ wird vom BusinessConnector bereitbestellt und leistet die Konvertierung.

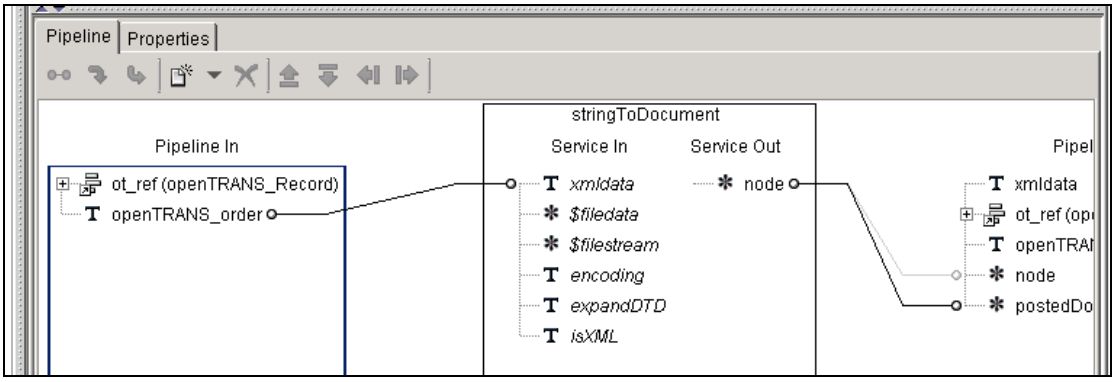

Abbildung 7-5 Konvertierung der eingelesenen Nachricht

Nach der Konvertierung steht die Bestellnachricht in der Pipeline des Flows als „Document“ zur Verfügung. Der Datentyp ist vergleichbar mit einem Objekt, dessen genaue Spezifikation noch unbestimmt ist.

Um das Objekt als openTRANS-Dokument verarbeiten zu können, sind verschiedene weitere Schritte nötig. Um dem BusinessConnector die Struktur eines openTRANS-Dokuments zugänglich zu machen, wurde aus dem openTRANS-Schema der Bestellnachricht ORDER (vgl. 5.2) eine für den BusinessConnector lesbarere Datenstruktur vom Typ „Record“ erzeugt.

Vor der Erzeugung musste das offizielle openTRANS-Schema in Teilen modifiziert werden, da der Business Connector nur eine Teilmenge der vollständigen Schema-Spezifikation [W3C-XSD2001] versteht. In Folge dessen mussten alle XML-Namensraum-Deklarationen [W3C-NS2001] aus dem Schema entfernt werden.

Die folgende Abbildung zeigt die aus dem Schema erzeugte Struktur vom Typ „Record“.

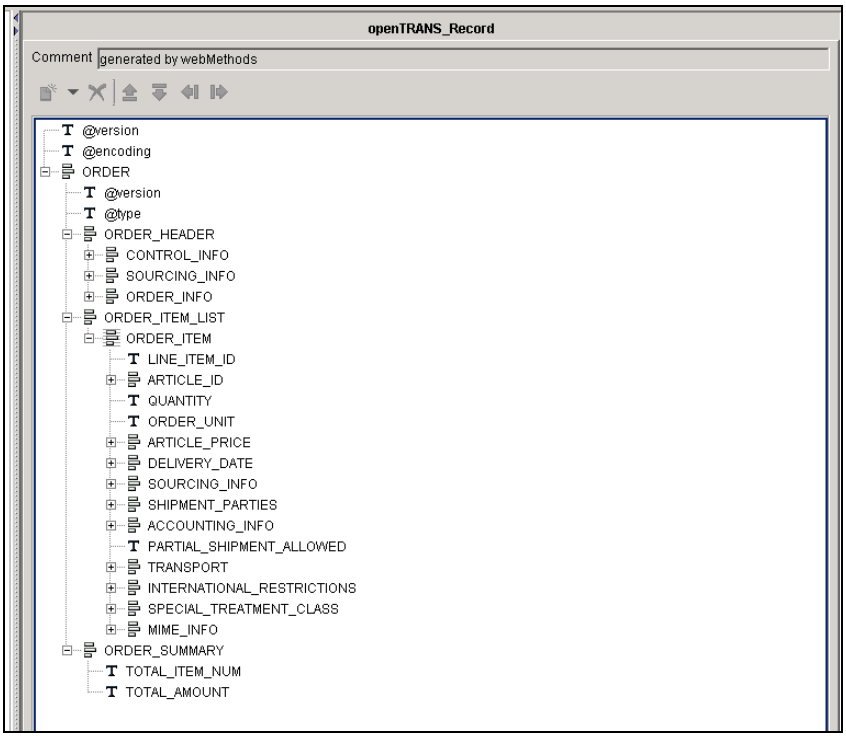

Abbildung 7-6 Erzeugte openTRANS-Struktur

Das im ersten Schritt erzeugte Objekt vom Typ „Document“ kann nun durch eine Konvertierung (pub.web:documentToRecord) in den Typ „Record“ und eine anschließende Zuweisung zum Typ „openTRANS_Record“ in eine für den BusinessConnector lesbare openTRANS-Struktur überführt werden.

Die übermittelte Nachricht steht somit als Baumstruktur im Business Connector zur Verfügung und kann weiterverarbeitet werden. Durch die Zuweisung zum „openTRANS_Record“ kann im Anschluss auf jedes Element der Nachricht grafisch über die Baumstruktur zugegriffen werden.

2. Umsetzung / Überbrückung von Formatunterschieden

openTRANS und SAP-BAPIs sind zwei in unterschiedlichen Zeiten und zum Teil mit unterschiedlichen Anforderungen gewachsene Standards. Daher kann ein openTRANS-Dokument nicht 1:1 in eine für den BAPI-Aufruf benötigte Struktur übertragen werden.

Folgende Umsetzungen müssen durchgeführt werden:

Konvertierung	Bescheibung
Mengenangaben	SAP schreibt bei Mengenangaben auf Artikelebene die Angabe des tausendfachen Wertes vor. So muss bei der Angaben einer Bestellmenge von 5 Stück der Wert 5000 übergeben werden, um eine korrekte Interpretation im System zu erzeugen. Die Konvertierung wurde durch die Funktion „pub.math.mult“ realisiert, wobei alle Artikelpositionen in einer Schleife durchlaufen werden und der Wert des Feldes QUANTITY mit 1000 multipliziert wird. Die Multiplikation wird direkt in der Struktur durchgeführt, so das die openTRANS-Struktur nach Durchführung der Konvertierung jeweils die tausendfachen Bestellmengen enthält.
Auftragsart	Ein SAP-System unterscheidet zwischen verschieden Auftragsarten. Diese können im System konfiguriert werden und verschiedene Parameter in der Auftragsverarbeitung beeinflussen. SAP sieht standardmäßig 34 unterschiedliche Auftragsarten vor. Unter anderem die Auftragsarten „TA“ – Terminauftrag, „AA“ – Aktionsauftrag, „BV“ Barverkauf vor. Da openTRANS die Art der Bestellung nicht detailliert beschreibt, wird in der vorliegenden Realisierung standardmäßig ein Terminauftrag erstellt. Genauso wäre auch denkbar, das die Auftragart in Abhängigkeit von der Kundennummer gesetzt wird. So könnte für spezielle Unternehmen ein Sofortauftrag „SO“ erstellt werden, der im SAP-System bei der Reservierung von Ressourcen und Materialien mit Priorität behandelt wird.
Mengeneinheiten	Das openTRANS-Format schreibt die Angabe von

	Mengeneinheiten gemäß der internationalen UN/ECE-Norm [UNECE] vor. Darin sind verschiedenste Mengeneinheit auf generischen drei-Buchstaben-Folgen definiert. Die Einheit „Stück“ wird beispielsweise in UN/ECE als „C62“ definiert. In einem SAP-System hingegen lassen sich Mengeneinheiten vollkommen kundenspezifisch im Customizing-System anlegen und mit entsprechenden Auswirkungen für verschiedene Bereiche wie die Bestellverarbeitung, Kommissionierung, und Lagerhaltung belegen. Es obliegt dem Unternehmen selbst, welche Mengeneinheiten angelegt werden, und wie diese namentlich definiert sind. Im vorliegenden SAP IDES-System (Internet Demonstration and Education System) [SAP-IDES] ist beispielsweise die Bezeichnung „STK“ für „Stück“ hinterlegt. Bevor eine Bestellung angelegt werden kann, müssen daher die Mengeneinheiten auf die im SAP-System hinterlegten Einheiten angeglichen werden. Denkbar ist auch das Vorhalten von Konvertierungs-Tabellen, in denen für jede Mengeneinheit des Kunden die resultierende Mengeneinheit des Lieferanten hinterlegt ist. Dieses Vorgehen erfordert zum einen eine eigene Konvertierungstabelle für jeden Kunden des Unternehmens; zum anderen können bei der Umsetzung sehr leicht Interpretationsfehler auftreten. Beispielsweise definiert die Menge “Pack“ bezogen auf das Produkt Papier keine genaue Angabe: In der Regel wird der Einkäufer von der Lieferung von 500 Blatt Papier ausgehen. Genauso denkbar wäre aber ein Paket mit 250 Blatt oder auch nur 10 Blatt. Für die vorliegende Implementierung wurde auf eine Konvertierung der Mengeneinheiten verzichtet. Es wird stattdessen vorausgesetzt, das der Kunde eine beim Lieferanten hinterlegte Mengeneinheit in der Bestellung überträgt.
Vertriebsparameter	Ein SAP-System besitzt eine in Vertriebswege, Verkauforganisationen und Sparten untergliederte Unternehmensstruktur. Die Funktion zum Anlegen einer Bestellung erwartet diese Werte als Eingangparameter. Da in openTRANS keine Felder für die Übermittlung dieser Werte vorhanden sind, müssen Sie anderweitig ermittelt werden. Für dieses Szenario wurden die Werte durch feste Parameter gesetzt. Denkbar ist aber auch eine dynamische Ermittlung dieser Werte anhand der Kundendaten, die in der Bestellung mitübermittelt

	werden, oder die Fremdnutzung eines anderen Feldes (z.B. Kundennummer: Kundennummer / Vertriebsweg / Verkaufsorganisation / Sparte) wovon aber auf Grund der Standardkonformität Abstand genommen wird.

3. Aufruf des BAPIs zum Anlegen der Bestellung im SAP System

Nachdem die Überbrückung der Formatunterschiede durchgeführt ist, kann das BAPI „salesorder_createfromdat“ aufgerufen werden, um die Bestellung im System anzulegen.

Die Elemente der openTRANS-Record Struktur werden dazu an die Eingangs-Parameter des BAPI übergeben. Dies geschieht durch Zuordnung der openTRANS-Elemente zu den Elementen des BAPIS.

Eine detaillierte Übersicht der vorgenommenen Verknüpfungen findet sich im Anhang A1.

7.3.3 Erzeugung der Bestellbestätigung

Nachdem die Bestellung im System angelegt ist, werden verschiedene Prüfungen durchgeführt. Dazu zählen die Gültigkeit der Bestellparameter, die Existenz der bestellten Artikel, die Verfügbarkeit der bestellten Mengen etc.

Anschließend wird die Bestellung als gültig markiert und eine Bestätigung erzeugt. Diese wird von der SAP-Ausgangsverarbeitung im SAP IDoc-Format an den BusinessConnector gesendet.

Für das vorliegende Szenario wurde der im BusinessConnector enthaltene Partner Manager so konfiguriert, das in diesem Fall der Flow zur Erzeugung der openTRANS-Bestellbestätigung aufgerufen wird. Dies wurde durch Hinzufügen einer Regel im Partner Manager erreicht, die eine Übereinstimmung mit der Nachricht in folgenden Parametern besitzen muss:

Element	Gesetzter Wert
Sender	C11 – Dies ist die Kennung die das SAP-System im vorliegenden Prototypen besitzt. Der Eintrag schreibt vor, das die Regel nur für Nachrichten zutrifft, die vom SAP-System gesendet werden
Empfänger	7777 – Dies ist die Kundennummer des im Szenario verwendeten Beispielunternehmens. Der Eintrag filtert Nachrichten, die an diesen Kunden gesendet werden
Nachricht	ORDERS02 – Dies ist der Nachrichtentyp der IDoc-Bestellbestätigung (OrderResponse).

Als Transport wird die Ausführung des Flows zur Erzeugung der Bestellbestätigung definiert. Dieser gliedert sich in verschiedene Teile, die im Folgenden näher erläutert werden sollen:

1. Empfang und Einlesen des IDoc

Beim Aufruf des Flows übergibt der BusinessConnector das vom SAP-System gesendete IDoc in der Pipeline des Skripts. Um das IDoc aus der Pipeline herauszulesen, wird die Funktion „pub.sap.IDoc:transformFlatToHierarchy“ verwendet. Sie liest die in der Pipeline liegende IDoc-Nachricht ein und konvertiert Sie in eine Struktur vom Typ „Record“.

Die Nachricht liegt als Objekt vom Typ „Record“ vor, der jedoch noch unbestimmt und somit nicht für den Business Connector lesbar ist. Der BusinessConnector hat keine Informationen darüber, welche Art bzw. welcher IDoc-Typ vorliegt.

Um das IDoc als Bestellbestätigung „ORDERS02“ kenntlich zu machen, wurde die Struktur der Bestellbestätigung im BusinessConnector angelegt. Diese kann über den BusinessConnector aus dem SAP-System herausgeladen werden und in eine „Record“-Struktur gespeichert werden.

Durch anschließende Zuweisung des IDoc zu dieser Struktur wird der Zugriff für die weiteren Schritte des Programms ermöglicht.

2. Umsetzung / Überbrückung von Formatunterschieden

Vor der Erzeugung der openTRANS-Bestellnachricht müssen die artikelpositionsbezogenen Mengenangaben wieder mit dem Divisor 1000 umgerechnet werden. Dazu wird, wie bereits in der Eingangsverarbeitung als Multiplikation durchgeführt, in einer Schleife über alle Artikelpositionen die Funktionen „pub.math.div“ verwendet. Der Einfachheit halber wird mit dem Ergebnis der Division der alte Wert überschieben, so das nach der Konvertierung in allen Mengenangaben mit dem Divisor 1000 geteilte Mengenangabe stehen.

3. Erzeugen der openTRANS-Bestellbestätigung

Für die Erzeugung der Bestellbestätigung (ORDERRESPONSE) wurde aus dem openTRANS-Schema eine für den Business Connector lesbare „Record-Stuktur“ erzeugt. Dies hat den Vorteil, das die XML-Baumstruktur der Bestellbestätigung im Business Connector zur Verfügung steht, und mittels Verknüpfungen mit Werten gefüllt werden kann.

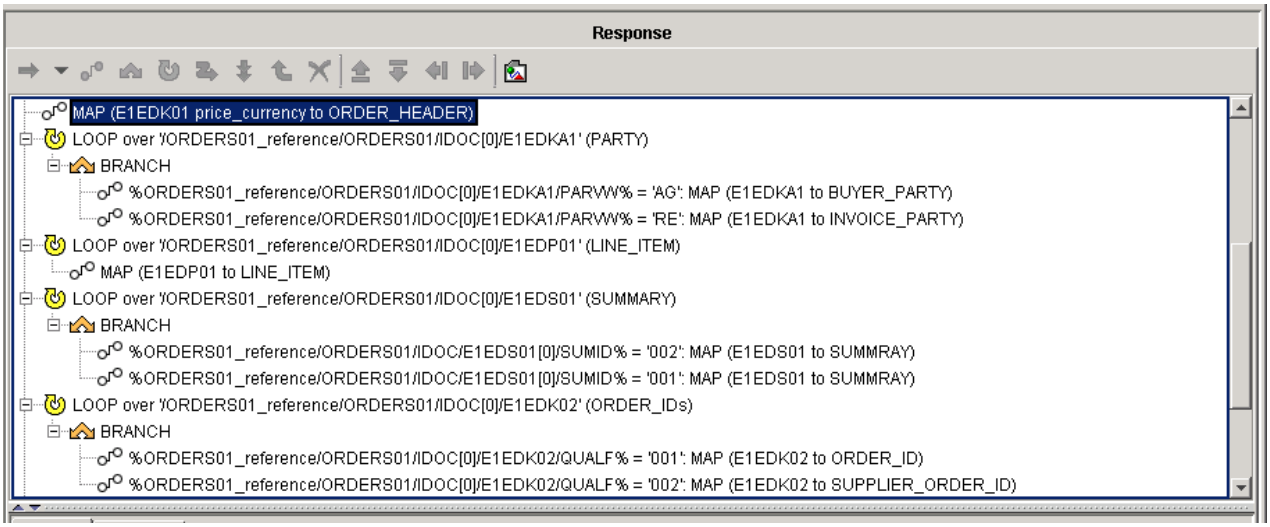

Abbildung 7-7 Erzeugen der openTRANS Bestellbestätigung aus den IDoc-Elementen

Die vollständige Beschreibung der Feldzuordnungen findet sich im Anhang A2

4. Senden der Bestellbestätigung an den Kunden

Nachdem die Bestellbestätigung vollständig erzeugt ist, bieten sich mehrere Wege an, diese an den Kunden zu übermitteln. Für das Szenario wurde das Senden einer Email verwendet.

Die Bestellung muss dazu im Bereich „ADDRESS“ des einkaufenden Unternehmens die Email-Adresse im Feld EMAIL übertragen, welche für die Sendung der Bestellbestätigung verwendet wird. In Folge dessen konfigurier sich der Transport für die Bestellbestätigung automatisch aus der bei der Bestellung übermittelten Partnerdaten.

Zum Senden von Emails per SMTP bietet der Business Connector die Standard-Funktion „pub.client:smtp“ an. Diese wurde mit Standard-Parametern für Betreffzeile, Absenderadresse, etc. konfiguriert und sendet die openTRANS-Bestellbestätigung an den Kunden.

7.4 Realisierung mittels XSLT

Während bei der Realisierung mittels BAPIs der Aufruf der BAPI-Funktion zum Anlegen der Bestellung im Mittelpunkt steht, verwendet die folgende Realisierung eine XSLT-Transformation zur Umsetzung von openTRANS auf SAP-IDoc.

7.4.1 Senden einer Nachricht

Wie in der vorangegangenen Realisierung erzeugt das einkaufende Untenehmen eine Bestellung im openTRANS-Format (vgl. Kapitel 5.2) und sendet diese per HTTPS-Aufruf an das System des Lieferanten. Dazu vereinbaren Einkäufer und Lieferant eine URL-Adresse, an welche die Nachricht gesendet wird.

Die Nachricht wird an den Business Connector geschickt, der ein spezielles Skript für die Verarbeitung der Nachricht vorhält. Die Nachricht wird als Parameter an dieses Skript übergeben.

7.4.2 Transformation der Bestellung

Der Flow zum Anlegen der Bestellung muss die Nachricht zuerst in ein für die weitere Verarbeitung geeignetes internes Format bringen. Auf diesem Format kann die XSLT-Transformation der openTRANS-Nachricht zu einem SAP IDoc durchgeführt werden. Das resultierende, transformierte Dokument wird für den Transport zum System in ein SAP-Format codiert und schließlich an das System übergeben.

1. Empfang und Einlesen des Parameters als openTRANS-Dokument

Wie bei der Realisierung mittels BAPIs übergibt der Business Connector die per HTTPS empfangene openTRANS-Nachricht als Parameter vom Typ „String“ an den Flow. In Folge dessen muss auch hier die Nachricht zuerst in einen Objekt vom Typ „Document“ konvertiert werden. Die verwendete Funktion „pub.web:stringToDocument“ leistet diese Konvertierung.

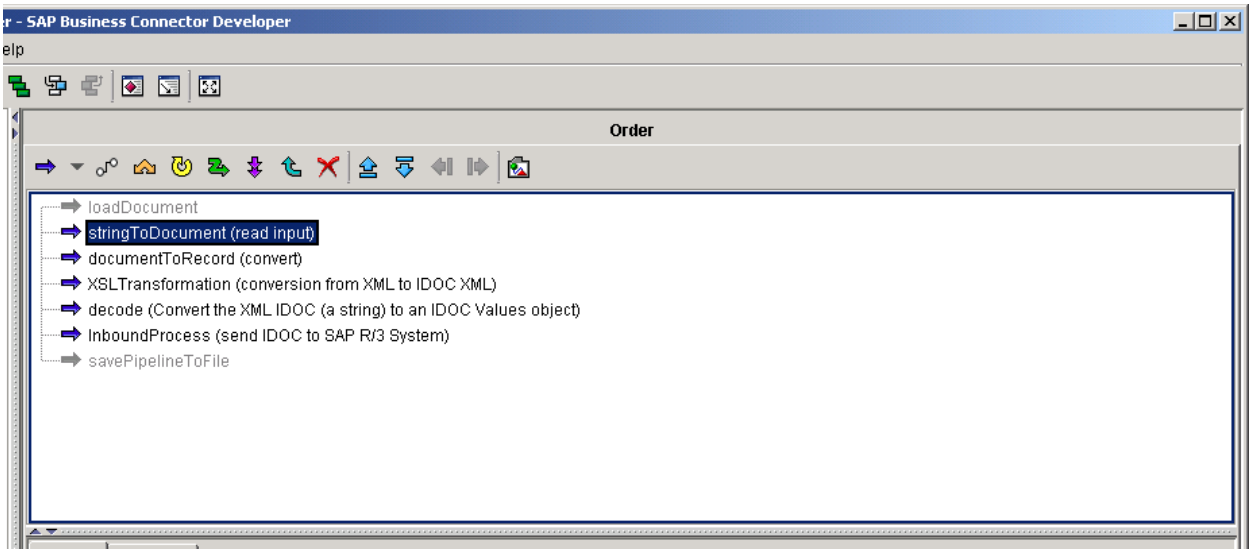

Abbildung 7-8 Flow für die XSLT-Realisierung

Nach der Konvertierung steht die Bestellnachricht in der Pipeline des Flow als „Document“ zur Verfügung. Da für den Aufruf der XSLT-Transformation ein Objekt vom Typ „Record“ benötigt wird, wird eine weitere Konvertierung durchgeführt.
Im Gegensatz zur vorangegangenen Realisierungsalternative ist es nicht notwendig, dem Business Connector die Nachricht als openTRANS-Nachricht kenntlich zu machen. Da lediglich eine Transformation des Dokuments vorgenommen wird, und nicht einzelne Felder und Attribute der Struktur zugreifbar sein müssen, kann die eigentliche Struktur der eingelesenen Nachricht vor der Software verborgen bleiben.

2. Transformation von openTRANS zu IDoc

Für die XSLT-Transformation werden das Objekt vom Typ „Record“, das die gesendete openTRANS-Nachricht enthält, sowie der Name des zu verwendenden XSLT-Stylesheets an die Funktion „pub.sap.xslt.transformation:XSLTransformation“ übergeben. Das Stylesheet wird zuvor auf einem festgelegten Pfad im Arbeitsverzeichnis des BusinessConnectors bereitgestellt.

Im folgenden sollen einige Herausforderungen, die bei der Erstellung des XSLT-Stylesheets zu bewältigen waren, detaillierter erläutert werden.

Herausforderung	Bescheibung
Positionsnummern	Ein im Hinblick auf die Stornierung von Bestellungen wichtiger Aspekt ist die Speicherung einer Bestellpositionsnummer, die in einer openTRANS-Bestellung als „LINE_ITEM_ID“ übermittelt wird. Sie dient bei der Stornierung von Teilen einer Bestellung als eindeutiger Identifikator. SAP bietet für diesen Zweck das Feld „POSEX“ an. Es ist vom SAP-System nicht vorgesehen, die in einer Bestellung übermittelten Positionsnummern zu speichern. Das System generiert eine eigne fortlaufende Nummer für jede Position. Bei der Bestellbestätigung werden in Folge dessen nicht die in der Bestellung übermittelten

	Bestellpositionsnummern verwendet, sondern vom SAP-System erstellte Nummern.
Datumsformate	SAP verwendet bei der Übermittlung von Nachrichten das Datumsformat tt.mm.jjjj. Das openTRANS-Format verwendet die Zeitangabe gemäß ISO 8601 Datetime [W3C-DT] jjjj-mm-tt. Die Umsetzung zwischen den beiden Formaten wurde mittels der XSLT-„Substring"-Funktion durchgeführt.
Festwerte	Wie bei der Realisierung mittels BAPIs werden gewisse für das SAP-System notwendige Angaben in einer openTRANS-Bestellung nicht übermittelt. Dazu gehören die SAP-Unternehmensstruktur-Parameter (Verkaufsorganisation, Vertriebsweg und Sparte) . Im Gegensatz zur Realisierung mittels BAPIs, wird in der Realisierung mittels XSLT eine IDoc-Nachricht erzeugt. Ein IDoc muss für die ALE-Eingangsverarbeitung gewisse Parameter, wie den Namen des logischen Systems übermitteln, von dem Sie geschickt worden ist. Sie muss Nachrichten-Typ und den Business Objekt-Namen übermitteln.

Eine genaue Aufstellung der Mappings findet sich im Anhang B1.

3. Übergabe des IDoc an das SAP-System

Da SAP selbst keine XML-Dokumente entgegennehmen kann muss das IDoc, welches nach der Transformation in einer XML-Darstellung vorliegt in eine äquivalente interne SAP-Struktur (ASCII-Format) überführt werden.

Der Business Connector bietet dazu die Funktion „pub.sap.IDoc:decode", die aus einem XML-IDoc-Dokument die entsprechende interne Darstellung erzeugt.

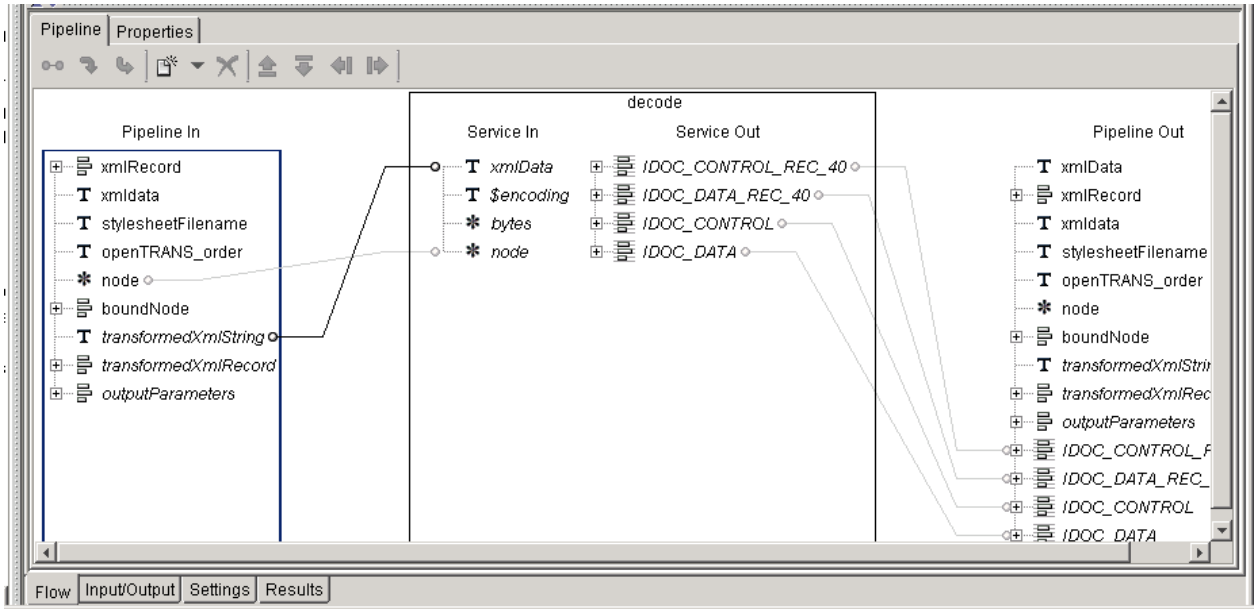

Abbildung 7-9 Erzeugung des XML-IDocs

Nach dem Aufruf der Funktion steht das IDoc unterteilt in seinen Kopfdatenteil „IDOC_CONTROL“, der allgemeine Verarbeitungsinformationen enthält, und einen Datenteil „IDOC_DATA“, der die Positionszeilen, Partnerinformationen etc. enthält, zur Verfügung.

Ausgehend von dieser Struktur kann das Dokument mit Hilfe der Funktion „pub.sap.transport.ALE:InboundProcess“ an das SAP-System gesendet werden.

7.4.3 Transformation der Bestellbestätigung

Da die eingehende Bestellung im Business Connector nur eine Formatumwandlung erfahren hat, wurde bei der Erstellung der Bestellbestätigung dieser Sachverhalt ausgenutzt, und der Flow für die Erzeugung der Bestellbestätigung als ein „Pre-Processing“-Service erstellt.

In Folge dessen kann der im Business Connector integrierten Partner Manager den Versand der Bestellbestätigung übernehmen. Da der Partner Manager die Bestellbestätigung vom SAP-System als IDoc erhält, wird die Transformation zu einer entsprechenden openTRANS-Nachricht durch einen Flow, der als Pre-Processing Service, implementiert ist, durchgeführt, und so vor dem Versand der Nachricht ausgeführt.

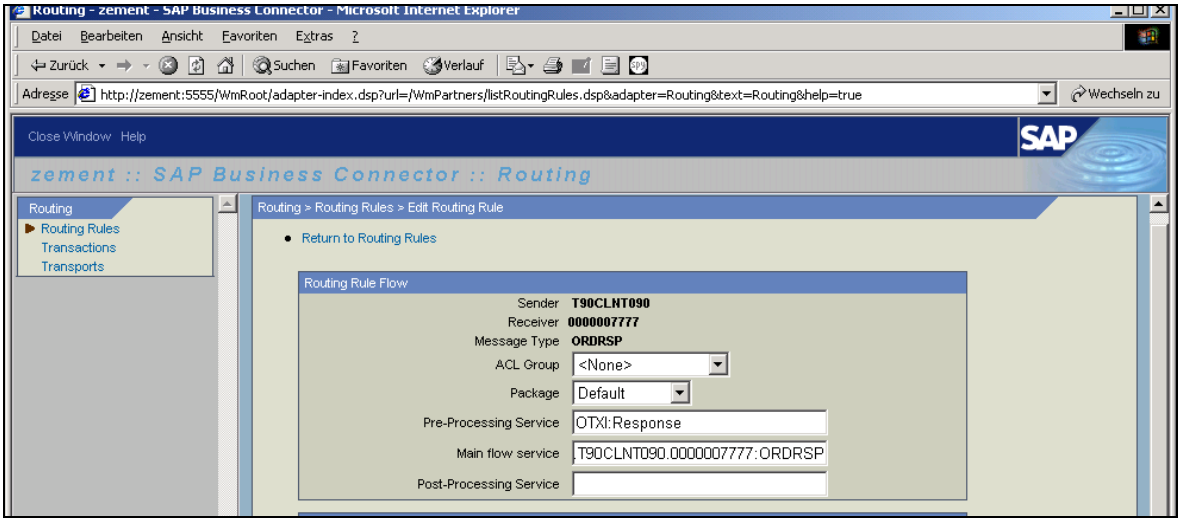

Abbildung 7-10 Konfiguration des Flow als Pre-Processing Service

Die Realisierung des Flow gliedert sich in die drei Schritte Eingangskonvertierung, Transformation und Übergabe. Da der Flow zum Pre-Processing eingesetzt wird, gibt er nach seiner Ausführung einen definierten Rückgabewert vom Typ „Record“ an den Partner Manager zurück.

1. Empfang und Einlesen des IDoc

Beim Aufruf des Flow übergibt der Business Connector das vom SAP-System gesendete IDoc in der Pipeline des Skripts. Um das IDoc aus der Pipeline herauszulesen, wird die Funktion „pub.sap.IDoc:transformFlatToHierarchy“ verwendet. Sie liest die in der Pipeline liegende IDoc-Nachricht und konvertiert Sie in eine Struktur vom Typ „Record“.

2. Transformation von IDoc zu openTRANS

Für die XSLT-Transformation wird, wie auch schon bei der Eingangskonvertierung des openTRANS-Dokuments, die Funktion „pub.sap.xslt.transformation:XSLTransformation“ genutzt. Dieser wird die IDoc-Nachricht als „Record“, sowie der Dateiname des entsprechenden XSLT-Stylesheets übergeben. Rückgabewert der Funktion ist ein String, der die Bestellbestätigung als openTRANS-Orderresponse enthält.

Die Herausforderungen waren wie bei der Eingangsverarbeitung der openTRANS Bestellung zu bewältigen, und werden daher nicht gesondert aufgeführt.

Eine genaue Aufstellung der Mappings findet sich im Anhang B2.

3. Übergabe der openTRANS-Nachricht an den Business Connector

Aus der Sicht des Business Connectors stellt der Flow eine Funktion dar, die aus einem IDoc eine openTRANS-Bestellbestätigung erzeugt. Der Rückgabewert des Flow ist ein Objekt vom Typ „Record“.

In Folge dessen muss als letzter Verarbeitungsschritt der Rückgabewert gesetzt werden. Dies erfolgt durch eine Zuweisung des XSLT-Transformations-Outputs an den Rückgabewert. Da beide Werte vom selben Typ sind, muss keine weitere Konvertierung erfolgen.

4. Senden der Bestätigung im Partner Manager

Nach Durchführung der Schritte 1-3 wird die Ausführung des Flow beendet, und der Rückgabewert, der die konvertierte openTRANS-Nachricht enthält, an den Partner Manager zurück übergeben.

Dieser führt nun den Versand der Nachricht durch. Der Partner Manager bietet eine Reihe vorkonfigurierter Transportarten. Darunter befindet sich auch der Email-Versand. Die Transportparameter wurden mit Standard-Werten gesetzt.

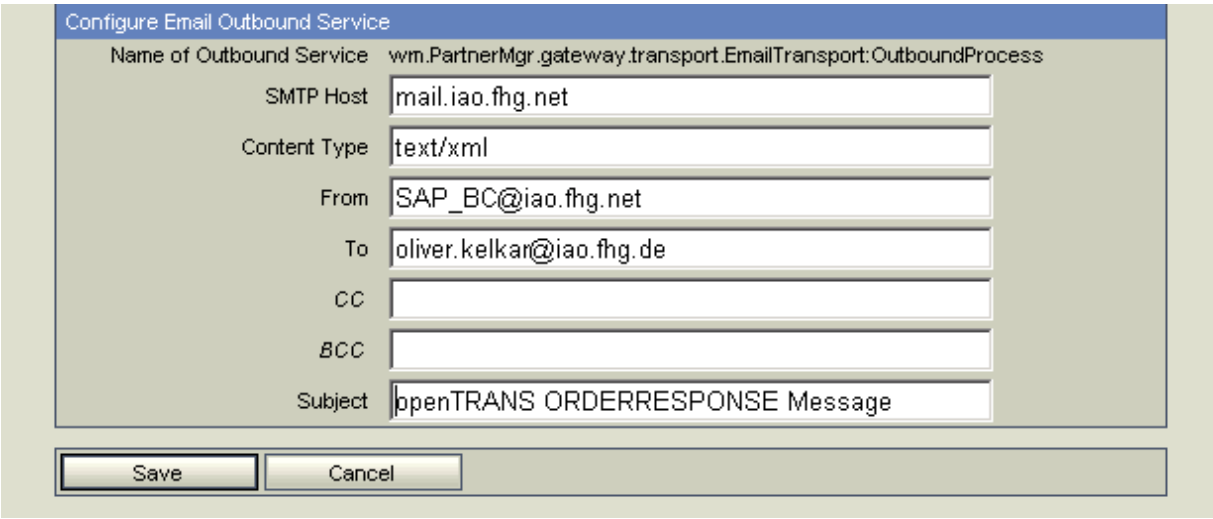

Abbildung 7-11 Konfiguration des Transport-Parameter

Anzumerken ist hier, das der Versand nicht an die in der Bestellung übermittelte Mailadresse erfolgt, sondern an die im Partnermanager hinterlegte Adresse erfolgt. Dies hat zur Folge, das

für jeden Kunden eine Regel im Partnermanager angelegt werden muss, die seine Transportdaten enthält.

7.5 Bewertung

Beide vorgestellten Realisierungen ermöglichen die vollautomatische Anbindung von Kunden über einen openTRANS-basierenden Nachrichtenaustausch. Beide Realisierungen laufen in Form von Flow-Services im SAP Business Connector ab und kommunizieren mit dem SAP-System.

Für eine Bewertung und Differenzierung der Realisierungen mittels BAPI (Kapitel 7.3) und XSLT (Kapitel 7.4) wurden folgende Kriterien mit entsprechender Relevanz für eine Integration erarbeitet:

Kriterium	Bescheibung	Rele-vanz	BAPI	XSLT
Integrations-aufwand	Bezüglich des Integrationsaufwands kann die Lösung mittel BAPIs besser bewertet werden. In der Realisierung mittels XSLT muss für jeden Geschäftspartner eine eigene Regel im Partnermanager angelegt werden, und mit seinen Transportdaten konfiguriert werden. Die Realisierung mittels BAPIs ermittelt die Email-Adresse selbsttätig. Für einen neuen Geschäftspartner muss daher keine neue Konfiguration erfolgen.	hoch	+	-
Flexibilität / Anpassungs-fähigkeit	Insbesondere bei der Überbrückung von Formatunterschieden stellten sich beide Realisierungen als flexibel und anpassungsfähig heraus. Konvertierungen, Formatanpassungen und flexible Mappings beherrschen beide Ansätze. Die Realisierung mittels XSLT verwendet in der Formatumsetzung den Industriestandard XSLT, für den weit aus mehr Dokumentation und Hilfestellung geboten wird, als für die auf Flowsteps basierende grafische Programmierung des Business Connectors.	hoch	o	+
Komplexität / Wartbarkeit	Die grafische Programmierung mittels Flowsteps eignet sich, um nach kurzer Einarbeitungszeit Resultate zu erzeugen. Bei der Erstellung von komplexen Konstrukten wird jedoch die Struktur der Flowsteps leicht unübersichtlich und fehleranfällig. Einige Eigenheiten bzw. Besonderheiten der Flowsteps (z.B. das automatische Zwischenpuffern von Stylesheets) sind zum Teil unzureichend Dokumentiert und nur schwer ausfindig zu manchen.	hoch	-	+
Reproduktions-fähigkeit /	Um die erarbeiteten Realisierungen in anderen Unternehmen(-teilen) zum Einsatz zu bringen,	mittel	o	+

Übertragbarkeit	müssen bei der Realisierung mittels BAPIs im Flow, der für die Eingangsverarbeitung zuständig ist, die Festwerte entsprechend des Unternehmens angepasst werden. Bei der Realisierung mittels XSLT sind die Festwert in den Stylesheets hinterlegt. Bei einer Betrachtung des Aufwands für eine derartige Anpassung kann die Realisierung mittels XSLT aufgrund Ihrer XSLT-Standardkonformität als einfacher bewertet werden.			

Zusammenfassend ist die Flexibilität, Komplexibilität, Zukunftsoffenheit und Übertragbarkeit bei der Realisierung mittels XSLT in höherem Maß gewährleistet. Bezüglich des Integrationsaufwands ist die Realisierung mittels BAPIs geeigneter.

8 Umsetzung synchroner Nachrichtenaustausch

8.1 Beschreibung eines Preisanfrage-Systems

Häufig betreiben Unternehmen digitale Kataloge, aus denen Bestellungen generiert werden. Dies sind zum einen Kataloge auf der Seite des einkaufenden Unternehmens (eProcurement-Systeme) oder auch Kataloge auf der (Web-)Seite des Lieferanten (eShops).

In beiden Fällen kann nach Produkten gesucht werden, ein oder mehrere Produkte in einen virtuellen Warenkorb gelegt werden und schließlich eine Bestellung generiert werden.

Vielfach werden ein einem Katalog spezielle in Rahmenverträgen verhandelte Preise gelistet, die abhängig von der bestellten Menge, aktuellen Rabattaktionen und weiteren Parametern sind. In Folge dessen werden häufig Kataloge ausgetauscht, um die Preise aktuell zu halten.

Ein anderer Ansatz verfolgt ein sog. PunchOut-Verfahren, bei dem der exakte Preis nicht im Katalog hinterlegt ist, sondern bei einer Bestellung dynamisch ermittelt wird. Dabei findet ein „Absprung“ vom Katalog zum Warenwirtschaftsystem des Lieferanten statt, das einen Preis für den Kunden in Abhängigkeit von speziell verhandelten Rabatten, der ausgewählten Bestellmenge und weiteren Parametern berechnet und an den Einkäufer zurücksendet.
Auf diese Art berechnete Preise sind stets exakt, aktuell und kundenindividuell.

Das Katalog-System des Kunden sendet eine Preisanfrage-Nachricht in Form einer Angebotsanfrage an das System des Lieferanten. Das System verarbeitet die Angebotsanfrage und erstellt auf Basis der verhandelten Rahmenverträge ein Angebot über die angefragten Produkte mit den berechneten Preisen.

Der gesamte Prozess läuft synchron ab. Die Datenübermittlung findet über HTTPS in Verbindung mit einer HTTP-Authentifizierung statt. Das aus der Angebotsanfrage erzeugte Angebot wird direkt als Antwort auf die Anfrage an das System des Einkäufers zurückgesendet.

Das System des Lieferanten, für das im Folgenden eine Implementierung vorgestellt wird, ist ein SAP R/3 System mit vorgeschaltetem SAP BusinessConnector. [SAP-BC2003] Dabei ist der Business Connector die über das Internet erreichbare zentrale Schnittstelle des Lieferanten nach Außen. Er kommuniziert mit dem Katalog-System des Einkäufers und ermittelt die Preise aus dem SAP-System des Lieferanten. Als eBusiness-Standard wird openTRANS für die eingehende und ausgehende Kommunikation verwendet.

8.2 Systemkonzept

Der Aufbau wird durch Abbildung 8.1 verdeutlicht. Dargestellt ist die Übermittlung einer Angebotsanfrage im openTRANS-Format (RFQ) (vgl. Kapitel 5.3) über das Internet. Die Anfrage wird per HTTPS-Aufruf an ein dafür vorgesehenes Programm des Business Connectors übergeben.

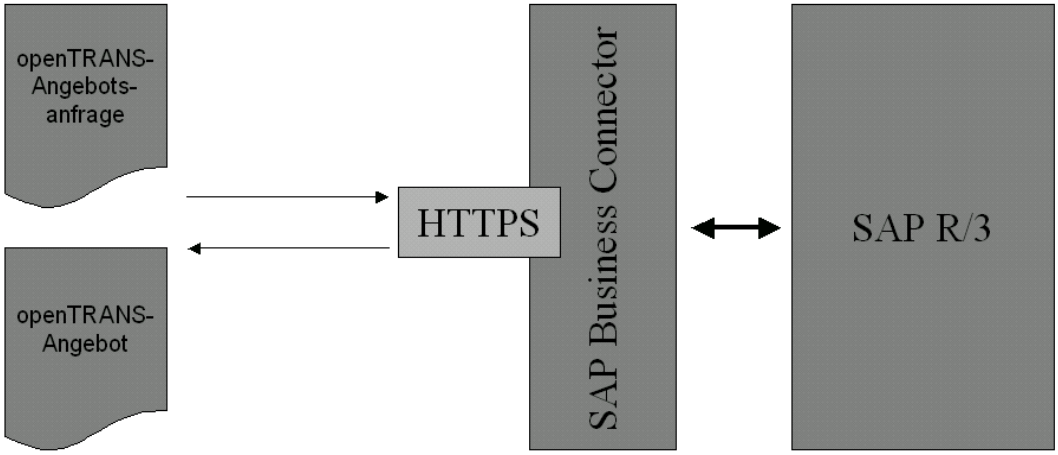

Abbildung 8-1 Systemkonzept

Das Programm ermittelt für die in der Nachricht angefragten Produkte aus dem angeschlossenen SAP-System die entsprechenden Preisinformationen. Dazu werden die in der Anfrage enthaltenen Kundendaten, Produktnummern und Angebotsmengen als Aufrufparameter an ein BAPI übergeben. Diese ermittelt im SAP-System die Preise und gibt diese als Rückgabewerte aus.

Die Rückgabewerte werden in eine openTRANS-Angebots-Nachricht (QUOTATION) konvertiert und als Ausgabe direkt als Antwort auf die Anfrage an den Kunden gesendet.

8.3 Realisierung

Bei der Realisierung steht der Aufruf des BAPIs zur Preisermittlung im Mittelpunkt. Das verwendete BAPI simuliert eine Bestellung und erhält als Rückgabeparameter die berechneten Preise.

8.3.1 Senden einer Nachricht

Wie in den vorangegangenen Realisierungen erzeugt das Katalog-System des einkaufende Untenehmens eine Angebotsanfrage im openTRANS-Format und sendet diese per HTTPS-Aufruf an das System des Lieferanten. Dazu wird zwischen Einkäufer und dem Lieferant eine URL-Adresse vereinbart, an welche die Nachricht gesendet wird.

Die Nachricht wird, wie in den vorangegangenen Realisierungen (vgl. Kapitel 7), an den Business Connector geschickt, der ein spezielles Skript für die Verarbeitung der Nachricht vorhält. Die Nachricht wird als Parameter an dieses Skript übergeben.

8.3.2 Preisermittlung

Die Ermittlung der Preise gliedert sich in vier logische Blöcke:

1. **Einlesen und Vorbereiten der Nachricht**

   Das Einlesen und Vorbereiten der Nachricht erfolgt analog zu der in Kapitel 7.3 beschriebenen Realisierung mittels BAPIs.

2. **Berechnung der Preise**

   Das verwendete BAPI „SalesOrder.simulate“ entspricht seiner Aufrufnotation exakt dem bei der unter Kapitel 7.3 beschriebenen Realisierung mittels BAPIs. Daher wird der Aufruf an dieser Stelle nicht weiter ausgeführt.
   Das BAPI simuliert eine Bestellung, in dem es die Bestellung vollständig anlegt, die Daten jedoch nicht speichert. Die so ermittelten Preise sind exakt die, die bei einer Bestellung gelten würden.

3. **Erstellung des openTRANS-Angebots**

   Für die Erzeugung des Angebots (OUOTATION) wurde aus dem openTRANS-Schema eine für den Business Connector lesbare „Record-Stuktur“ erzeugt. Dies hat den Vorteil, dass die XML-Baumstruktur der Bestellbestätigung im Business Connector zur Verfügung steht, und mittels Verknüpfungen mit Werten gefüllt werden kann.
   Die vollständige Beschreibung der Feldzuordnungen findet sich im Anhang C1.

4. **Ausgabe des openTRANS-Angebots**

Im Business Connector kann die Art einer Rückgabe über Templates konfiguriert werden. In einem Template können alle Variablen der Pipeline ausgegeben werden. Über das Ausgabeformat kann festgelegt werden, ob die Ausgabe in XML, HTML oder WDSL erfolgt.

Für die Ausgabe des openTRANS-Angebots wurde im Template die Ausgabe der entsprechenden Pipeline-Variable konfiguriert und als Format XML festgelegt.

8.4 Bewertung

Die vorgestellte Realisierung ermöglicht die automatische Angebotserstellung auf Basis von verhandelten Rahmenverträgen. Die Realisierung setzt auf den eBusiness-Standard openTRANS auf und ermöglicht daher eine einfache Integration in Katalogsysteme.

Preise werden aktuell, kundenindividuell und gemäß allen verhandelten Konditionen ermittelt. Der Prozess läuft synchron und bringt daher nur geringe Verzögerungen für den Katalogbenutzer.

9 Ergebnisse und Ausblick

Das in der vorliegenden Arbeit entwickelte Integrationskonzept für zwischenbetriebliche Geschäftstransaktionen ermöglicht die vollständig automatisierte Kommunikation auf Basis offener XML-Standards zur Abwicklung von Bestellprozessen.

Es wurde eine Architektur für die Integration entwickelt, in der die Integrationssoftware SAP Business Connector einem SAP R/3-System vorgeschalten wird, und das R/3-System um zusätzliche Schnittstellen erweitert.

Im Business Connector wurden Flows erstellt, welche die Integration des offenen XML-Standards openTRANS ermöglichen. Exemplarisch wurde für einen Bestelldatenaustausch und eine Online-Preisanfrage eine Realisierung vorgenommen.

Dabei wurden der asynchrone wie auch der synchrone Datenaustausch umgesetzt. Die Realisierung des asynchronen Austauschs wurde mittels zwei unterschiedlicher Technologien implementiert.

Für eine Umsetzung mittels der BAPI-Technologie wurden Konvertierungsfunktionen und Mappings erstellt. Bei Verwendung der XSLT-Technologie wurden Stylesheets erstellt und in einem Flow zur Ausführung gebracht.

Eine Bewertung der verwendeten Technologien zeigte, das die XSLT-Technologie die für eine Integration relevanten Kriterien Flexibilität, Wartbarkeit und Übertragbarkeit auf andere Nachrichtentypen in höherem Maß erfüllt.

Ein Bedarf für den Ausbau der vorgestellten Realisierung besteht in Hinsicht auf die dynamische Ermittlung von SAP R/3 spezifischen Vertriebsparametern, die in der vorliegenden Arbeit mit Festwerten belegt wurden. Der Realisierte Ansatz eignet sich für einen Unternehmensstandort, jedoch nicht für ein zentrale Integration von verschiedenen Unternehmensstandorten.

Für die in einer Nachricht übermittelten Mengenangaben konnte keine Konvertierung in die in einem SAP-System hinterlegten Angaben erfolgen. Hierfür ist eine individuelle Konvertierung für jeden Kunden eines Unternehmens zu entwickeln.

Bei der vorgestellten Realisierung können die im openTRANS-Format eindeutigen Positionsnummern nicht von SAP übernommen werden. Um dies zu ermöglichen ist das SAP-System im Quellcode der Eingangsverarbeitung entsprechend zu modifizieren.

10 Bibliografie

[ebXML2003] Frømyr, J.: Die Definition von Geschäftsprozessen, 21.10.2003, http://www.ebxml.eu.org/Deutsch/Gesch%C3%A4ftsprozesse.htm

[Janko1999] Janko: SAP R/3 Schnittstellentechnologie, Seminararbeit WS89/99,Wirtschaftsuniversität Wien, http://miss.wu-wien.ac.at/~mitloehn/se/w98/projekte/j9425343/api_bapi.htm

[Kaftan2000] Kaftan, H.: Betriebliche Standardsoftware für KMU, Universität Leipzig, 2000, http://www.iwi.uni-leipzig.de/d/download/Ka-camesp.pdf

[Kelkar2003] Kelkar, O.: Eine Referenzmodell für elektronische Geschäftstransaktionen im zwischenbetrieblichen Geschäftsverkehr, Stuttgart: Jost Jetter Verlag, 2003

[Mundt2003] Mundt, H.: Implementierung von Nachrichtenformaten in einem internationalen E-Procurement-Projekt bei Airbus, Diplomarbeit Universität Hamburg, 2003

[Risak2003] Risak, V.: Öffentliche Beschaffung über elektronische Netze (E-Beschaffung, Universität Salzburg, 2003

[SAP-Bapi2001] SAP AG: SAP Help Portal SAP R/3 Release 4.6C, 2001, http://help.sap.com/saphelp_46c/helpdata/de/ 5c/f3f0371bc15d73e10000009b38f8cf/frameset.htm

[SAP-BC2003] SAP AG: SAP Help Portal SAP R/3 Enterprise Release 4.7, 2003 http://help.sap.com/sapdocu/core/47x200/HELPDATA/EN/ 6f/1bd5c2a85b11d6b28500508b5d5211/frameset.htm

[SAP-IDoc1999] SAP AG: SAP Help Portal SAP R/3 Release 4.5B, 1999, http://help.sap.com/saphelp_45b/helpdata/de/ 72/c18ee5546a11d182cc0000e829fbfe/content.htm

[SAP-IDES] SAP AG: SAP Help Portal Internet Demo and Evaluation System – IDES -Das SAP-Modellunternehmen, 2003, http://help.sap.com/ saphelp_ides/helpdata/de/frameset.htm

[SAP-IFR2003] SAP AG: SAP Interface Repository, 2003, http://ifr.sap.com

[Schmitz1999] Schmitz, V.; Hümpel, C.; Renner, T.: Spezifikation BMEcat Version 1.01, 7.11.1999, http://www.bmecat.org

[Schmitz2001a] Schmitz, V.; Kelkar, O.; Pastoors, T.: Spezifikation BMEcat Version 1.2, 21.3.2001, http://www.bmecat.org

[Schmitz2001b] Schmitz, V.: BMEcat – Eine Anwendung des XML-

Standardformats, 13.09.2001, http://www.ec-ruhr.de/pdf/Vortrag-Schmitz-010913.pdf

[Schultz1997] Schultz, P.: Mythos Internet, Anmerkungen zur Netzkritik, Frankfurt/Main: Suhrkamp 1997

[Walter2001] Walter, M.: Beschreibung von Verfahren zur Anbindung von E-Commerce-Anwendungen an betriebswirtschaftliche Standard-Software und Evaluation durch eine beispielhafte Implementierung, 2001, Diplomarbeit Universität Leipzig, http://ais.informatik.uni-leipzig.de/download/sem_mmk/Michael_Walter_pdf.pdf

[W3C-XPath1999] World Wide Web Consortium (W3C), Clark, J.: XML Path Language (XPath), 16.11.1999 http://www.w3.org/TR/xpath

[W3C-DTD1999] World Wide Web Consortium (W3C), Ragget, D.: HTML 4.01 Specification, 24.12.1999, http://www.w3.org/TR/REC-html40/

[W3C-Schema2001] World Wide Web Consortium (W3C), Fallside, D.: XML Schema Part 0: Primer, 2.5.2001, http://www.w3.org/XML/Schema

[W3C-DT1997] World Wide Web Consortium (W3C), Wolf, M.;Wicksteed, C.: Date and Time Formats, 15.9.1997 http://www.w3.org/TR/NOTE-datetime

[W3C-NS1999] World Wide Web Consortium (W3C), Bary, T.; Hollander, D.: Layman, A.: Namespaces in XML, 14.1.1999, http://www.w3.org/TR/REC-xml-names/

11 Anhang A1 openTRANS-BAPI-Zuordungen

openTRANS (Quellformat)		BAPI SalesOrder.Createfromdat	
Kontext	Feld	Kontext	Feld / Wert
ORDER_HEADER	ORDER_ID	ORDER_HEADER_IN	PURCH_NO
ORDER_HEADER	CURRENCY	ORDER_HEADER_IN	PRICE_CURRENCY
BUYER_PARTY	-	ORDER_PARTNERS	“AG”
BUYER_PARTY	PARTY_ID	ORDER_PARTNERS	PARTN_NUMB
BUYER_PARTY	NAME	ORDER_PARTNERS	NAME
BUYER_PARTY	NAME2	ORDER_PARTNERS	NAME_2
BUYER_PARTY	STREET	ORDER_PARTNERS	STREET
BUYER_PARTY	DEPARTMENT	ORDER_PARTNERS	NAME_3
BUYER_PARTY	ZIP	ORDER_PARTNERS	POSTL_CODE
BUYER_PARTY	CITY	ORDER_PARTNERS	CITY
BUYER_PARTY	COUNTRY	ORDER_PARTNERS	COUNTRY
ORDER_ITEM	SUPPLIER_AID	ORDER_ITEMS_IN	MATERIAL
ORDER_ITEM	DESCRIPTION_SHORT	ORDER_ITEMS_IN	SHORT_TEXT
ORDER_ITEM	QUANTITY	ORDER_ITEMS_IN	REQ_QTY
ORDER_ITEM	ORDER_UNIT	ORDER_ITEMS_IN	UNIT

12 Anhang A2 Zuordnungen IDOC-openTRANS

IDoc ORDERS (Quellformat)		openTRANS	
Kontext	Feld	Kontext	Feld / Wert
E1EDK02	BELNR	ORDER_HEADER	ORDER_ID
E1EDK01	CURCY	ORDER_HEADER	PRICE_CURRENCY
E1EDKA1	PARVW ‘AG’	BUYER_PARTY	-
E1EDKA1	PARTN	BUYER_PARTY	PARTY_ID
E1EDKA1	NAME1	BUYER_PARTY	NAME
E1EDKA1	NAME2	BUYER_PARTY	NAME2
E1EDKA1	NAME3	BUYER_PARTY	STREET
E1EDKA1	STRAS	BUYER_PARTY	DEPARTMENT
E1EDKA1	ORT01	BUYER_PARTY	ZIP
E1EDKA1	PSTLZ	BUYER_PARTY	CITY
E1EDKA1	LAND1	BUYER_PARTY	COUNTRY
E1EDP19	IDTNR	ORDER_ITEM	SUPPLIER_AID
E1EDP19	KTEXT	ORDER_ITEM	DESCRIPTION_SHORT
E1EDP19	MENGE	ORDER_ITEM	QUANTITY
E1EDP19	MENGEE	ORDER_ITEM	ORDER_UNIT

13 Anhang B1 openTRANS-IDoc XSLT Schablonen

```
<?xml version="1.0" encoding="UTF-8"?>
<xsl:stylesheet version="1.0" xmlns:xsl="http://www.w3.org/1999/XSL/Transform">
   <xsl:output method="xml" version="1.0" encoding="UTF-8" indent="yes"/>
      <xsl:template match="/">
      <ORDERS01>
         <IDOC BEGIN="1">
            <EDI_DC SEGMENT="1">
               <TABNAM>EDI_DC</TABNAM>
               <DOCREL>45B</DOCREL>
               <DIRECT>1</DIRECT>
               <RCVPOR>SAPC11</RCVPOR>
               <RCVPRT>LS</RCVPRT>
               <RCVPRN>T90CLNT090</RCVPRN>
               <SNDPOR>A000000017</SNDPOR>
               <SNDPRT>KU</SNDPRT>
               <MESTYP>ORDERS</MESTYP>
               <IDOCTYP>ORDERS01</IDOCTYP>
               <EXPRSS/>
            </EDI_DC>
            <E1EDK01 SEGMENT="1">
               <ACTION>004</ACTION>
               <CURCY>
                  <xsl:value-of select="/ORDER/ORDER_HEADER/ORDER_INFO/PRICE_CURRENCY"/>
               </CURCY>
               <BELNR>
                  <xsl:value-of select="/ORDER/ORDER_HEADER/ORDER_INFO/ORDER_ID"/>
               </BELNR>
               <BSTZD>
                  <xsl:value-of
select="/ORDER/ORDER_HEADER/ORDER_INFO/ALT_CUSTOMER_ORDER_ID"/>
               </BSTZD>
            </E1EDK01>
            -<E1EDK14 SEGMENT="1">
               <QUALF>006</QUALF>
               <ORGID>00</ORGID>
            </E1EDK14>
            <E1EDK14 SEGMENT="1">
               <QUALF>007</QUALF>
               <ORGID>22</ORGID>
            </E1EDK14>
            <E1EDK14 SEGMENT="1">
               <QUALF>008</QUALF>
               <ORGID>1020</ORGID>
            </E1EDK14>
            <E1EDK14 SEGMENT="1">
               <QUALF>012</QUALF>
               <ORGID>TA</ORGID>
            </E1EDK14>
            <E1EDKA1 SEGMENT="1">
               <PARVW>AG</PARVW>
               <PARTN>
                  <xsl:value-of
select="/ORDER/ORDER_HEADER/ORDER_INFO/ORDER_PARTIES/BUYER_PARTY/PARTY/PARTY_ID[@type='supplie
r_specific']"/>
               </PARTN>
               <NAME1>
                  <xsl:value-of
select="/ORDER/ORDER_HEADER/ORDER_INFO/ORDER_PARTIES/BUYER_PARTY/PARTY/ADDRESS/NAME"/>
               </NAME1>
               <NAME2>
                  <xsl:value-of
select="/ORDER/ORDER_HEADER/ORDER_INFO/ORDER_PARTIES/BUYER_PARTY/PARTY/ADDRESS/NAME2"/>
               </NAME2>
               <NAME3>
                  <xsl:value-of
select="/ORDER/ORDER_HEADER/ORDER_INFO/ORDER_PARTIES/BUYER_PARTY/PARTY/ADDRESS/DEPARTMENT"/>
               </NAME3>
               <STRAS>
                  <xsl:value-of
select="/ORDER/ORDER_HEADER/ORDER_INFO/ORDER_PARTIES/BUYER_PARTY/PARTY/ADDRESS/STREET"/>
               </STRAS>
               <ORT01>
                  <xsl:value-of
select="/ORDER/ORDER_HEADER/ORDER_INFO/ORDER_PARTIES/BUYER_PARTY/PARTY/ADDRESS/CITY"/>
               </ORT01>
               <PSTLZ>
                  <xsl:value-of
select="/ORDER/ORDER_HEADER/ORDER_INFO/ORDER_PARTIES/BUYER_PARTY/PARTY/ADDRESS/ZIP"/>
               </PSTLZ>
               <LAND1>
                  <xsl:value-of
select="/ORDER/ORDER_HEADER/ORDER_INFO/ORDER_PARTIES/BUYER_PARTY/PARTY/ADDRESS/COUNTRY"/>
               </LAND1>
```

```
                <TELF1>
                   <xsl:value-of
select="/ORDER/ORDER_HEADER/ORDER_INFO/ORDER_PARTIES/BUYER_PARTY/PARTY/ADDRESS/PHONE[@type='of
fice']"/>
                </TELF1>
             </E1EDKA1>
             <xsl:if
test="/ORDER/ORDER_HEADER/ORDER_INFO/ORDER_PARTIES/INVOICE_PARTY/PARTY/PARTY_ID[@type='supplie
r_specific'] != ''">
                <E1EDKA1 SEGMENT="1">
                   <PARVW>RE</PARVW>
                   <PARTN>
                      <xsl:value-of
select="/ORDER/ORDER_HEADER/ORDER_INFO/ORDER_PARTIES/INVOICE_PARTY/PARTY/PARTY_ID[@type='suppl
ier_specific']"/>
                   </PARTN>
                   <NAME1>
                      <xsl:value-of
select="/ORDER/ORDER_HEADER/ORDER_INFO/ORDER_PARTIES/INVOICE_PARTY/PARTY/ADDRESS/NAME"/>
                   </NAME1>
                   <NAME2>
                      <xsl:value-of
select="/ORDER/ORDER_HEADER/ORDER_INFO/ORDER_PARTIES/INVOICE_PARTY/PARTY/ADDRESS/NAME2"/>
                   </NAME2>
                   <NAME3>
                      <xsl:value-of
select="/ORDER/ORDER_HEADER/ORDER_INFO/ORDER_PARTIES/INVOICE_PARTY/PARTY/ADDRESS/DEPARTMENT"/>
                   </NAME3>
                   <STRAS>
                      <xsl:value-of
select="/ORDER/ORDER_HEADER/ORDER_INFO/ORDER_PARTIES/INVOICE_PARTY/PARTY/ADDRESS/STREET"/>
                   </STRAS>
                   <ORT01>
                      <xsl:value-of
select="/ORDER/ORDER_HEADER/ORDER_INFO/ORDER_PARTIES/INVOICE_PARTY/PARTY/ADDRESS/CITY"/>
                   </ORT01>
                   <PSTLZ>
                      <xsl:value-of
select="/ORDER/ORDER_HEADER/ORDER_INFO/ORDER_PARTIES/INVOICE_PARTY/PARTY/ADDRESS/ZIP"/>
                   </PSTLZ>
                   <LAND1>
                      <xsl:value-of
select="/ORDER/ORDER_HEADER/ORDER_INFO/ORDER_PARTIES/INVOICE_PARTY/PARTY/ADDRESS/COUNTRY"/>
                   </LAND1>
                   <TELF1>
                      <xsl:value-of
select="/ORDER/ORDER_HEADER/ORDER_INFO/ORDER_PARTIES/INVOICE_PARTY/PARTY/ADDRESS/PHONE[@type='
office']"/>
                   </TELF1>
                </E1EDKA1>
             </xsl:if>
             <E1EDK02 SEGMENT="1">
                <QUALF>001</QUALF>
                <BELNR>
                   <xsl:value-of select="/ORDER/ORDER_HEADER/ORDER_INFO/ORDER_ID"/>
                </BELNR>
             </E1EDK02>
             <xsl:for-each select="/ORDER/ORDER_ITEM_LIST/ORDER_ITEM">
                <E1EDP01 SEGMENT="1">
                   <POSEX>
                      <xsl:value-of select="LINE_ITEM_ID"/>
                   </POSEX>
                   <ACTION>001</ACTION>
                   <PSTYP>0</PSTYP>
                   <MENGE>
                      <xsl:value-of select="QUANTITY"/>
                   </MENGE>
                   <MENEE>
                      <xsl:value-of select="ORDER_UNIT"/>
                   </MENEE>
                   <PMENE>
                      <xsl:value-of select="ORDER_UNIT"/>
                   </PMENE>
                   <VPREI>
                      <xsl:value-of select="ARTICLE_PRICE[@type='net_list']/PRICE_AMOUNT"/>
                   </VPREI>
                   <PEINH>
                      <xsl:value-of select="PRICE_QUANTITY"/>
                   </PEINH>
                   <NETWR>
                      <xsl:value-of select="PRICE_LINE_AMOUNT"/>
                   </NETWR>
                   <CURCY>
                      <xsl:value-of select="/ORDER/ORDER_HEADER/ORDER_INFO/PRICE_CURRENCY"/>
                   </CURCY>
                   <E1EDP02 SEGMENT="1">
                      <QUALF>001</QUALF>
                      <BELNR>
                         <xsl:value-of select="/ORDER/ORDER_HEADER/ORDER_INFO/ORDER_ID"/>
```

```
                    </BELNR>
                 </E1EDP02>
                 <E1EDP03 SEGMENT="1">
                    <IDDAT>025</IDDAT>
                    <DATUM>19990905</DATUM>
                    <UZEIT>164650</UZEIT>
                 </E1EDP03>
                 <E1EDP03 SEGMENT="1">
                    <IDDAT>023</IDDAT>
                    <DATUM>
                       <xsl:value-of
select="substring(/ORDER/ORDER_HEADER/ORDER_INFO/ORDER_DATE,1,4)"/>
                       <xsl:value-of
select="substring(/ORDER/ORDER_HEADER/ORDER_INFO/ORDER_DATE,6,2)"/>
                       <xsl:value-of
select="substring(/ORDER/ORDER_HEADER/ORDER_INFO/ORDER_DATE,9,2)"/>
                    </DATUM>
                 </E1EDP03>
                 <E1EDP03 SEGMENT="1">
                    <IDDAT>026</IDDAT>
                    <DATUM>
                       <xsl:value-of
select="substring(/ORDER/ORDER_HEADER/ORDER_INFO/ORDER_DATE,1,4)"/>
                       <xsl:value-of
select="substring(/ORDER/ORDER_HEADER/ORDER_INFO/ORDER_DATE,6,2)"/>
                       <xsl:value-of
select="substring(/ORDER/ORDER_HEADER/ORDER_INFO/ORDER_DATE,9,2)"/>
                    </DATUM>
                 </E1EDP03>
                 <xsl:if test="ARTICLE_ID/BUYER_AID">
                    <E1EDP19 SEGMENT="1">
                       <QUALF>001</QUALF>
                       <IDTNR>
                          <xsl:value-of select="ARTICLE_ID/BUYER_AID"/>
                    </IDTNR>
                       <KTEXT>
                          <xsl:value-of select="ARTICLE_ID/DESCRIPTION_SHORT"/>
                       </KTEXT>
                    </E1EDP19>
                 </xsl:if>
                 <xsl:if test="ARTICLE_ID/SUPPLIER_AID !='' ">
                    <E1EDP19 SEGMENT="1">
                       <QUALF>002</QUALF>
                       <IDTNR>
                          <xsl:value-of select="ARTICLE_ID/SUPPLIER_AID"/>
                       </IDTNR>
                       <KTEXT>
                          <xsl:value-of select="ARTICLE_ID/DESCRIPTION_SHORT"/>
                       </KTEXT>
                    </E1EDP19>
                 </xsl:if>
                 <xsl:if test="ARTICLE_ID/INTERNATIONAL_AID/@type = 'EAN'">
                    <E1EDP19 SEGMENT="1">
                       <QUALF>003</QUALF>
                       <IDTNR>
                          <xsl:value-of select="ARTICLE_ID/INTERNATIONAL_AID"/>
                       </IDTNR>
                       <KTEXT>
                          <xsl:value-of select="ARTICLE_ID/DESCRIPTION_SHORT"/>
                       </KTEXT>
                    </E1EDP19>
                 </xsl:if>
              </E1EDP01>
           </xsl:for-each>
           <E1EDS01 SEGMENT="1">
              <SUMID>001</SUMID>
              <SUMME>
                 <xsl:value-of select="count(/ORDER/ORDER_ITEM_LIST/ORDER_ITEM)"/>
              </SUMME>
           </E1EDS01>
           <E1EDS01 SEGMENT="1">
              <SUMID>002</SUMID>
              <SUMME>
                 <xsl:value-of
select="sum(/ORDER/ORDER_ITEM_LIST/ORDER_ITEM/ARTICLE_PRICE[@type='net_list']/PRICE_LINE_AMOUN
T)"/>
              </SUMME>
              <SUNIT>
                 <xsl:value-of select="/ORDER/ORDER_HEADER/ORDER_INFO/PRICE_CURRENCY"/>
              </SUNIT>
           </E1EDS01>
        </IDOC>
     </ORDERS01>
  </xsl:template>
</xsl:stylesheet>
```

14 Anhang B2 IDoc-openTRANS XSLT-Schablonen

```
<?xml version="1.0" encoding="UTF-8"?>
<xsl:stylesheet version="1.0" xmlns:xsl="http://www.w3.org/1999/XSL/Transform">
   <xsl:output    method="xml"    version="1.0"    encoding="UTF-8"    indent="yes"    doctype-
system="openTRANS_ORDERRESPONSE_1_0.dtd"/>
   <xsl:template match="/">
      <ORDERRESPONSE version="1.0">
         <ORDERRESPONSE_HEADER>
            <CONTROL_INFO>
               <GENERATOR_INFO>
                     Created by openTRANS-Generator V1.7
                     SAP-IDoc DOCNUM:<xsl:value-of select="/ORDERS01/IDOC/EDI_DC/DOCNUM"/>
               </GENERATOR_INFO>
               <GENERATION_DATE>
                  <xsl:for-each select="/ORDERS01/IDOC/E1EDK03">
                     <xsl:if test="IDDAT = '012'">
                        <xsl:value-of            select="substring(DATUM,1,4)"/>-<xsl:value-of
select="substring(DATUM,5,2)"/>-<xsl:value-of select="substring(DATUM,7,2)"/>
                     </xsl:if>
                  </xsl:for-each>
               </GENERATION_DATE>
            </CONTROL_INFO>
            <ORDERRESPONSE_INFO>
               <ORDER_ID>
                  <xsl:for-each select="/ORDERS01/IDOC/E1EDK02">
                     <xsl:if test="QUALF = '001'">
                        <xsl:value-of select="BELNR"/>
                     </xsl:if>
                  </xsl:for-each>
               </ORDER_ID>
               <SUPPLIER_ORDER_ID>
                  <xsl:for-each select="/ORDERS01/IDOC/E1EDK02">
                     <xsl:if test="QUALF = '002'">
                        <xsl:value-of select="BELNR"/>
                     </xsl:if>
                  </xsl:for-each>
               </SUPPLIER_ORDER_ID>
               <ORDER_DATE>
                  <xsl:for-each select="/ORDERS01/IDOC/E1EDK03">
                     <xsl:if test="IDDAT = '025'">
                        <xsl:value-of            select="substring(DATUM,1,4)"/>-<xsl:value-of
select="substring(DATUM,5,2)"/>-<xsl:value-of select="substring(DATUM,7,2)"/>
                     </xsl:if>
                  </xsl:for-each>
               </ORDER_DATE>
               <ORDERRESPONSE_DATE>
                  <xsl:for-each select="/ORDERS01/IDOC/E1EDK03">
                     <xsl:if test="IDDAT = '012'">
                        <xsl:value-of            select="substring(DATUM,1,4)"/>-<xsl:value-of
select="substring(DATUM,5,2)"/>-<xsl:value-of select="substring(DATUM,7,2)"/>
                     </xsl:if>
                  </xsl:for-each>
               </ORDERRESPONSE_DATE>
               <ORDER_PARTIES>
                  <xsl:for-each select="/ORDERS01/IDOC/E1EDKA1">
                     <xsl:if test="PARVW = 'AG'">
                        <BUYER_PARTY>
                           <PARTY>
                              <PARTY_ID type="supplier_specific">
                                 <xsl:value-of select="PARTN"/>
                              </PARTY_ID>
                              <ADDRESS>
                                 <NAME>
                                    <xsl:value-of select="NAME1"/>
                                 </NAME>
                                 <NAME2>
                                    <xsl:value-of select="NAME2"/>
                                 </NAME2>
                                 <DEPARTMENT>
                                    <xsl:value-of select="NAME3"/>
                                 </DEPARTMENT>
                                 <STREET>
                                    <xsl:value-of select="STRAS"/>
                                 </STREET>
                                 <ZIP>
                                    <xsl:value-of select="PSTLZ"/>
                                 </ZIP>
                                 <CITY>
                                    <xsl:value-of select="ORT"/>
                                 </CITY>
                                 <COUNTRY>
                                    <xsl:value-of select="LAND1"/>
                                 </COUNTRY>
                                 <PHONE type="office">
```

```
                        <xsl:value-of select="TELF1"/>
                     </PHONE>
                  </ADDRESS>
               </PARTY>
            </BUYER_PARTY>
         </xsl:if>
      </xsl:for-each>
      <SUPPLIER_PARTY>
         <PARTY>
            <PARTY_ID type="buyer_specific">123456789</PARTY_ID>
            <ADDRESS>
               <NAME>IDES Holding Inc.</NAME>
            </ADDRESS>
         </PARTY>
      </SUPPLIER_PARTY>
      <xsl:for-each select="/ORDERS01/IDOC/E1EDKA1">
         <xsl:if test="PARVW = 'RE'">
            <INVOICE_PARTY>
               <PARTY>
                  <PARTY_ID type="supplier_specific">
                     <xsl:value-of select="PARTN"/>
                  </PARTY_ID>
                  <ADDRESS>
                     <NAME>
                        <xsl:value-of select="NAME1"/>
                     </NAME>
                     <NAME2>
                        <xsl:value-of select="NAME2"/>
                     </NAME2>
                     <DEPARTMENT>
                        <xsl:value-of select="NAME3"/>
                     </DEPARTMENT>
                     <STREET>
                        <xsl:value-of select="STRAS"/>
                     </STREET>
                     <ZIP>
                        <xsl:value-of select="PSTLZ"/>
                     </ZIP>
                     <CITY>
                        <xsl:value-of select="ORT"/>
                     </CITY>
                     <COUNTRY>
                        <xsl:value-of select="LAND1"/>
                     </COUNTRY>
                     <PHONE type="office">
                        <xsl:value-of select="TELF1"/>
                     </PHONE>
                  </ADDRESS>
               </PARTY>
            </INVOICE_PARTY>
         </xsl:if>
      </xsl:for-each>
   </ORDER_PARTIES>
   <PRICE_CURRENCY>
      <xsl:value-of select="/ORDERS01/IDOC/E1EDK01/CURCY"/>
   </PRICE_CURRENCY>
</ORDERRESPONSE_INFO>
</ORDERRESPONSE_HEADER>
<ORDERRESPONSE_ITEM_LIST>
   <xsl:for-each select="/ORDERS01/IDOC/E1EDP01">
      <ORDERRESPONSE_ITEM>
         <LINE_ITEM_ID>
            <xsl:value-of select="POSEX"/>
         </LINE_ITEM_ID>
         <ARTICLE_ID>
            <xsl:for-each select="E1EDP19">
               <xsl:if test="QUALF = '002'">
                  <SUPPLIER_AID>
                     <xsl:value-of select="IDTNR"/>
                  </SUPPLIER_AID>
               </xsl:if>
               <xsl:if test="QUALF = '003'">
                  <INTERNATIONAL_AID type="EAN">
                     <xsl:value-of select="IDTNR"/>
                  </INTERNATIONAL_AID>
               </xsl:if>
               <xsl:if test="QUALF = '001'">
                  <BUYER_AID type="buyer_specific">
                     <xsl:value-of select="IDTNR"/>
                  </BUYER_AID>
               </xsl:if>
            </xsl:for-each>
            <xsl:for-each select="E1EDP19">
               <xsl:if test="QUALF = '002'">
                  <DESCRIPTION_SHORT>
                     <xsl:value-of select="KTEXT"/>
                  </DESCRIPTION_SHORT>
               </xsl:if>
            </xsl:for-each>
```

```
                  </ARTICLE_ID>
                  <QUANTITY>
                     <xsl:value-of select="format-number(MENGE,'#.######')"/>
                  </QUANTITY>
                  <ORDER_UNIT>
                     <xsl:value-of select="MENEE"/>
                  </ORDER_UNIT>
                  <ARTICLE_PRICE type="net_list">
                     <PRICE_AMOUNT>
                        <xsl:value-of select="format-number(VPREI,'#.######')"/>
                     </PRICE_AMOUNT>
                     <PRICE_LINE_AMOUNT>
                        <xsl:value-of select="format-number(NETWR,'#.######')"/>
                     </PRICE_LINE_AMOUNT>
                  </ARTICLE_PRICE>
               </ORDERRESPONSE_ITEM>
            </xsl:for-each>
         </ORDERRESPONSE_ITEM_LIST>
         <ORDERRESPONSE_SUMMARY>
            <xsl:for-each select="/ORDERS01/IDOC/E1EDS01">
               <xsl:if test="SUMID = '001'">
                  <TOTAL_ITEM_NUM>
                     <xsl:value-of select="format-number(SUMME,'#.######')"/>
                  </TOTAL_ITEM_NUM>
               </xsl:if>
               <xsl:if test="SUMID = '002'">
                  <TOTAL_AMOUNT>
                     <xsl:value-of select="format-number(SUMME,'#.######')"/>
                  </TOTAL_AMOUNT>
               </xsl:if>
            </xsl:for-each>
         </ORDERRESPONSE_SUMMARY>
      </ORDERRESPONSE>
   </xsl:template>
</xsl:stylesheet>
```

15 Anhang C1 openTRANS-Zuordungen

openTRANS (Quellformat)		**BAPI SalesOrder.Simulate**	
Kontext	Feld	Kontext	Feld / Wert
ORDER_HEADER	ORDER_ID	ORDER_HEADER_IN	PURCH_NO
ORDER_HEADER	PRICE_CURRENCY	ORDER_HEADER_IN	CURRENCY
BUYER_PARTY	-	ORDER_PARTNERS	“AG”
BUYER_PARTY	PARTY_ID	ORDER_PARTNERS	PARTN_NUMB
BUYER_PARTY	NAME	ORDER_PARTNERS	NAME
BUYER_PARTY	NAME2	ORDER_PARTNERS	NAME_2
BUYER_PARTY	STREET	ORDER_PARTNERS	STREET
BUYER_PARTY	DEPARTMENT	ORDER_PARTNERS	NAME_3
BUYER_PARTY	ZIP	ORDER_PARTNERS	POSTL_CODE
BUYER_PARTY	CITY	ORDER_PARTNERS	CITY
BUYER_PARTY	COUNTRY	ORDER_PARTNERS	COUNTRY
SUPPLIER_PARTY	-	ORDER_PARTNERS	“AN”
SUPPLIER_PARTY	PARTY_ID	ORDER_PARTNERS	PARTN_NUMB
SUPPLIER _PARTY	NAME	ORDER_PARTNERS	NAME
SUPPLIER _PARTY	NAME2	ORDER_PARTNERS	NAME_2
SUPPLIER _PARTY	STREET	ORDER_PARTNERS	STREET
SUPPLIER _PARTY	DEPARTMENT	ORDER_PARTNERS	NAME_3
SUPPLIER _PARTY	ZIP	ORDER_PARTNERS	POSTL_CODE
SUPPLIER _PARTY	CITY	ORDER_PARTNERS	CITY
SUPPLIER _PARTY	COUNTRY	ORDER_PARTNERS	COUNTRY
ORDER_ITEM	SUPPLIER_AID	ORDER_ITEMS_IN	MATERIAL
ORDER_ITEM	DESCRIPTION_SHORT	ORDER_ITEMS_IN	SHORT_TEXT
ORDER_ITEM	QUANTITY	ORDER_ITEMS_IN	REQ_QTY
ORDER_ITEM	ORDER_UNIT	ORDER_ITEMS_IN	UNIT

16 Anhang D1 IDoc Nachricht ORDER02 (ASCII)

```
EDI_DC 035000000000001301230C 30ZSDEL3111SAPI10TESTLICK92046 X3050 850 2 SAPI10 KUCK92046
E2EDK09001035000000000001301200000 1E2EDK09001000000010 5500000002 199605310017 16 19960531
E2EDKA1 035000000000001301200000 2E2EDKA1 000001020 AG EDI DEFAULT TEST ABCD Diruscia P001 US
Purchasing                                    Org                                    EDI
E2EDKA1    035000000000001301200000 3E2EDKA1    000001020    LF    0020000023    717-986-7512
E2EDP10  035000000000001301200000 4E2EDP10  000001020  A3228-80002  PCE5200  30.000  19960529
0000000000000000 178.000 48.000 19960605 19960630 000000000000000000300000000 00000000 00000000
00000000
Z2EDP91 035000000000001301200000 5Z2EDP91 000004030 92288888888 000100.14 3 Tax exempt code
Z2EDP19  035000000000001301200000 6Z2EDP19  000004030 90512345678 Engineering Change Number
Z2EDP19    035000000000001301200000 7Z2EDP19    000004030    90745689278    Drawing    Number
Z2EDP19  035000000000001301200000 8Z2EDP19  000004030  90478897988  Drawing  Revision  Number
Z2EDP19    035000000000001301200000 9Z2EDP19    000004030    90699998988    Packaging    Number
Z2EDP19  035000000000001301200000 10Z2EDP19  000004030  90412345678  Drawing  revision  number
Z2EDP19 035000000000001301200000 11Z2EDP19 000004030 90888779966 General Specification Number
Z2EDPA1  035000000000001301200000 12Z2EDPA1  000004030  WE  IN  THE  SKY  BUILDING  THE_CODE
Z2EDPA1     035000000000001301200000 13Z2EDPA1     000004030     OC     SMDO     West     5200
E2EDP16   035000000000001301200000 14E2EDP16   000004030   19960531000019960531   60   108.000
Z2EDP16              035000000000001301200000 15Z2EDP16              000014040              1
E2EDP16   035000000000001301200000 16E2EDP16   000004030   19960628000019960628   50   158.000
Z2EDP16              035000000000001301200000 17Z2EDP16              000016040              2
E2EDP16   035000000000001301200000 18E2EDP16   000004030   19960712000019960712   20   178.000
Z2EDP16 035000000000001301200000 19Z2EDP16 000018040
```

17 Anhang D2 IDoc Nachricht ORDERS (XML)


```xml
<?xml version="1.0" encoding="iso-8859-1"?>
<ORDERS01>
   <IDOC BEGIN="1">
      <EDI_DC SEGMENT="1">
         <TABNAM>EDI_DC</TABNAM>
         <MANDT>000</MANDT>
         <DOCNUM>0000471108150000</DOCNUM>
         <DOCREL>45B</DOCREL>
         <DIRECT>1</DIRECT>
         <RCVPOR>SAPC11</RCVPOR>
         <RCVPRT>LS</RCVPRT>
         <RCVPRN>T90CLNT090</RCVPRN>
         <SNDPOR>A000000017</SNDPOR>
         <SNDPRT>KU</SNDPRT>
         <SNDPRN>0000007777</SNDPRN>
         <MESTYP>ORDERS</MESTYP>
         <IDOCTYP>ORDERS01</IDOCTYP>
         <SNDPFC>AG</SNDPFC>
      </EDI_DC>
      <E1EDK01 SEGMENT="1">
         <ACTION>004</ACTION>
         <CURCY>DEM</CURCY>
         <BELNR>PO4500002323</BELNR>
         <BSTZD>1000</BSTZD>
      </E1EDK01>
      <E1EDK14 SEGMENT="1">
         <QUALF>006</QUALF>
         <ORGID>00</ORGID>
      </E1EDK14>
      <E1EDK14 SEGMENT="1">
         <QUALF>007</QUALF>
         <ORGID>22</ORGID>
      </E1EDK14>
      <E1EDK14 SEGMENT="1">
         <QUALF>008</QUALF>
         <ORGID>1020</ORGID>
      </E1EDK14>
      <E1EDK14 SEGMENT="1">
         <QUALF>012</QUALF>
         <ORGID>TA</ORGID>
      </E1EDK14>
      <E1EDKA1 SEGMENT="1">
         <PARVW>AG</PARVW>
         <PARTN>0000007777</PARTN>
         <NAME1>Fraunhofer IAO</NAME1>
         <NAME2>Gebaeude TIII</NAME2>
         <NAME3>Marktstrategie Team E-Procurement</NAME3>
         <STRAS>Nobelstr. 12</STRAS>
         <ORT01>Stuttgart</ORT01>
         <PSTLZ>70569</PSTLZ>
         <LAND1>DE</LAND1>
         <TELF1>+49 711 970 0</TELF1>
      </E1EDKA1>
      <E1EDK02 SEGMENT="1">
         <QUALF>001</QUALF>
         <BELNR>PO4500002323</BELNR>
      </E1EDK02>
      <E1EDP01 SEGMENT="1">
         <POSEX>P10001</POSEX>
         <ACTION>001</ACTION>
         <PSTYP>0</PSTYP>
         <MENGE>20</MENGE>
         <MENEE>KGM</MENEE>
         <PMENE>KGM</PMENE>
         <VPREI>3.50</VPREI>
         <CURCY>DEM</CURCY>
         <E1EDP02 SEGMENT="1">
            <QUALF>001</QUALF>
            <BELNR>PO4500002323</BELNR>
         </E1EDP02>
         <E1EDP03 SEGMENT="1">
            <IDDAT>025</IDDAT>
            <DATUM>19990905</DATUM>
            <UZEIT>164650</UZEIT>
         </E1EDP03>
         <E1EDP19 SEGMENT="1">
            <QUALF>001</QUALF>
            <IDTNR>4711</IDTNR>
            <KTEXT>blaue Farbe</KTEXT>
         </E1EDP19>
         <E1EDP19 SEGMENT="1">
            <QUALF>002</QUALF>
            <IDTNR>Y-500</IDTNR>
            <KTEXT>blaue Farbe</KTEXT>
         </E1EDP19>
```

```
         <E1EDP19 SEGMENT="1">
            <QUALF>003</QUALF>
            <IDTNR>8712670911213</IDTNR>
            <KTEXT>blaue Farbe</KTEXT>
         </E1EDP19>
      </E1EDP01>
      <E1EDP01 SEGMENT="1">
         <POSEX>P10002</POSEX>
         <ACTION>001</ACTION>
         <PSTYP>0</PSTYP>
         <MENGE>20</MENGE>
         <MENEE>KGM</MENEE>
         <PMENE>KGM</PMENE>
         <VPREI>3.50</VPREI>
         <CURCY>DEM</CURCY>
         <E1EDP02 SEGMENT="1">
            <QUALF>001</QUALF>
            <BELNR>PO4500002323</BELNR>
         </E1EDP02>
         <E1EDP03 SEGMENT="1">
            <IDDAT>025</IDDAT>
            <DATUM>19990905</DATUM>
            <UZEIT>164650</UZEIT>
         </E1EDP03>
         <E1EDP19 SEGMENT="1">
            <QUALF>001</QUALF>
            <IDTNR>10815</IDTNR>
            <KTEXT>blaue Farbe</KTEXT>
         </E1EDP19>
         <E1EDP19 SEGMENT="1">
            <QUALF>002</QUALF>
            <IDTNR>Y-500</IDTNR>
            <KTEXT>blaue Farbe</KTEXT>
         </E1EDP19>
         <E1EDP19 SEGMENT="1">
            <QUALF>003</QUALF>
            <IDTNR>8712670911581</IDTNR>
            <KTEXT>blaue Farbe</KTEXT>
         </E1EDP19>
      </E1EDP01>
      <E1EDS01 SEGMENT="1">
         <SUMID>001</SUMID>
         <SUMME>2</SUMME>
      </E1EDS01>
      <E1EDS01 SEGMENT="1">
         <SUMID>002</SUMID>
         <SUMME>180</SUMME>
         <SUNIT>DEM</SUNIT>
      </E1EDS01>
   </IDOC>
</ORDERS01>
```

18 Anhang E1 openTRANS-Beispielnachrichten

18.1 ORDER (Bestellung)


```
<?xml version="1.0" encoding="UTF-8"?>
<ORDER version="1.0" type="standard">
   <ORDER_HEADER>
      <CONTROL_INFO>
         <STOP_AUTOMATIC_PROCESSING>testimport</STOP_AUTOMATIC_PROCESSING>
         <GENERATOR_INFO>Created by openTRANS-Generator V1.7  1.03.2001</GENERATOR_INFO>
         <GENERATION_DATE>2001-03-21T18:00:30+01:00</GENERATION_DATE>
         <MIME_ROOT>/mime-dateien/</MIME_ROOT>
      </CONTROL_INFO>
      <ORDER_INFO>
         <ORDER_ID>PO4500002323</ORDER_ID>
         <ALT_CUSTOMER_ORDER_ID>1000000122</ALT_CUSTOMER_ORDER_ID>
         <ORDER_DATE>2001-03-21T16:55:30+01:00</ORDER_DATE>
         <ORDER_PARTIES>
            <BUYER_PARTY>
               <PARTY>
                  <PARTY_ID type="buyer_specific">FhG-IAO-4356</PARTY_ID>
                  <ADDRESS>
                     <NAME>Fraunhofer IAO</NAME>
                     <NAME2>Gebaeude TIII</NAME2>
                     <DEPARTMENT>Marktstrategie Team E-Procurement</DEPARTMENT>
                     <STREET>Nobelstr. 12</STREET>
                     <ZIP>70569</ZIP>
                     <CITY>Stuttgart</CITY>
                     <COUNTRY>DE</COUNTRY>
                     <VAT_ID>DE 129515865</VAT_ID>
                     <PHONE type="office">+49 711 970 0</PHONE>
                     <URL>http://www.iao.fhg.de</URL>
                  </ADDRESS>
               </PARTY>
            </BUYER_PARTY>
            <SUPPLIER_PARTY>
               <PARTY>
                  <PARTY_ID type="buyer_specific">UEG-FB5-BLI</PARTY_ID>
                  <ADDRESS>
                     <NAME>Universitaet Essen</NAME>
                     <NAME2>FB Wirtschaftswissenschaften/Wirtschaftsinformatik</NAME2>
                     <DEPARTMENT>Beschaffung, Logistik und Informationsmanagement</DEPARTMENT>
                     <STREET>Universitaetsstr. 9</STREET>
                     <ZIP>45117</ZIP>
                     <CITY>Essen</CITY>
                     <COUNTRY>DE</COUNTRY>
                     <PHONE type="office">+49 201 183 0</PHONE>
                     <URL>http://www.bli.uni-essen.de</URL>
                  </ADDRESS>
               </PARTY>
            </SUPPLIER_PARTY>
         </ORDER_PARTIES>
         <PRICE_CURRENCY>EUR</PRICE_CURRENCY>

   <TERMS_AND_CONDITIONS>http://www.iao.fhg.de/procurement/agb.pdf</TERMS_AND_CONDITIONS>
         <TRANSPORT>
            <INCOTERM>CFR</INCOTERM>
            <LOCATION>Zentale Postannahmestelle</LOCATION>
            <TRANSPORT_REMARK>Bitte bei Fr. Mueller, 2. Tuere, abgeben</TRANSPORT_REMARK>
         </TRANSPORT>
      </ORDER_INFO>
   </ORDER_HEADER>
   <ORDER_ITEM_LIST>
      <ORDER_ITEM>
         <LINE_ITEM_ID>P100012</LINE_ITEM_ID>
         <ARTICLE_ID>
            <SUPPLIER_AID>55-K-31</SUPPLIER_AID>
            <INTERNATIONAL_AID type="EAN">8712670911213</INTERNATIONAL_AID>
            <BUYER_AID type="BRZNR">4711</BUYER_AID>
            <DESCRIPTION_SHORT>Standard Briefkorb DIN A4</DESCRIPTION_SHORT>
            <DESCRIPTION_LONG>Ein Klassiker unter den Briefkoerben.</DESCRIPTION_LONG>
            <MANUFACTURER_INFO>
               <MANUFACTURER_NAME>Plastikpartner</MANUFACTURER_NAME>
               <MANUFACTURER_AID>123-RD-67-U</MANUFACTURER_AID>
               <MANUFACTURER_TYPE_DESCR>Briefkorb 02133</MANUFACTURER_TYPE_DESCR>
            </MANUFACTURER_INFO>
         </ARTICLE_ID>
         <QUANTITY>20</QUANTITY>
         <ORDER_UNIT>C62</ORDER_UNIT>
         <ARTICLE_PRICE type="net_list">
            <PRICE_AMOUNT>2.90</PRICE_AMOUNT>
            <PRICE_FLAG type="incl_freight">TRUE</PRICE_FLAG>
```

```
            <PRICE_LINE_AMOUNT>58.00</PRICE_LINE_AMOUNT>
            <TAX>0.16</TAX>
            <TAX_AMOUNT>9.28</TAX_AMOUNT>
            <PRICE_QUANTITY>1</PRICE_QUANTITY>
         </ARTICLE_PRICE>
         <ACCOUNTING_INFO>
            <COST_CATEGORY_ID type="cost_center">11047115</COST_CATEGORY_ID>
            <COST_TYPE>Verbrauch</COST_TYPE>
            <COST_ACCOUNT>S0202556600</COST_ACCOUNT>
         </ACCOUNTING_INFO>
         <PARTIAL_SHIPMENT_ALLOWED>false</PARTIAL_SHIPMENT_ALLOWED>
      </ORDER_ITEM>
      <ORDER_ITEM>
         <LINE_ITEM_ID>P100028</LINE_ITEM_ID>
         <ARTICLE_ID>
            <SUPPLIER_AID>55-K-56</SUPPLIER_AID>
            <INTERNATIONAL_AID type="EAN">8712670911581</INTERNATIONAL_AID>
            <BUYER_AID type="BRZNR">10815</BUYER_AID>
            <DESCRIPTION_SHORT>Exklusiv Briefoeffner</DESCRIPTION_SHORT>
            <DESCRIPTION_LONG>
               Besonders formschoener Briefoeffner aus hochglanzpoliertem Edelstahl
            </DESCRIPTION_LONG>
            <MANUFACTURER_INFO>
               <MANUFACTURER_NAME>Metallpartner</MANUFACTURER_NAME>
               <MANUFACTURER_AID>EBO-368-13</MANUFACTURER_AID>
               <MANUFACTURER_TYPE_DESCR>Briefoeffner Edelmann</MANUFACTURER_TYPE_DESCR>
            </MANUFACTURER_INFO>
         </ARTICLE_ID>
         <QUANTITY>4</QUANTITY>
         <ORDER_UNIT>C62</ORDER_UNIT>
         <ARTICLE_PRICE type="net_list">
            <PRICE_AMOUNT>30.5</PRICE_AMOUNT>
            <PRICE_LINE_AMOUNT>122</PRICE_LINE_AMOUNT>
            <TAX>0.16</TAX>
            <TAX_AMOUNT>19.52</TAX_AMOUNT>
            <PRICE_QUANTITY>1</PRICE_QUANTITY>
         </ARTICLE_PRICE>
      </ORDER_ITEM>
   </ORDER_ITEM_LIST>
   <ORDER_SUMMARY>
      <TOTAL_ITEM_NUM>2</TOTAL_ITEM_NUM>
      <TOTAL_AMOUNT>180</TOTAL_AMOUNT>
   </ORDER_SUMMARY>
</ORDER>
```

18.2 ORDERRESPONSE (Bestellbestätigung)


```
<?xml version="1.0" encoding="UTF-8"?>
<ORDERRESPONSE version="1.0">
   <ORDERRESPONSE_HEADER>
      <CONTROL_INFO>
         <STOP_AUTOMATIC_PROCESSING>testimport </STOP_AUTOMATIC_PROCESSING>
         <GENERATOR_INFO>Created by openTRANS-Generator V1.7 1.09.2001</GENERATOR_INFO>
         <GENERATION_DATE>2001-09-06-10T09:00:30+02:00</GENERATION_DATE>
         <MIME_ROOT>/mime-dateien/</MIME_ROOT>
      </CONTROL_INFO>
      <ORDERRESPONSE_INFO>
         <ORDER_ID>PO4500002323</ORDER_ID>
         <ALT_CUSTOMER_ORDER_ID>1000000122</ALT_CUSTOMER_ORDER_ID>
         <SUPPLIER_ORDER_ID>AUF200110010</SUPPLIER_ORDER_ID>
         <ORDER_DATE>2001-09-06T16:55:30+01:00</ORDER_DATE>
         <ORDERRESPONSE_DATE>2001-09-10T09:00:30+02:00</ORDERRESPONSE_DATE>
         <ORDER_PARTIES>
            <BUYER_PARTY>
               <PARTY>
                  <PARTY_ID type="buyer_specific">FhG-IAO-4356</PARTY_ID>
                  <ADDRESS>
                     <NAME>Fraunhofer IAO</NAME>
                     <NAME2>Gebaeude TIII</NAME2>
                     <DEPARTMENT>Marktstrategie Team E-Procurement</DEPARTMENT>
                     <STREET>Nobelstr. 12</STREET>
                     <ZIP>70569</ZIP>
                     <CITY>Stuttgart</CITY>
                     <COUNTRY>Germany</COUNTRY>
                     <VAT_ID>DE 129515865</VAT_ID>
                     <PHONE type="office">+49 711 970 0</PHONE>
                     <URL>http://www.iao.fhg.de</URL>
                  </ADDRESS>
               </PARTY>
            </BUYER_PARTY>
            <SUPPLIER_PARTY>
               <PARTY>
                  <PARTY_ID type="buyer_specific">UEG-FB5-BLI</PARTY_ID>
                  <ADDRESS>
                     <NAME>Universitaet Essen</NAME>
                     <NAME2>FB Wirtschaftswissenschaften/Wirtschaftsinformatik</NAME2>
                     <DEPARTMENT>Beschaffung, Logistik und Informationsmanagement</DEPARTMENT>
                     <STREET>Universitaetsstr. 9</STREET>
                     <ZIP>45117</ZIP>
                     <CITY>Essen</CITY>
                     <COUNTRY>Germany</COUNTRY>
                     <PHONE type="office">+49 201 183 0</PHONE>
                     <URL>http://www.bli.uni-essen.de</URL>
                  </ADDRESS>
               </PARTY>
            </SUPPLIER_PARTY>
            <INVOICE_PARTY>
               <PARTY>
                  <PARTY_ID type="buyer_specific">FhG-IAO-4356</PARTY_ID>
                  <ADDRESS>
                     <NAME>Fraunhofer IAO</NAME>
                     <NAME2>Gebaeude TI</NAME2>
                     <DEPARTMENT>Verwaltung IAO</DEPARTMENT>
                     <STREET>Nobelstr. 12</STREET>
                     <ZIP>70569</ZIP>
                     <CITY>Stuttgart</CITY>
                     <COUNTRY>Germany</COUNTRY>
                     <PHONE type="office">+49 711 970 01</PHONE>
                  </ADDRESS>
               </PARTY>
            </INVOICE_PARTY>
         </ORDER_PARTIES>
         <PRICE_CURRENCY>EUR</PRICE_CURRENCY>
      </ORDERRESPONSE_INFO>
   </ORDERRESPONSE_HEADER>
   <ORDERRESPONSE_ITEM_LIST>
      <ORDERRESPONSE_ITEM>
         <LINE_ITEM_ID>P100012</LINE_ITEM_ID>
         <ARTICLE_ID>
            <SUPPLIER_AID>55-K-31</SUPPLIER_AID>
            <INTERNATIONAL_AID type="EAN">8712670911213</INTERNATIONAL_AID>
            <BUYER_AID type="BRZNR">4711</BUYER_AID>
            <DESCRIPTION_SHORT>Standard Briefkorb DIN A4</DESCRIPTION_SHORT>
            <DESCRIPTION_LONG>Ein Klassiker unter den Briefkoerben.</DESCRIPTION_LONG>
         </ARTICLE_ID>
         <QUANTITY>20</QUANTITY>
         <ORDER_UNIT>C62</ORDER_UNIT>
         <ARTICLE_PRICE type="net_list">
            <PRICE_AMOUNT>2.90</PRICE_AMOUNT>
```

```
            <PRICE_LINE_AMOUNT>58.00</PRICE_LINE_AMOUNT>
            <TAX>0.16</TAX>
            <TAX_AMOUNT>9.28</TAX_AMOUNT>
            <PRICE_QUANTITY>1</PRICE_QUANTITY>
         </ARTICLE_PRICE>
      </ORDERRESPONSE_ITEM>
      <ORDERRESPONSE_ITEM>
         <LINE_ITEM_ID>P100028</LINE_ITEM_ID>
         <ARTICLE_ID>
            <SUPPLIER_AID>55-K-56</SUPPLIER_AID>
            <INTERNATIONAL_AID type="EAN">8712670911581</INTERNATIONAL_AID>
            <BUYER_AID type="BRZNR">10815</BUYER_AID>
            <DESCRIPTION_SHORT>Exklusiv Briefoeffner</DESCRIPTION_SHORT>
            <DESCRIPTION_LONG>
               Besonders formschoener Briefoeffner aus hochglanzpoliertem Edelstahl
            </DESCRIPTION_LONG>
            <MANUFACTURER_INFO>
               <MANUFACTURER_NAME>Metallpartner</MANUFACTURER_NAME>
               <MANUFACTURER_AID>EBO-368-13</MANUFACTURER_AID>
               <MANUFACTURER_TYPE_DESCR>Briefoeffner Edelmann.</MANUFACTURER_TYPE_DESCR>
            </MANUFACTURER_INFO>
         </ARTICLE_ID>
         <QUANTITY>4</QUANTITY>
         <ORDER_UNIT>C62</ORDER_UNIT>
         <ARTICLE_PRICE type="net_list">
            <PRICE_AMOUNT>30.5</PRICE_AMOUNT>
            <PRICE_LINE_AMOUNT>122.00</PRICE_LINE_AMOUNT>
            <TAX>0.16</TAX>
            <TAX_AMOUNT>19.52</TAX_AMOUNT>
            <PRICE_QUANTITY>1</PRICE_QUANTITY>
         </ARTICLE_PRICE>
      </ORDERRESPONSE_ITEM>
   </ORDERRESPONSE_ITEM_LIST>
   <ORDERRESPONSE_SUMMARY>
      <TOTAL_ITEM_NUM>2</TOTAL_ITEM_NUM>
      <TOTAL_AMOUNT>180</TOTAL_AMOUNT>
   </ORDERRESPONSE_SUMMARY>
</ORDERRESPONSE>
```

18.3 RFQ (Angebotsanforderung)


```
<?xml version="1.0" encoding="UTF-8"?>
<RFQ version="1.0" type="standard">
   <RFQ_HEADER>
      <CONTROL_INFO>
         <STOP_AUTOMATIC_PROCESSING>testimport</STOP_AUTOMATIC_PROCESSING>
         <GENERATOR_INFO>Created by openTRANS-Generator V1.7  1.09.2001</GENERATOR_INFO>
         <GENERATION_DATE>2001-08-31T18:00:30+01:00</GENERATION_DATE>
         <MIME_ROOT>/mime-dateien/</MIME_ROOT>
      </CONTROL_INFO>
      <RFQ_INFO>
         <RFQ_ID>80000221211</RFQ_ID>
         <RFQ_DATE>2001-08-31T17:14:30+01:00</RFQ_DATE>
         <ORDER_PARTIES>
            <BUYER_PARTY>
               <PARTY>
                  <PARTY_ID type="buyer_specific">FhG-IAO-4356</PARTY_ID>
                  <ADDRESS>
                     <NAME>Fraunhofer IAO</NAME>
                     <NAME2>Gebaeude TIII</NAME2>
                     <DEPARTMENT>Marktstrategie Team E-Procurement</DEPARTMENT>
                     <STREET>Nobelstr. 12</STREET>
                     <ZIP>70569</ZIP>
                     <CITY>Stuttgart</CITY>
                     <COUNTRY>DE</COUNTRY>
                     <VAT_ID>DE 129515865</VAT_ID>
                     <PHONE type="office">+49 711 970 0</PHONE>
                     <URL>http://www.iao.fhg.de</URL>
                  </ADDRESS>
               </PARTY>
            </BUYER_PARTY>
            <SUPPLIER_PARTY>
               <PARTY>
                  <PARTY_ID type="buyer_specific">UEG-FB5-BLI</PARTY_ID>
                  <ADDRESS>
                     <NAME>Universitaet Essen</NAME>
                     <NAME2>FB Wirtschaftswissenschaften/Wirtschaftsinformatik</NAME2>
                     <DEPARTMENT>Beschaffung, Logistik und Informationsmanagement</DEPARTMENT>
                     <STREET>Universitaetsstr. 9</STREET>
                     <ZIP>45117</ZIP>
                     <CITY>Essen</CITY>
                     <COUNTRY>DE</COUNTRY>
                     <PHONE type="office">+49 201 183 0</PHONE>
                     <URL>http://www.bli.uni-essen.de</URL>
                  </ADDRESS>
               </PARTY>
            </SUPPLIER_PARTY>
         </ORDER_PARTIES>
         <CATALOG_REFERENCE>
            <CATALOG_ID>12348s5121</CATALOG_ID>
            <CATALOG_VERSION>7.0</CATALOG_VERSION>
            <CATALOG_NAME>Bueromaterial 2001</CATALOG_NAME>
         </CATALOG_REFERENCE>
         <PRICE_CURRENCY>EUR</PRICE_CURRENCY>
      </RFQ_INFO>
   </RFQ_HEADER>
   <RFQ_ITEM_LIST>
      <RFQ_ITEM>
         <LINE_ITEM_ID>00100</LINE_ITEM_ID>
         <SUPPLIER_AID>55-K-31</SUPPLIER_AID>
         <INTERNATIONAL_AID type="EAN">8712670911213</INTERNATIONAL_AID>
         <BUYER_AID type="BRZNR">4711</BUYER_AID>
         <DESCRIPTION_SHORT>Standard Briefkorb DIN A4</DESCRIPTION_SHORT>
         <DESCRIPTION_LONG>Ein Klassiker unter den Briefkoerben.</DESCRIPTION_LONG>
         <QUANTITY>20</QUANTITY>
         <ORDER_UNIT>C62</ORDER_UNIT>
         <ARTICLE_PRICE type="net_list">
            <PRICE_AMOUNT>2.90</PRICE_AMOUNT>
            <PRICE_FLAG type="incl_freight">TRUE</PRICE_FLAG>
            <PRICE_LINE_AMOUNT>58.00</PRICE_LINE_AMOUNT>
            <TAX>0.16</TAX>
            <TAX_AMOUNT>9.28</TAX_AMOUNT>
            <PRICE_QUANTITY>1</PRICE_QUANTITY>
         </ARTICLE_PRICE>
      </RFQ_ITEM>
      <RFQ_ITEM>
         <LINE_ITEM_ID>00200</LINE_ITEM_ID>
         <SUPPLIER_AID>55-K-56</SUPPLIER_AID>
         <INTERNATIONAL_AID type="EAN">8712670911581</INTERNATIONAL_AID>
         <BUYER_AID type="BRZNR">10815</BUYER_AID>
         <DESCRIPTION_SHORT>Exklusiv Briefoeffner</DESCRIPTION_SHORT>
         <DESCRIPTION_LONG>
            Besonders formschoener Briefoeffner aus hochglanzpoliertem Edelstahl
```

```
            </DESCRIPTION_LONG>
            <QUANTITY>4</QUANTITY>
            <ORDER_UNIT>C62</ORDER_UNIT>
            <ARTICLE_PRICE type="net_list">
               <PRICE_AMOUNT>30.5</PRICE_AMOUNT>
               <PRICE_LINE_AMOUNT>122</PRICE_LINE_AMOUNT>
               <TAX>0.16</TAX>
               <TAX_AMOUNT>19.52</TAX_AMOUNT>
               <PRICE_QUANTITY>1</PRICE_QUANTITY>
            </ARTICLE_PRICE>
            <PARTIAL_SHIPMENT_ALLOWED>true</PARTIAL_SHIPMENT_ALLOWED>
         </RFQ_ITEM>
      </RFQ_ITEM_LIST>
      <RFQ_SUMMARY>
         <TOTAL_ITEM_NUM>2</TOTAL_ITEM_NUM>
      </RFQ_SUMMARY>
   </RFQ>
```

18.4 QUOTATION (Angebot)


```xml
<?xml version="1.0" encoding="UTF-8"?>
<QUOTATION version="1.0">
   <QUOTATION_HEADER>
      <CONTROL_INFO>
         <STOP_AUTOMATIC_PROCESSING>testimport</STOP_AUTOMATIC_PROCESSING>
         <GENERATOR_INFO>Created by openTRANS-Generator V1.7  1.09.2001</GENERATOR_INFO>
         <GENERATION_DATE>2001-09-04T18:00:30+01:00</GENERATION_DATE>
         <MIME_ROOT>/mime-dateien/</MIME_ROOT>
      </CONTROL_INFO>
      <QUOTATION_INFO>
         <QUOTATION_ID>80000221211</QUOTATION_ID>
         <QUOTATION_DATE>2001-09-04T17:14:30+01:00</QUOTATION_DATE>
         <VALID_START_DATE>2001-09-04T17:14:30+01:00</VALID_START_DATE>
         <VALID_END_DATE>2001-09-25T10:00:00+01:00</VALID_END_DATE>
         <ORDER_PARTIES>
            <BUYER_PARTY>
               <PARTY>
                  <PARTY_ID type="buyer_specific">FhG-IAO-4356</PARTY_ID>
                  <ADDRESS>
                     <NAME>Fraunhofer IAO</NAME>
                     <NAME2>Gebaeude TIII</NAME2>
                     <DEPARTMENT>Marktstrategie Team E-Procurement</DEPARTMENT>
                     <CONTACT>
                        <CONTACT_NAME>Sekretariat</CONTACT_NAME>
                        <PHONE type="office">+49 711 970 2331</PHONE>
                        <FAX>+49 711 970 2300</FAX>
                        <URL>http://www.iao.fhg.de</URL>
                     </CONTACT>
                     <STREET>Nobelstr. 12</STREET>
                     <ZIP>70569</ZIP>
                     <CITY>Stuttgart</CITY>
                     <COUNTRY>DE</COUNTRY>
                     <VAT_ID>DE 129515865</VAT_ID>
                     <PHONE type="office">+49 711 970 0</PHONE>
                     <URL>http://www.iao.fhg.de</URL>
                  </ADDRESS>
               </PARTY>
            </BUYER_PARTY>
            <SUPPLIER_PARTY>
               <PARTY>
                  <PARTY_ID type="buyer_specific">UEG-FB5-BLI</PARTY_ID>
                  <ADDRESS>
                     <NAME>Universitaet Essen</NAME>
                     <NAME2>FB Wirtschaftswissenschaften/Wirtschaftsinformatik</NAME2>
                     <DEPARTMENT>Beschaffung, Logistik und Informationsmanagement</DEPARTMENT>
                     <CONTACT>
                        <CONTACT_NAME>Volker Schmitz</CONTACT_NAME>
                        <PHONE type="office">+49 201 183 4084</PHONE>
                        <PHONE type="office">+49 201 183 4076</PHONE>
                        <FAX>+49 201 183 934084</FAX>
                        <EMAIL>volker.schmitz@uni-essen.de</EMAIL>
                     </CONTACT>
                     <STREET>Universitaetsstr. 9</STREET>
                     <ZIP>45117</ZIP>
                     <CITY>Essen</CITY>
                     <COUNTRY>DE</COUNTRY>
                     <PHONE type="office">+49 201 183 0</PHONE>
                     <URL>http://www.bli.uni-essen.de</URL>
                  </ADDRESS>
               </PARTY>
            </SUPPLIER_PARTY>
         </ORDER_PARTIES>
         <CATALOG_REFERENCE>
            <CATALOG_ID>12348s5121</CATALOG_ID>
            <CATALOG_VERSION>7.0</CATALOG_VERSION>
            <CATALOG_NAME>Bueromaterial 2001</CATALOG_NAME>
         </CATALOG_REFERENCE>
         <PRICE_CURRENCY>EUR</PRICE_CURRENCY>

   <TERMS_AND_CONDITIONS>http://www.iao.fhg.de/procurement/agb.pdf</TERMS_AND_CONDITIONS>
         <TRANSPORT>
            <INCOTERM>CFR</INCOTERM>
            <LOCATION>Zentale Postannahmestelle</LOCATION>
            <TRANSPORT_REMARK>Bitte bei Fr. Mueller, 2. Tuere, abgeben</TRANSPORT_REMARK>
         </TRANSPORT>
      </QUOTATION_INFO>
   </QUOTATION_HEADER>
   <QUOTATION_ITEM_LIST>
      <QUOTATION_ITEM>
         <LINE_ITEM_ID>00100</LINE_ITEM_ID>
         <SUPPLIER_AID>55-K-31</SUPPLIER_AID>
         <INTERNATIONAL_AID type="EAN">8712670911213</INTERNATIONAL_AID>
```

```
            <BUYER_AID type="BRZNR">4711</BUYER_AID>
            <DESCRIPTION_SHORT>Standard Briefkorb DIN A4</DESCRIPTION_SHORT>
            <DESCRIPTION_LONG>Ein Klassiker unter den Briefkoerben.</DESCRIPTION_LONG>
            <MANUFACTURER_INFO>
               <MANUFACTURER_NAME>Plastikpartner</MANUFACTURER_NAME>
               <MANUFACTURER_AID>123-RD-67-U</MANUFACTURER_AID>
               <MANUFACTURER_TYPE_DESCR>Briefkorb 02133</MANUFACTURER_TYPE_DESCR>
            </MANUFACTURER_INFO>
            <QUANTITY>20</QUANTITY>
            <ORDER_UNIT>C62</ORDER_UNIT>
            <ARTICLE_PRICE type="net_list">
               <PRICE_AMOUNT>2.90</PRICE_AMOUNT>
               <PRICE_FLAG type="incl_freight">TRUE</PRICE_FLAG>
               <PRICE_FLAG type="incl_assurance">FALSE</PRICE_FLAG>
               <PRICE_FLAG type="incl_duty">TRUE</PRICE_FLAG>
               <PRICE_LINE_AMOUNT>58.00</PRICE_LINE_AMOUNT>
               <TAX>0.16</TAX>
               <TAX_AMOUNT>9.28</TAX_AMOUNT>
               <PRICE_QUANTITY>1</PRICE_QUANTITY>
            </ARTICLE_PRICE>
            <PARTIAL_SHIPMENT_ALLOWED>false</PARTIAL_SHIPMENT_ALLOWED>
            <TRANSPORT>
               <INCOTERM>CFR</INCOTERM>
               <LOCATION>Zentale Postannahmestelle</LOCATION>
               <TRANSPORT_REMARK>Bitte bei Fr. Mueller, 2. Tuere, abgeben</TRANSPORT_REMARK>
            </TRANSPORT>
         </QUOTATION_ITEM>
         <QUOTATION_ITEM>
            <LINE_ITEM_ID>00200</LINE_ITEM_ID>
            <SUPPLIER_AID>55-K-56</SUPPLIER_AID>
            <INTERNATIONAL_AID type="EAN">8712670911581</INTERNATIONAL_AID>
            <BUYER_AID type="BRZNR">10815</BUYER_AID>
            <DESCRIPTION_SHORT>Exklusiv Briefoeffner</DESCRIPTION_SHORT>
            <DESCRIPTION_LONG>
                  Besonders formschoener Briefoeffner aus hochglanzpoliertem Edelstahl
            </DESCRIPTION_LONG>
            <MANUFACTURER_INFO>
               <MANUFACTURER_NAME>Metallpartner</MANUFACTURER_NAME>
               <MANUFACTURER_AID>EBO-368-13</MANUFACTURER_AID>
               <MANUFACTURER_TYPE_DESCR>Briefoeffner Edelmann</MANUFACTURER_TYPE_DESCR>
            </MANUFACTURER_INFO>
            <QUANTITY>4</QUANTITY>
            <ORDER_UNIT>C62</ORDER_UNIT>
            <ARTICLE_PRICE type="net_list">
               <PRICE_AMOUNT>30.5</PRICE_AMOUNT>
               <PRICE_LINE_AMOUNT>122</PRICE_LINE_AMOUNT>
               <TAX>0.16</TAX>
               <TAX_AMOUNT>19.52</TAX_AMOUNT>
               <PRICE_QUANTITY>1</PRICE_QUANTITY>
            </ARTICLE_PRICE>
            <PARTIAL_SHIPMENT_ALLOWED>true</PARTIAL_SHIPMENT_ALLOWED>
            <TRANSPORT>
               <INCOTERM>CFR</INCOTERM>
               <LOCATION>Zentale Postannahmestelle</LOCATION>
               <TRANSPORT_REMARK>Bitte bei Fr. Mueller, 2. Tuere, abgeben</TRANSPORT_REMARK>
            </TRANSPORT>
         </QUOTATION_ITEM>
      </QUOTATION_ITEM_LIST>
      <QUOTATION_SUMMARY>
         <TOTAL_ITEM_NUM>2</TOTAL_ITEM_NUM>
         <TOTAL_AMOUNT>180</TOTAL_AMOUNT>
      </QUOTATION_SUMMARY>
   </QUOTATION>
```

Erklärung

Ich versichere, dass ich diese Arbeit selbständig verfasst und nur die angegebenen Hilfsmittel verwendet habe.

Stuttgart, den 13.11.2003

_____

Sven Sprandel