

# XML im Bereich EAI und B2B

Einführung und Überblick

von

Markus Schütze

# Übersicht I

- XML im Bereich EDI
  - Electronic Data Interchange (EDI)
  - Der klassische EDI-Vorgang
  - Der UN/EDIFACT – Standard
  - Aufbau einer EDIFACT-Datei
  - Probleme im klassischem EDI
  - XML/EDI
  - Änderung der EDI-Systemstruktur unter Einfluss von XML
- Ist XML/EDI die Lösung ?

# Übersicht II

- Enterprise Application Integration
  - Was ist EAI ?
  - Qualitäten der Anwendungsintegration
  - EAI System Ansätze (P2P, HUB-Spoke)
  - mqSeries
  - Die Rolle von XML in EAI
    - RPC Aufrufe mit XML
  - Web-Services
    - Systemstruktur
    - UDDI
    - WSDL
    - SOAP
- Zusammenfassung

# Electronic Data Interchange (EDI)

## Definition (EDI):

*Unter Electronic Data Interchange versteht man den elektronischen Datenaustausch über Geschäftstransaktionen (Bestellungen, Rechnungen, Überweisungen, Warenerklärungen etc.) zwischen Betrieben. Die Daten werden in Form von strukturierten, nach vereinbarten Regeln formatierten Nachrichten übertragen. Dadurch ist es dem Empfänger möglich, die Daten direkt in sein Anwendungsprogramm weiterzuverarbeiten.*

# Electronic Data Interchange (EDI)

## Definition (EDI):

*Unter Electronic Data Interchange versteht man den elektronischen Datenaustausch über Geschäftstransaktionen (Bestellungen, Rechnungen, Überweisungen, Warenerklärungen etc.) zwischen Betrieben. Die Daten werden in Form von strukturierten, nach vereinbarten Regeln formatierten Nachrichten übertragen. Dadurch ist es dem Empfänger möglich, die Daten direkt in sein Anwendungsprogramm weiterzuverarbeiten.*



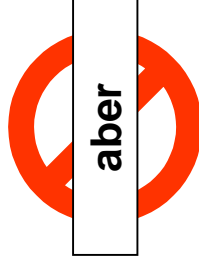
**Datenaustausch mit  
Minimum an menschlichen  
Eingriffen**

# Ziele von EDI

- Konsistente und beschleunigte zwischenbetriebliche Workflows
- Vermeidung wiederholter Datenerfassung, Medienbrüche
- Intensivierung von Partnerbetreuung, Kommunikation
- Kosteneinsparungen (z.B.: Abbau von Lagerbeständen via JiT)
- Reduzierung der Verwaltungskosten
- Verbesserung der Kundenservice
- Erhaltung der Wettbewerbsfähigkeit

# Ziele von EDI

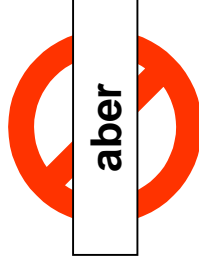
- Konsistente und beschleunigte zwischenbetriebliche Workflows
- Vermeidung wiederholter Datenerfassung, Medienbrüche
- Intensivierung von Partnerbetreuung, Kommunikation
- Kosteneinsparungen (z.B.: Abbau von Lagerbeständen via JiT)
- Reduzierung der Verwaltungskosten
- Verbesserung der Kundenservice
- Erhaltung der Wettbewerbsfähigkeit



ca. **5 – 10 %** der Unternehmen,  
für die EDI vorteilhaft wäre, nutzen EDI

# Ziele von EDI

- Konsistente und beschleunigte zwischenbetriebliche Workflows
- Vermeidung wiederholter Datenerfassung, Medienbrüche
- Intensivierung von Partnerbetreuung, Kommunikation
- Kosteneinsparungen (z.B.: Abbau von Lagerbeständen via JiT)
- Reduzierung der Verwaltungskosten
- Verbesserung der Kundenservice
- Erhaltung der Wettbewerbsfähigkeit



ca. 5 – 10 %  
für die EDI  
nutzen EDI

WARUM ?



# Standardvielfalt

Bezeichnung	Ursprungsbranche	Kurzbeschreibung
CEFIC	Chemische Industrie	Conseil Europeen de Federations del'Industrie Chimique
SEDAS	Konsumgüterindustrie	Standartregeln einheitlicher Datenaustauschsysteme
EDIFICE	Computer und Elektroindustrie	Electronic Data Interchange Forum for Companies
TRADACOMS	Trade Data Communication Standard	Trade Data Communication Standard
RINET	Versicherungen	Reinsurance and Insurance Net
SWIFT	Finanzgewerbe	Society for Worldwide Interbank Financial Telecom
ODETTE	Automobilindustrie	Organization for Data Exchange by Teletransmissions in Europe
VDA	Automobilindustrie	Verband der Automobilindustrie

# Standardvielfalt

Bezeichnung	Ursprungsbranche	Kurzbeschreibung
CEFIC	Chemische Industrie	Conseil de l'Europe des Fabricants de Produits Chimiques
SEDAS	Konsumgüterindustrie	Société Européenne des Distributeurs de Produits Industriels
EDIFICE	Computer- und Elektroanlagen	European Data Interchange for Companies
TRADACOMS		Trade Data Communication Standard
RINET		Reinsurance and Insurance Net
SWIFT	Bank- und Finanzgewerbe	Society for Worldwide Interbank Financial Telecom
EDIFACT	Automobilindustrie	Organization for Data Exchange by Teletransmissions in Europe
VDA	Automobilindustrie	Verband der Automobilindustrie

Lösung ?!

## Der UN/EDIFACT- Standard

- Globaler, branchenübergreifender Standard unter Schirmherrschaft der United Nations
- Anderer branchenübergreifender Standard: X12 (ist aber nur national in Amerika gültig)
- EDIFACT ist sehr komplex
- besteht aus Subsets, diese vereinigen unterschiedlich Standards aus unterschiedlichen Branchen (z.B. EANCOM)

# EDIFACT- Messages

- Subsets bestehen aus Messages , die Geschäftsvorfälle kapseln.

Beispiel :

<b>APERAK</b>	Anwendungsfehler- und Bestätigungsantwort
<b>AUTHOR</b>	Autorisierungsnachricht
.....	
<b>FINCAN</b>	Stornonachricht
<b>FINPAY</b>	Multipler Interbank-Zahlungsauftrag
<b>GENERAL</b>	Allgemeine Nachricht
<b>GESMES</b>	Allgemeine statistische Nachricht
....	
<b>ORDERS</b>	Bestellung
<b>ORDERSP</b>	Bestellantwort
....	
<b>WKGRRE</b>	Antrag auf Arbeitserlaubnis

# EDIFACT- Messages

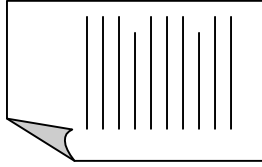
- Subsets bestehen aus Messages , die Geschäftsvorfälle kapseln.

Beispiel :

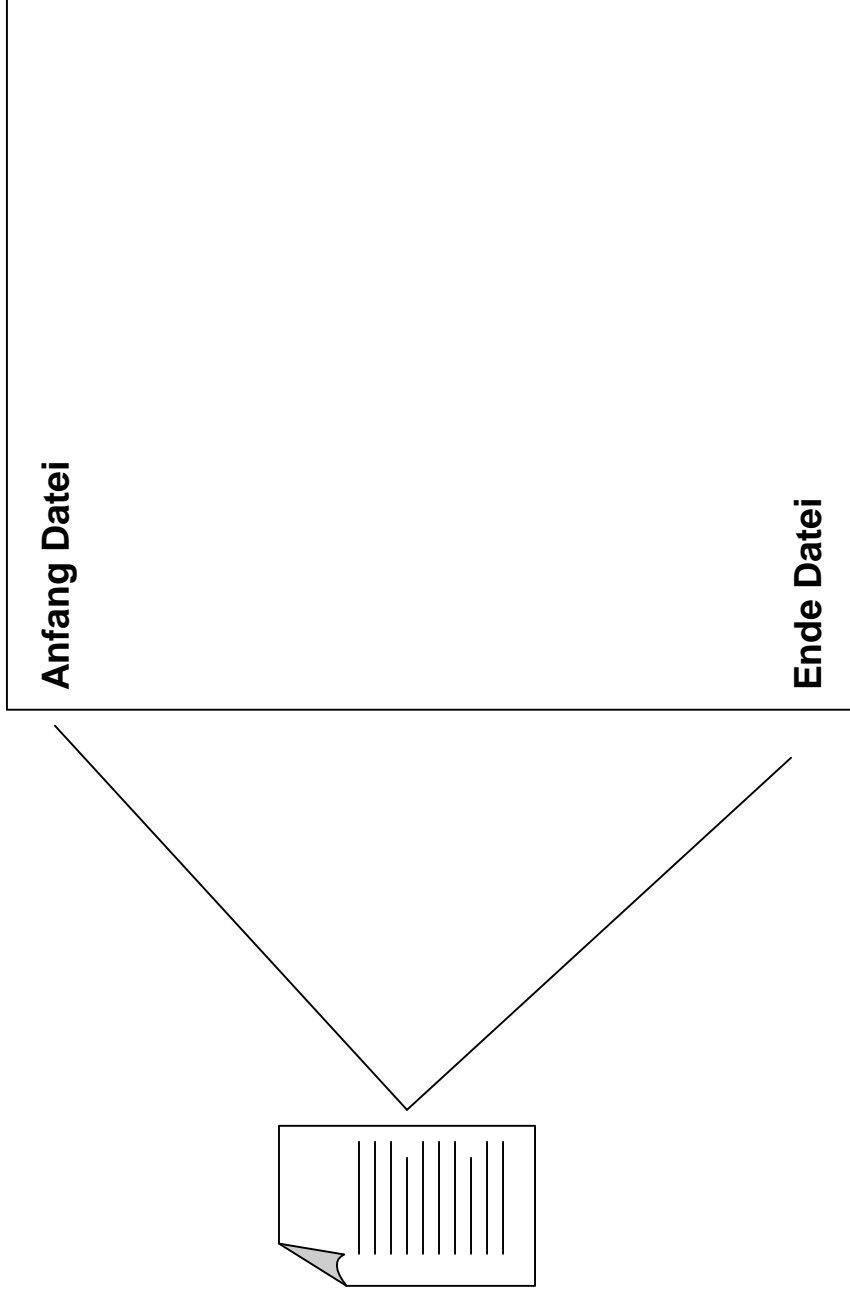
<b>APERAK</b>	Anwendungsfehler- und Bestätigungsantwort
<b>AUTHOR</b>	Autorisierungsnachricht
.....	
<b>FINCAN</b>	Stornonachricht
<b>FINPAY</b>	Multipler Interbank
<b>GENERAL</b>	Allgemein
<b>GESMES</b>	
.....	
<b>W</b>	Antrag auf Arbeiterlaubnis

Message Implementation Guides (MIG's)  
 definieren den Aufbau einer Message  
 Nach DIN Norm ISO 9735

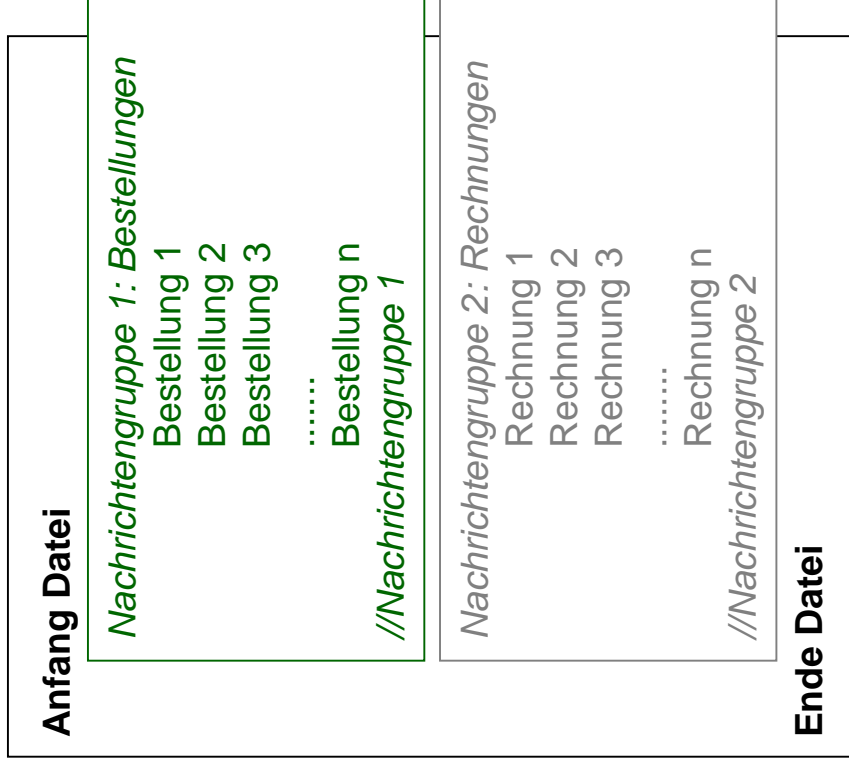
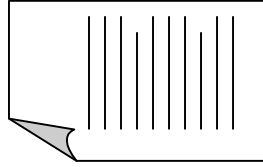
# Aufbau einer EDIFACT-Datei



# Aufbau einer EDIFACT-Datei

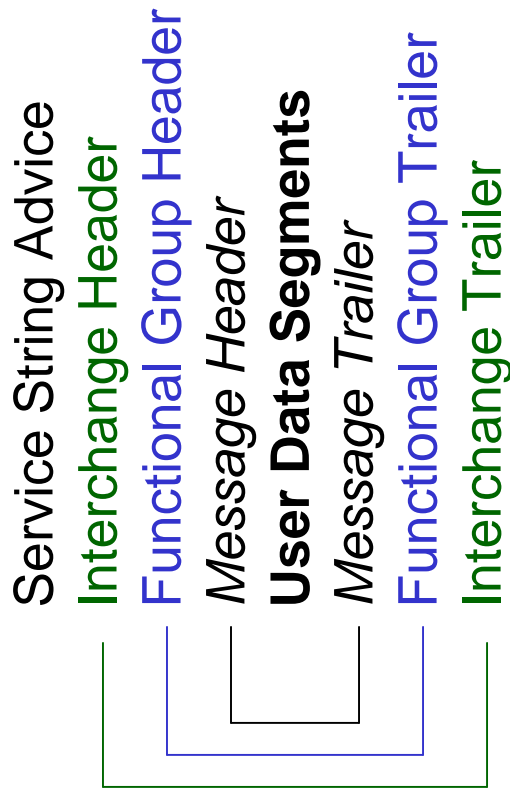


# Aufbau einer EDIFACT-Datei





# Aufbau einer EDIFACT - Message



EDIFACT -  
Kürzel

**UNA**

**UNB**

**UNG**

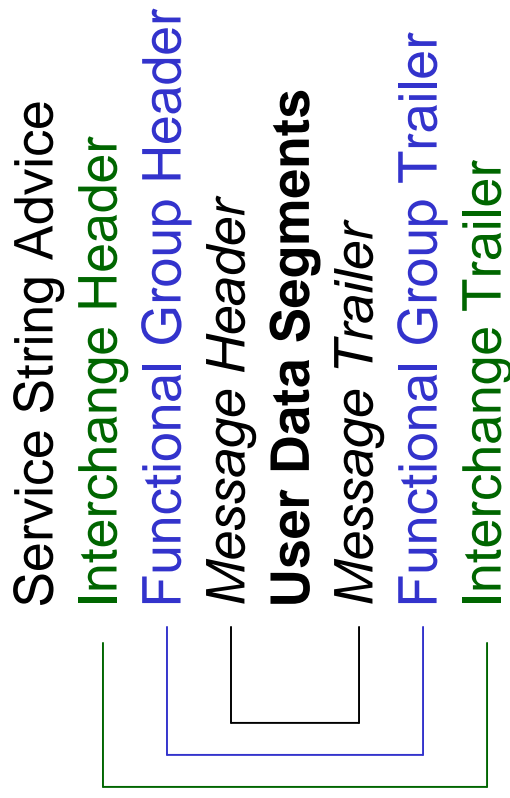
**UNH**

**UNT**

**UNE**

**UNZ**

# Aufbau einer EDIFACT - Message



EDIFACT -  
Kürzel

**UNA**

**UNB**

**UNG**

**UNH**

**UNT**

**UNE**

**UNZ**

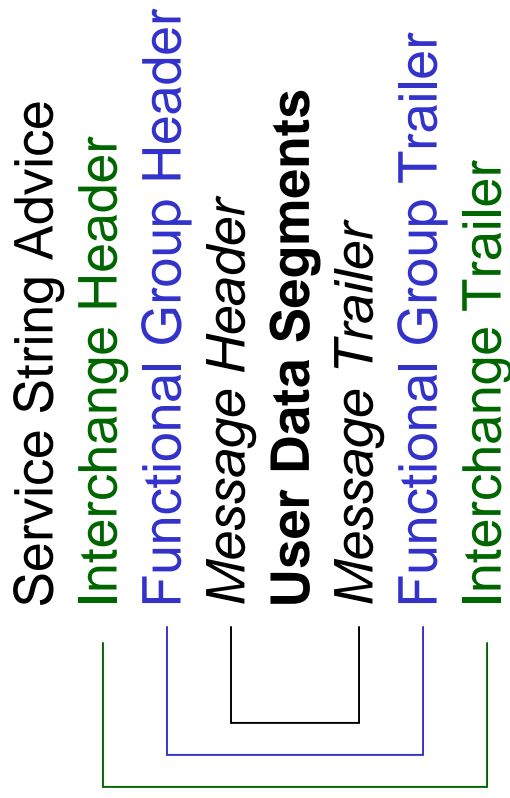
zwingend

zwingend

zwingend

zwingend

# Aufbau einer EDIFACT - Message



EDIFACT -  
Kürzel

**UNA**

*optional*

**UNB**

*zwingend*

**UNG**

*optional*

**UNH**

*zwingend*

**UNT**

*zwingend*

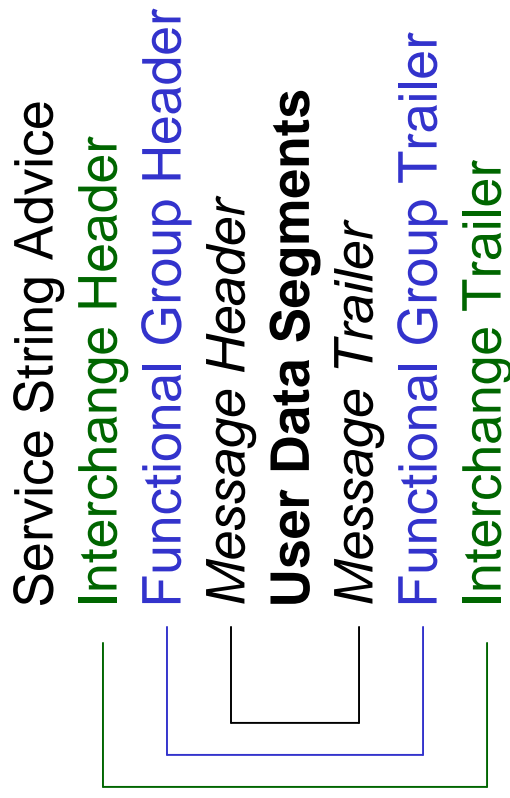
**UNE**

*optional*

**UNZ**

*zwingend*

# Aufbau einer EDIFACT - Message



EDIFACT -  
Kürzel

**UNA**

*optional*

**UNB**

*zwingend*

**UNG**

*optional*

**UNH**

*zwingend*

*wie erfordert*

**UNT**

*zwingend*

**UNE**

*optional*

**UNZ**

*zwingend*

# Aufbau einer EDIFACT - Message

Legt Trennzeichen fest

Legt Zeichensatz, Absender, Empfänger fest  
Spezifiziert Messagegruppe

Kennzeichnet verwendeten Standard, Nachrichtentyp

Letzte Segment einer Nachricht

Beendet eine MessageGruppe

Kennzeichnet das Ende einer EDIFACT-Datei  
(# der übertragenen Nachrichten,  
Nachrichtenreferenznummer)

EDIFACT -  
Kürzel

**UNA**

optional

**UNB**

zwingend

**UNG**

optional

**UNH**

zwingend

wie erfordert

**UNT**

zwingend

**UNE**

optional

**UNZ**

zwingend

# Ein Beispiel

**UNA**:+,? '**UNB**+UNOA:2+FHPEDAL+HUBERGMBH+990802:1557+  
9908021557'**UNH**+INVOIC0001+INVOIC:D:96A:UN'BGM+380+  
9908001+9'DTM+3:19990802:102'RFF+ON:00010001'DTM+4  
:19999715:102'NAD+SE++Fahrradhandel Pedal++Waginge  
rstr. 5+München++81549'NAD+BY++Huber GmbH++Obstgas  
se 2+München++81549'LIN+1++4711.001'IMD+F++:::Fahr  
rad, Damen'QTY+47:1:PCE'MOA+66:750'PRI+AAA:750'LIN  
+2++4711.002'IMD+F++:::Luftpumpe, Stand-'QTY+47:1:  
PCE'MOA+66:19,9'PRI+AAA:19,9'LIN+3++4711.003'IMD+F  
++:::Ersatzventil'QTY+47:3:PCE'MOA+66:7,5'PRI+AAA:  
2,5'UNS+S'MOA+79:777,4'MOA+124:124,38'MOA+128:901,  
78'TAX+7+VAT+++:::16+S'**UNT**+28+INVOIC0001'**UNZ**+1+990  
8021557'

# Ein Beispiel

**UNA**:+,? '**UNB**+UNOA:2+FHPEDAL+HUBERGMBH+990802:1557+  
9908021557'**UNH**+INVOIC0001+INVOIC:D:96A:UN'BGM+380+  
9908001+9'**DTM**+3:**19990802**:102'RFF+ON:00010001'DTM+4  
:19999715:102'NAD+SE++Fahrradhandel Pedal++Waginge  
rstr. 5+München++81549'NAD+BY++Huber GmbH++Obstgas  
se 2+München++81549'LIN+1++4711.001'IMD+F++:::Fahr  
rad, Damen'QTY+47:1:PCE'MOA+66:750'PRI+AAA:750'LIN  
+2++4711.002'IMD+F++:::Luftpumpe, Stand-'QTY+47:1:  
PCE'MOA+66:19,9'PRI+AAA:19,9'LIN+3++4711.003'IMD+F  
++:::Ersatzventil'QTY+47:3:PCE'MOA+66:7,5'PRI+AAA:  
2,5'UNS+S'MOA+79:777,4'MOA+124:124,38'MOA+128:901,  
78'TAX+7+VAT+++:::16+S'**UNT**+28+INVOIC0001'**UNZ**+1+990  
8021557'


# Ein Beispiel


**'DTM+3:19990802:102'**

DTM - 2 Date/time/period		D6	D
UN/EDIFACT Directory 96.A C			
A segment identifying the effective date and/or time the rate of exchange was fixed. The other occurrence identifies the reference date.			
Segment number : 20			
UN/EDIFACT Directory 96.A		D6 Description	
<b>C507</b>	<b>Date/time/period</b>	M	
2005	Date/time/period qualifier	M an..3	*R I71 = Reference date/time. Usage of this segment requires the use of the following RFF segment.
2380	Date/time/period	R	
2379	Date/time/period format qualifier	R C an..35 R C an..3	* 102 = CCYYMMDD Code 102 is recommended. * 203 = CCYYMMDDHHMM
<u>Example:</u>			
DTM+171:19970630:102'			

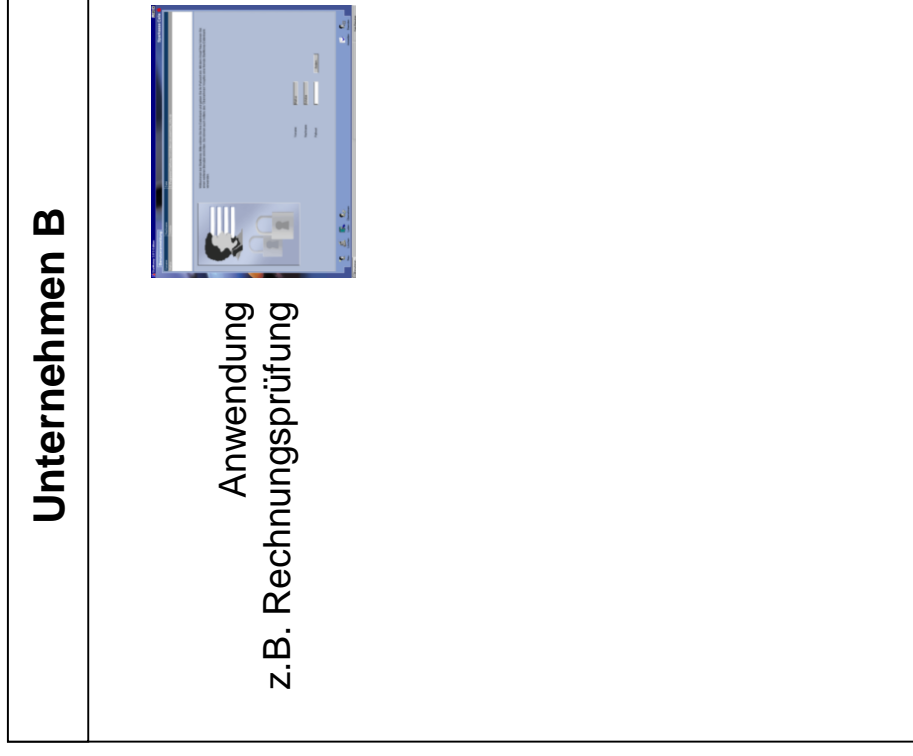
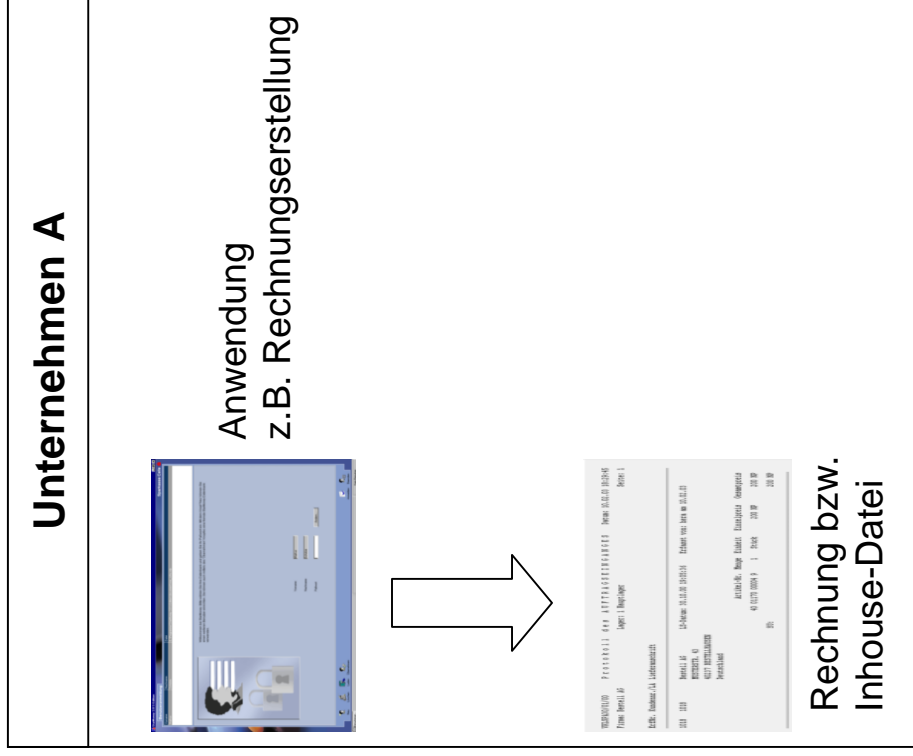


# Der klassische EDI - Vorgang

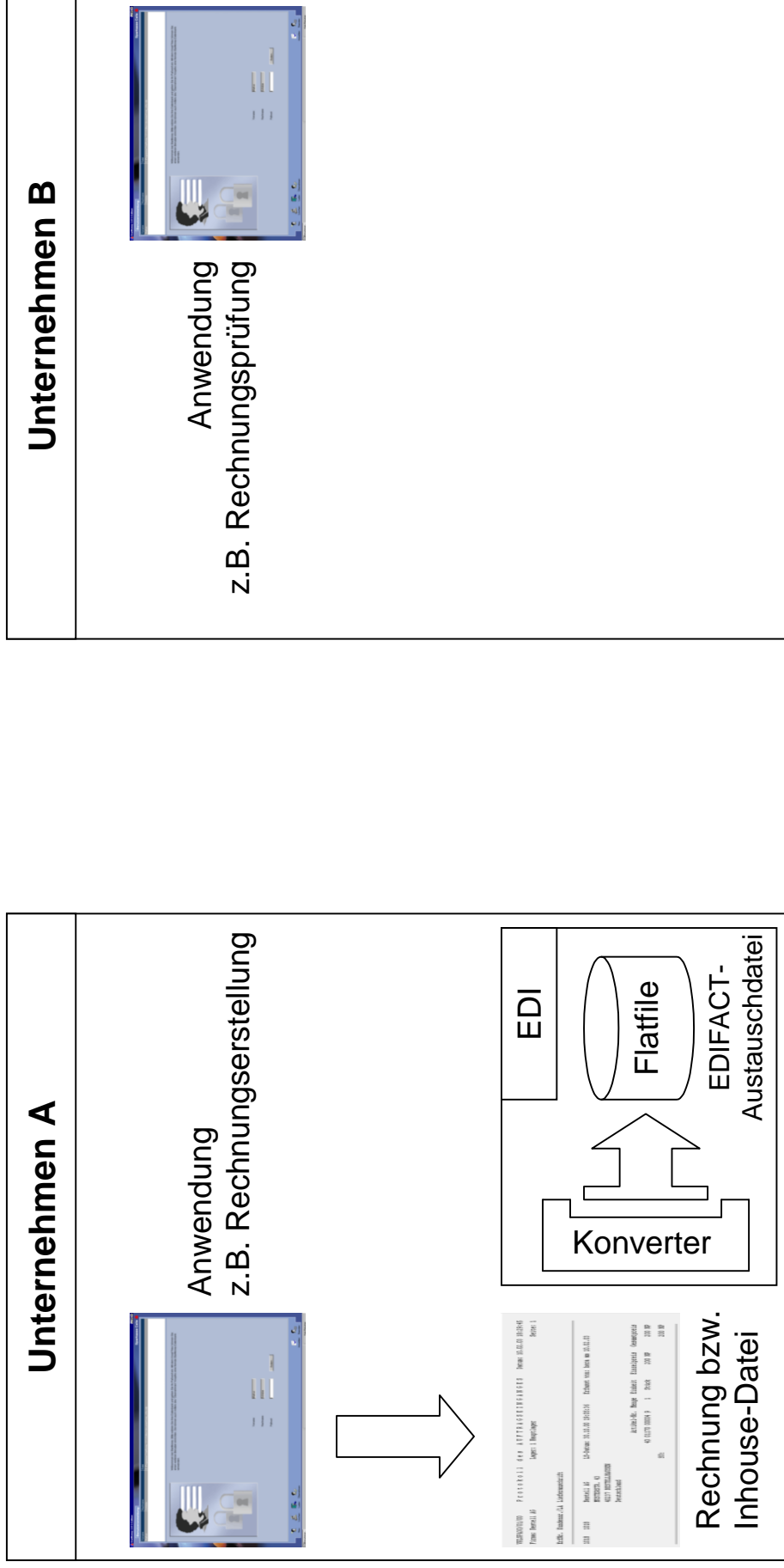
<p><b>Unternehmen A</b></p>	 <p>Anwendung z.B. Rechnungserstellung</p>
-----------------------------	---

<p><b>Unternehmen B</b></p>	 <p>Anwendung z.B. Rechnungsprüfung</p>
-----------------------------	--

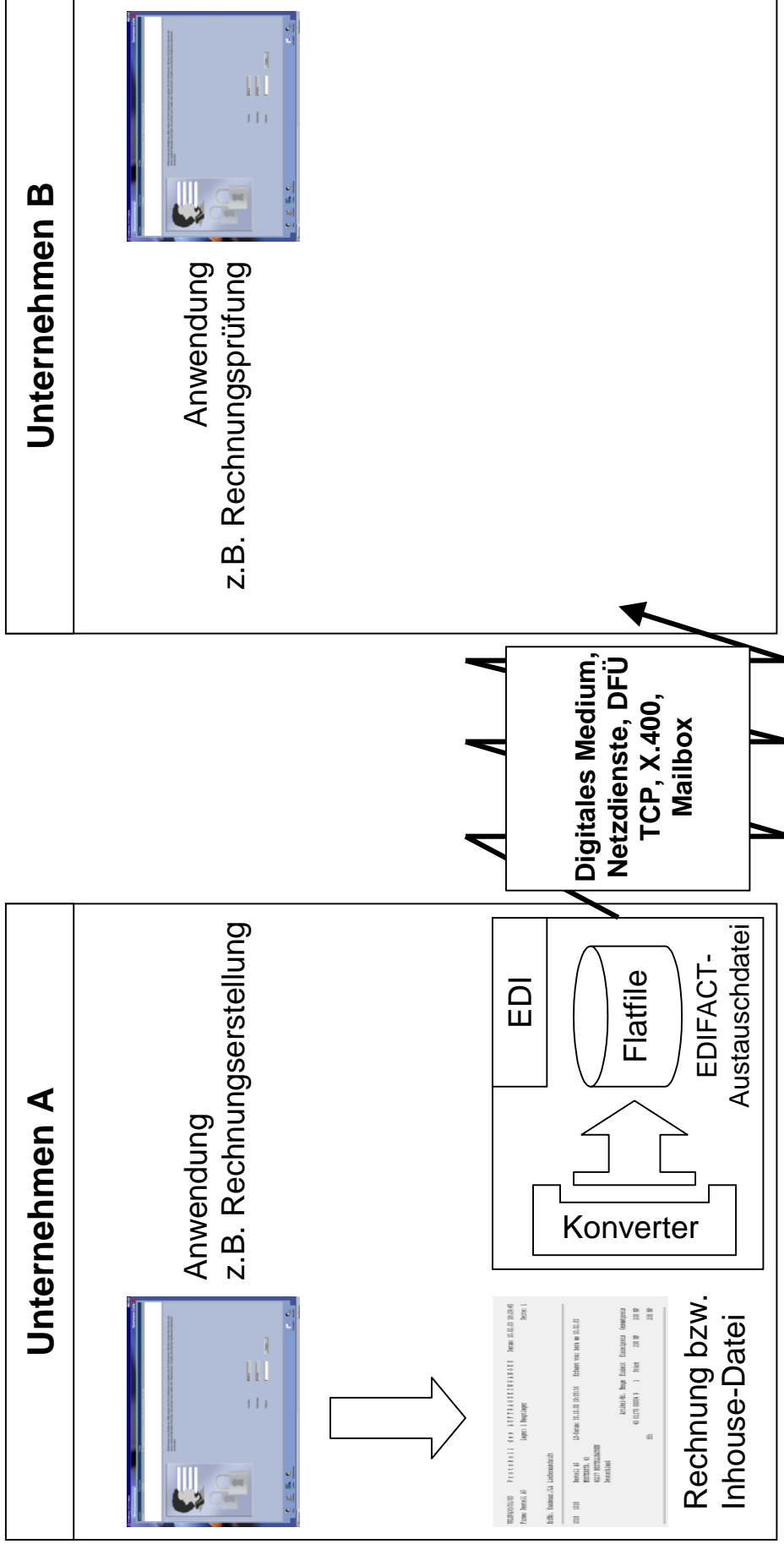
# Der klassische EDI - Vorgang



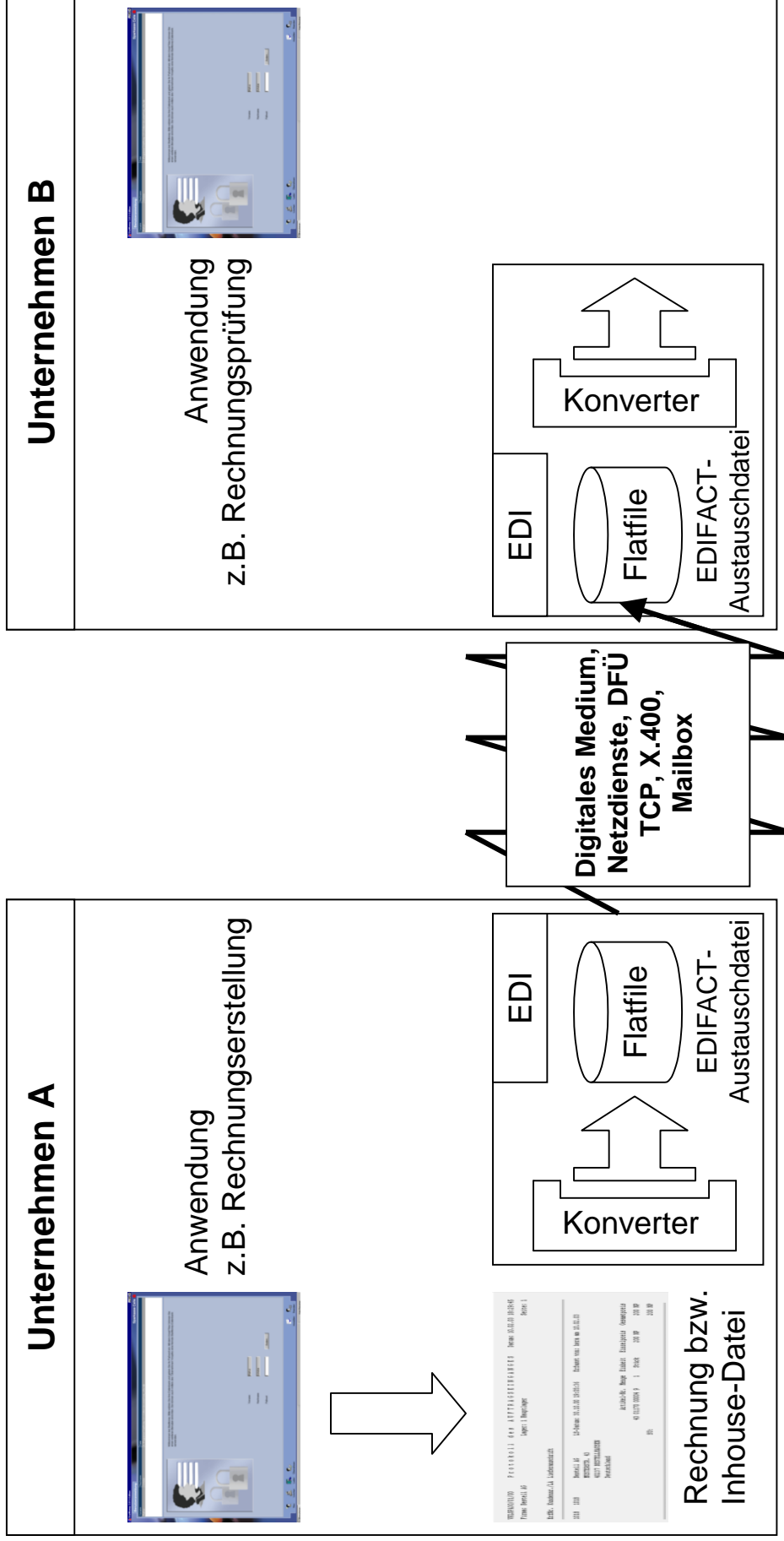
# Der klassische EDI - Vorgang



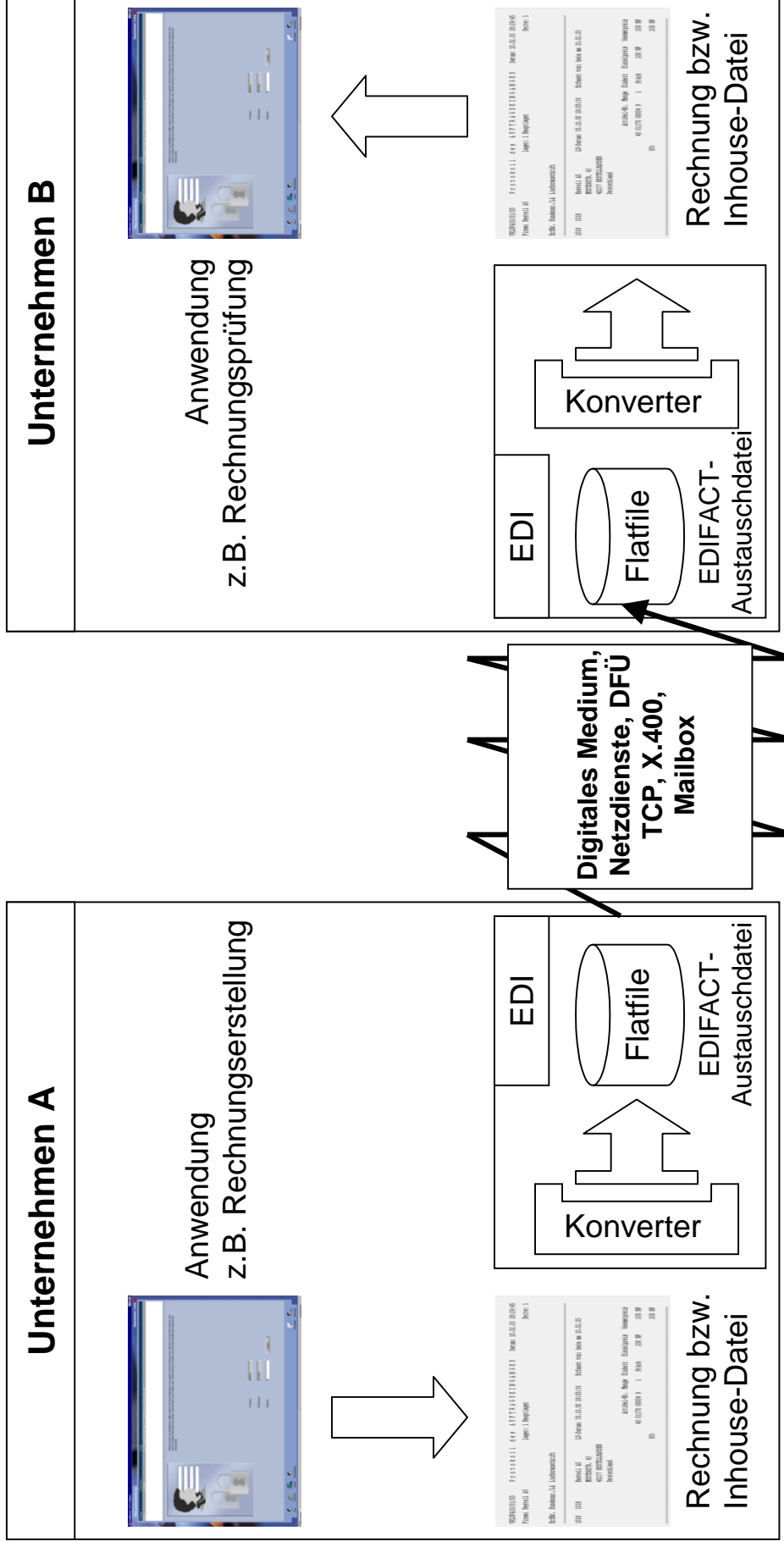
# Der klassische EDI - Vorgang



# Der klassische EDI - Vorgang



# Der klassische EDI - Vorgang



# EDI-Probleme

- EDI-Formate sind ‚starre‘- Regelwerke
- EDI-Formate sind eng mit Geschäftsvorfällen verknüpft
  - Erfordert zentralisierte Verwaltung der Standards
  - Langwierige Weiterentwicklung der Standards bei neuen Geschäftsprozessen
- EDI- Lösungen sind plattformabhängig



- **Klassisches EDI ist teuer**
- wird den ‚Kleinen‘ von den ‚Grossen‘ aufgezwungen

# Mögliche Lösung : XML

## Vorteile von XML

- XML ist leichter lesbar
- Durch DTD's und XML-Schemas werden Syntax und Semantik beschrieben
  - XML Dokumente sind selbstbeschreibend
- Nachrichten können leicht validiert werden
- XML ermöglicht baumartige Traversierung der Dokumente
  - einfacher Zugriff auf bestimmte Elemente
- XML – Dokumente kann durch Verwendung von Stylesheets in unterschiedlichen Formaten oder Ansichten dargestellt werden
- XML ermöglicht eine relativ einfache Konvertierung des EDIFACT-Formates in XML



# XML-EDI Produkte

- **cXML (Commerce XML)**  
geschaffener Standard zum Austausch von Geschäftsnachrichten (ARIBA)
- **xCBL (XML based CommonBusiness Library)**  
Standard zum Austausch von strukturierten Daten wie Rechnungen, Bestellungen, etc. (CommerceOne)
- **RosettaNet**  
XML-Framework, Nachrichtenformate, Abläufe, Regeln von Transaktionen (mehrere Hersteller)
- **Electronic Catalog XML**
- **OCF (Open Catalog Format)**
- **Produkt data Markup Language**

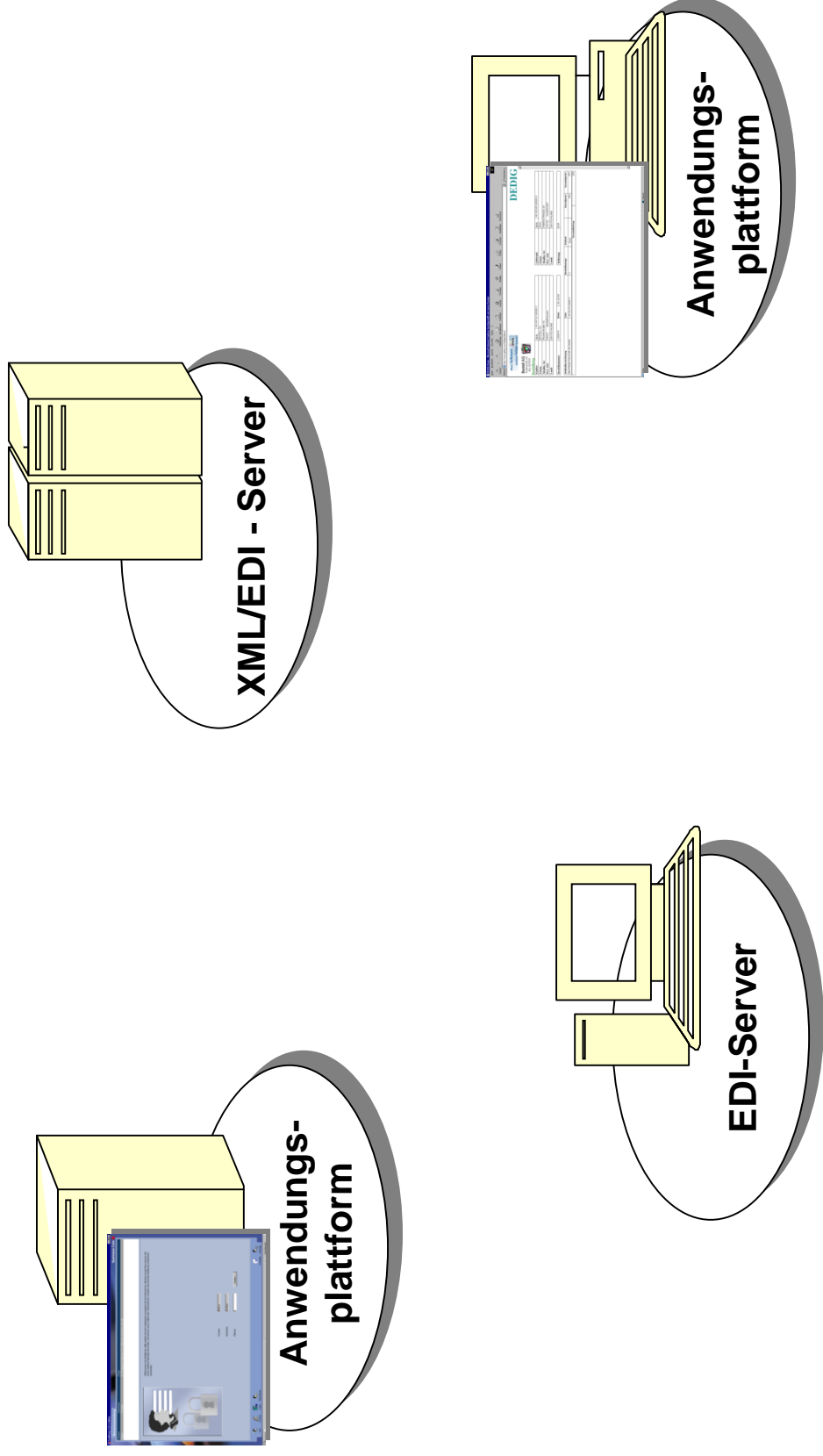
# XML/EDI

- Vorteile:
  - XML/EDI ist kompatibel zum klassischen EDI
  - Schnittstellen ermöglichen Weiterverwendung von ,ältern' EDI-Systemen
  - Globale Repository's
  - Einfache Erweiterung bestehender Anwendungen
  - Browserfähig

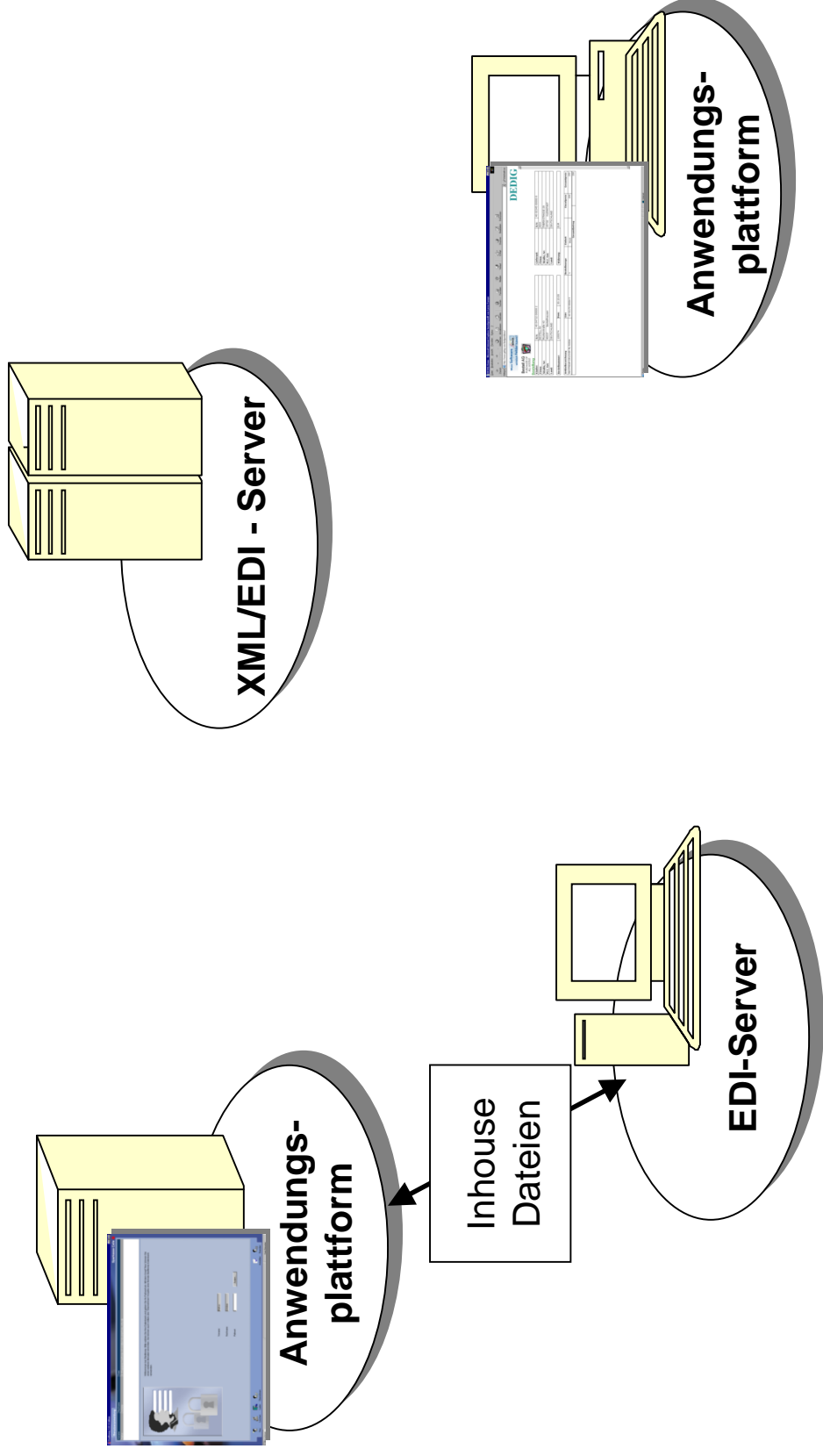
# Änderung der klassischen Systemstruktur I

- XML/EDI ist Verbund verschiedener Technologien
  - **XML**
  - **EDI-Standards**  
z.B. EDIFACT-MIBS
  - **Templates**  
DTD's, XML-Schemas, XSL-Definitionen
  - **Agenten**  
z.B.: Java- Applet's, führen von den Templates definierte Aktionen aus
  - **Repositories**  
sichern Wiederverwendbarkeit, globale ‚Nachschlagewerke‘ für Agenten

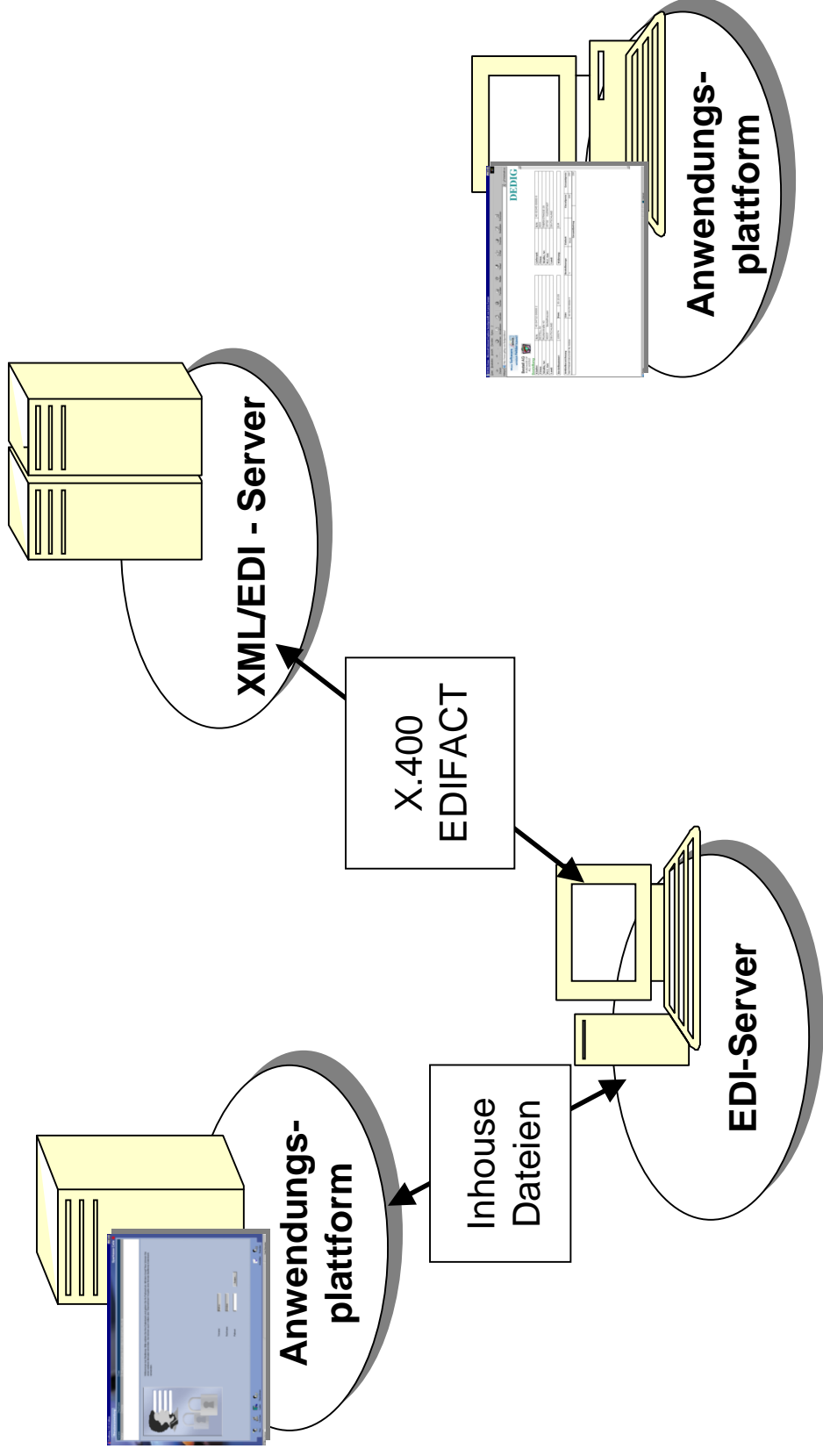
# Änderung der klassischen Systemstruktur II



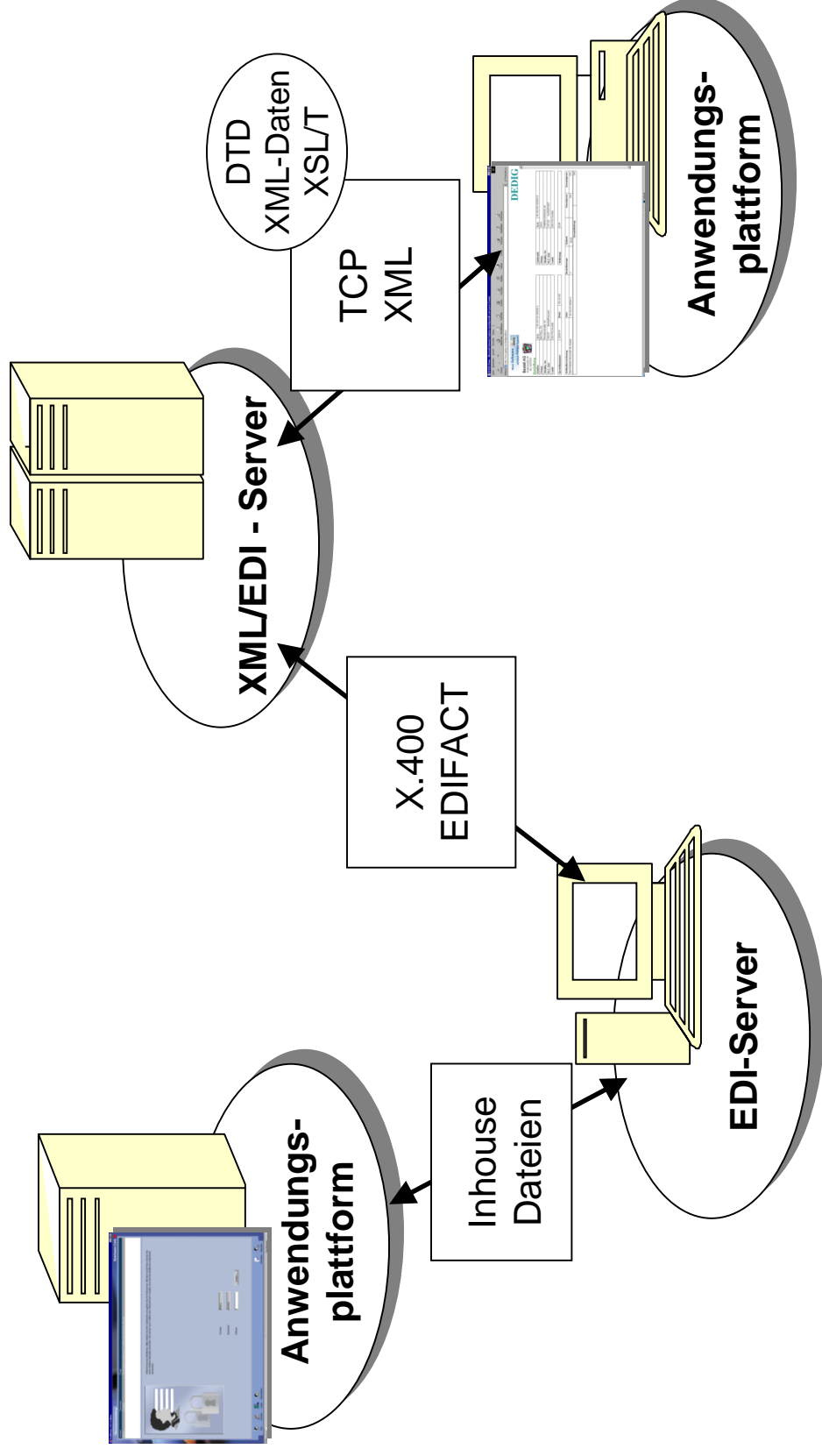
# Änderung der klassischen Systemstruktur II



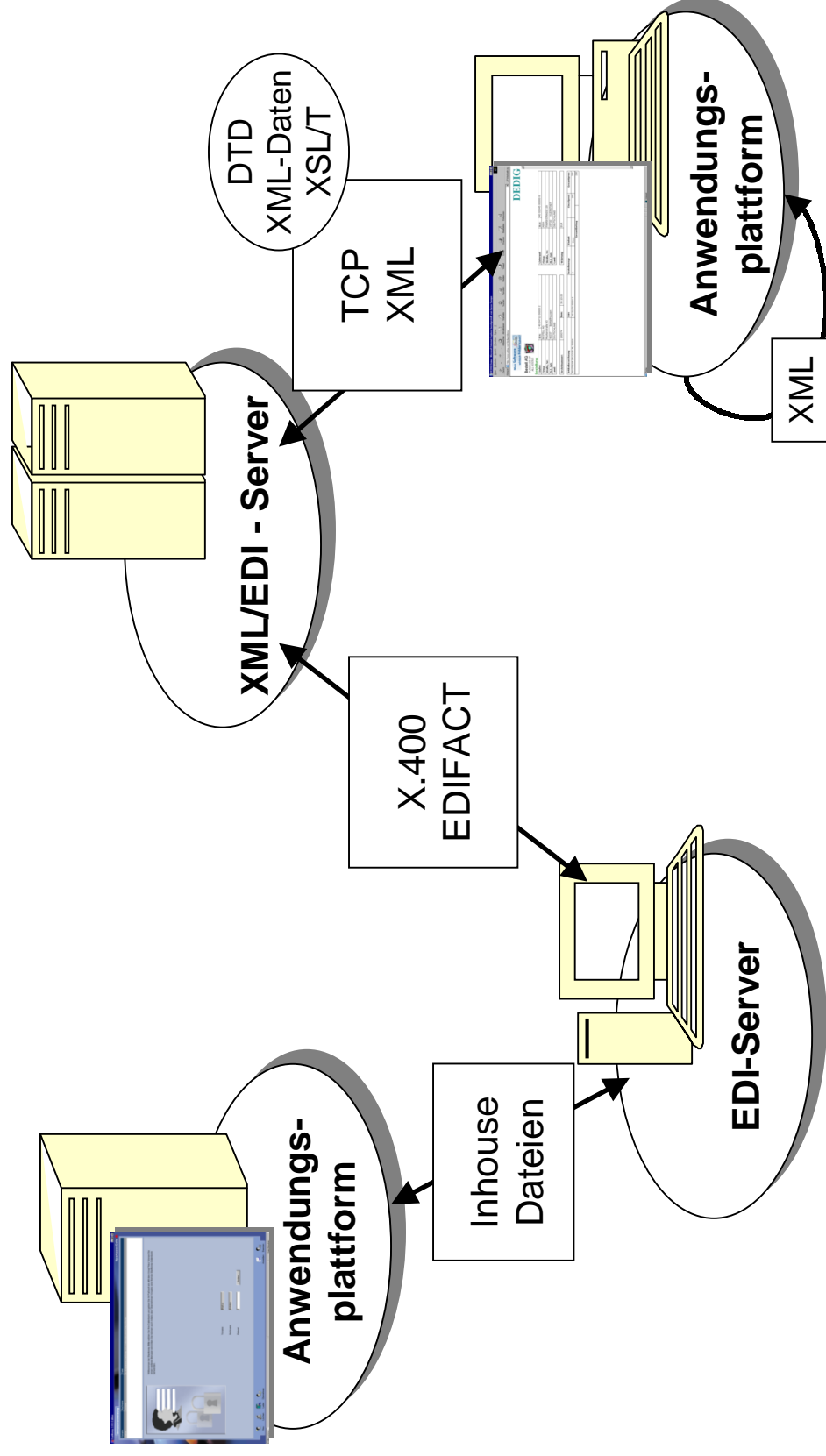
# Änderung der klassischen Systemstruktur II



# Änderung der klassischen Systemstruktur II



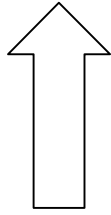
# Änderung der klassischen Systemstruktur II



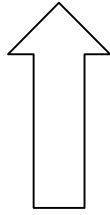


# Probleme

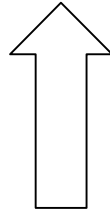
- **Metadaten – Nutzdatenverhältnis :**  
**6 : 1 – 10 : 1**



**Schlechteres ‚Preis/Leistungs‘-Verhältnis**



**Höhere Anforderungen an Kommunikationsschicht**



**Längere Antwortzeiten**

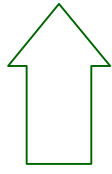
# Enterprise Application Integration

*Definition EAI :*

Enterprise Application Integration (EAI) kombiniert Technologien und Prozesse, die es ermöglichen, dass unterschiedliche, unabhängig voneinander entwickelte Anwendungen eines oder verschiedener Unternehmen Geschäftsvorfälle in Daten und Kontexten austauschen können, die sie alle verstehen können.

# EAI - Eigenschaften

- EAI ist keine ‚Instant‘-Lösung
- EAI ist nicht statisch, sondern dynamisch an ändernde Anforderungen anpassbar
- EAI bildet Geschäftsprozesse nach, indem es beliebige Kombination von Geschäftsapplikationen zulässt



EAI bildet den Klebstoff der die unterschiedlichsten Protokolle, Software und Hardware miteinander verbindet, mit Hilfe dieser Bausteine Geschäftsprozesse nachbildet und automatisiert

# Ebenen der Anwendungsintegration

- Anwendungsintegration auf Datenebene
  - Reiner Austausch von Daten
- Anwendungsintegration auf Objektebene
  - Abbildung von Daten auf Gegenstände von Geschäftsvorfällen
  - ‚Einbettung‘ dieser Objekte in Messages (z.B.: CORBA, XML-RPC, SOAP)
  - Auch: ‚Message-level-EAI‘
- Anwendungsintegration auf Prozessebene
  - Integration von Anwendungen in einen ‚Prozessflow‘

# XML im Bereich EAI

- Einfaches Beispiel -

Kunde	
ID	= 1
Vorname	= 'Hermann'
Nachname	= 'Müller'
Adresse	= 'Dawnstreet 4'
Telefon	= '064236626'

Umwandlung in XML durch Konnektor

```
<KUNDE>  
<ID>1</ID>  
<VORNAME>Hermann</VORNAME>  
<NACHNAME>Müller</NACHNAME>  
<ADRESSE>Dawnstreet</ADRESSE>  
<TELEFON>064236626</TELEFON>  
</KUNDE>
```

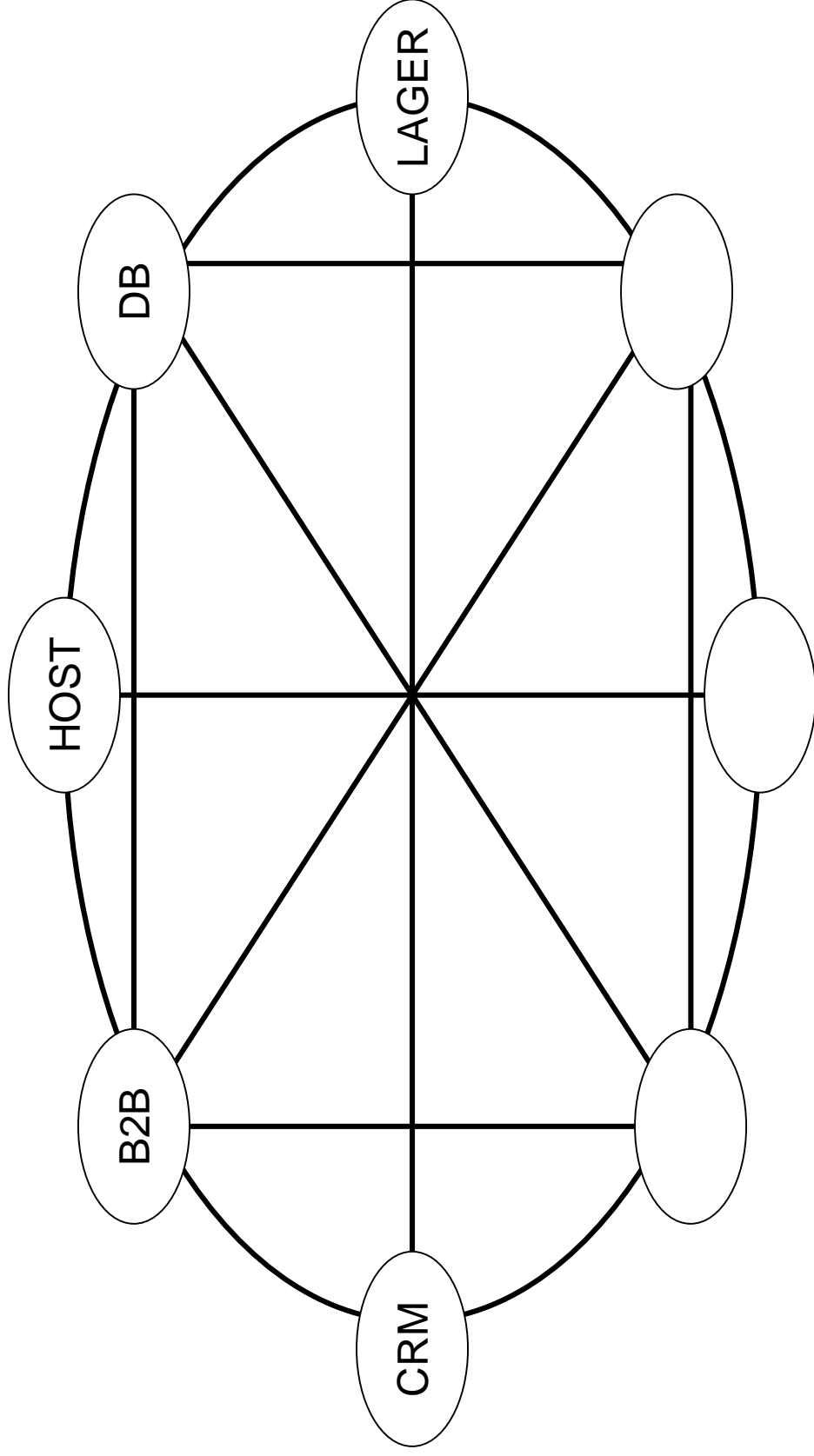
# XML im Bereich EAI

- Einfaches Beispiel -

```
<?xml version="1.0"?>
<INVOICE>
  <SHIPTO>
    <KUNDE>
      <ID>1</ID>
      <VORNAME>Hermann</VORNAME>
      <NAME>Müller</NAME>
      <ADRESSE>Dawnstreet 4</ADRESSE>
      <TELEFON>064236626</TELEFON>
    </KUNDE>
  </SHIPTO>
  <BILLTO>
    <KUNDE>
      <ID>1</ID>
      <VORNAME>Hermann</VORNAME>
      <NAME>Müller</NAME>
      <ADRESSE>Dawnstreet 4</ADRESSE>
      <TELEFON>064236626</TELEFON>
    </KUNDE>
  </BILLTO>
  <TOTAL currency="€">200.50</TOTAL>
  <LINE_ITEMS>
    <LINE>
      <BESCHREIBUNG>NVIDIA Geforce FX</BESCHREIBUNG>
      <ARTNR>124345</ARTNR>
    </LINE>
  </LINE_ITEMS>
</INVOICE>
```

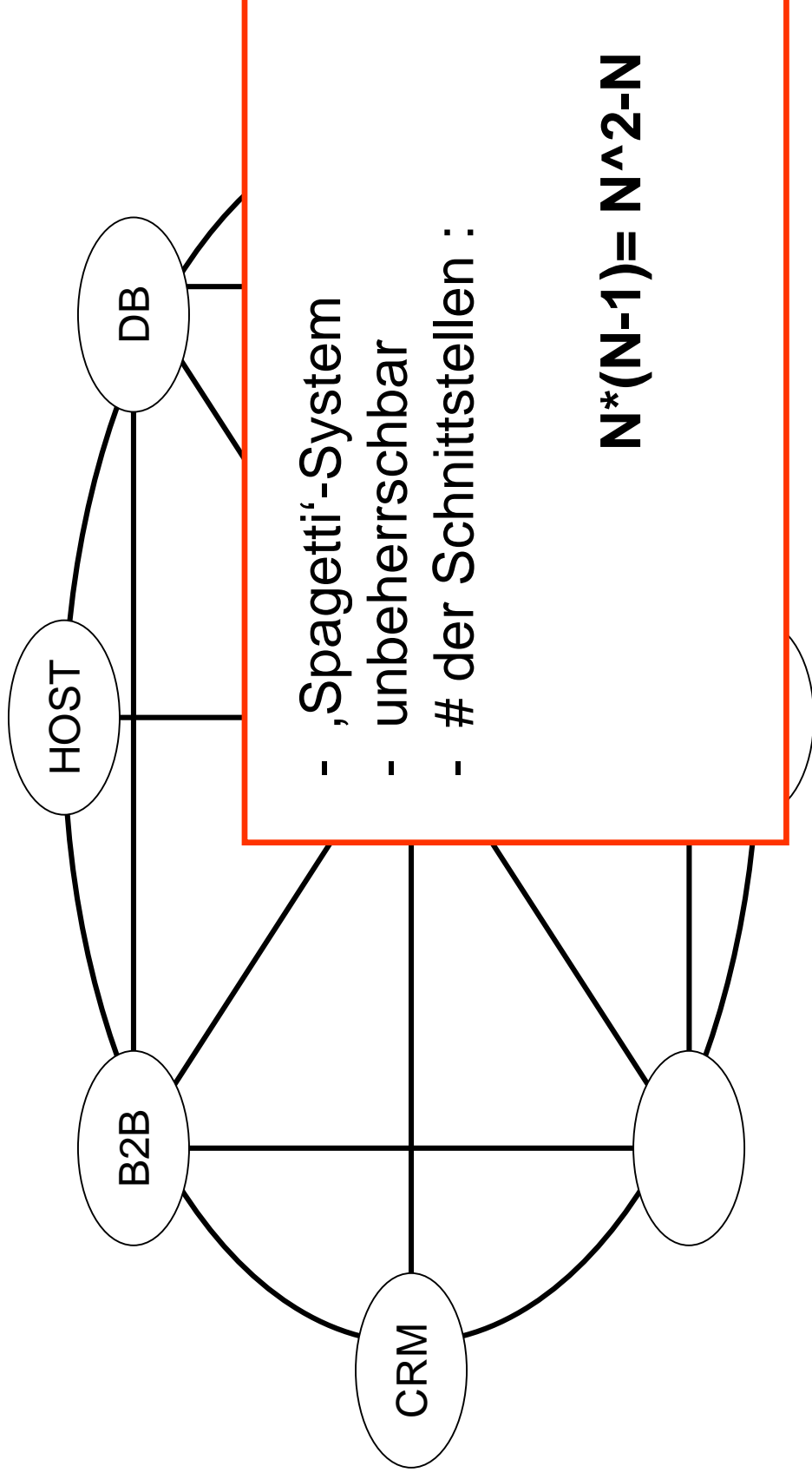
# EAI – System Ansätze I

## Point-to-Point Verbindungen



# EAI – System Ansätze I

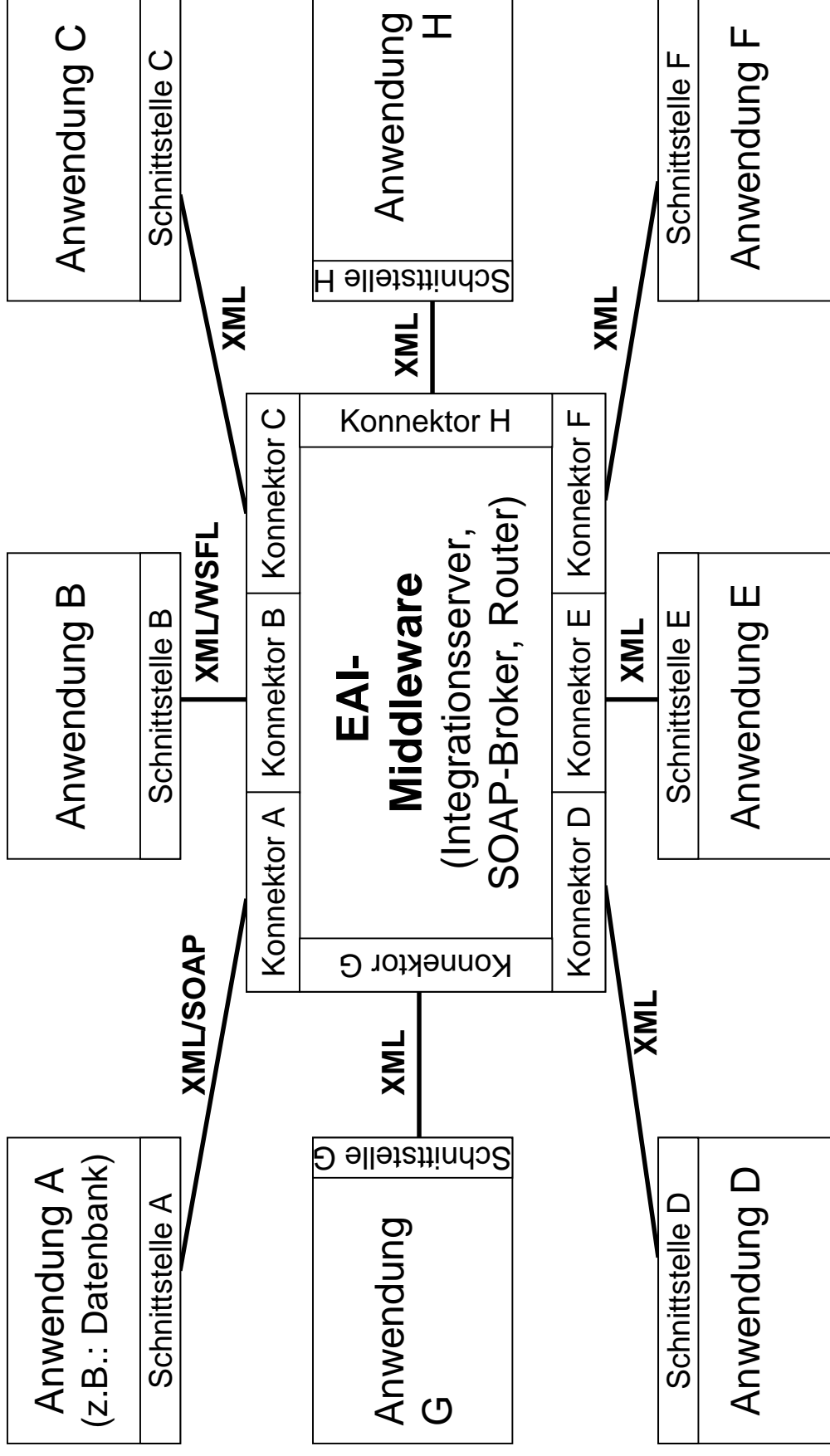
## Point-to-Point Verbindungen





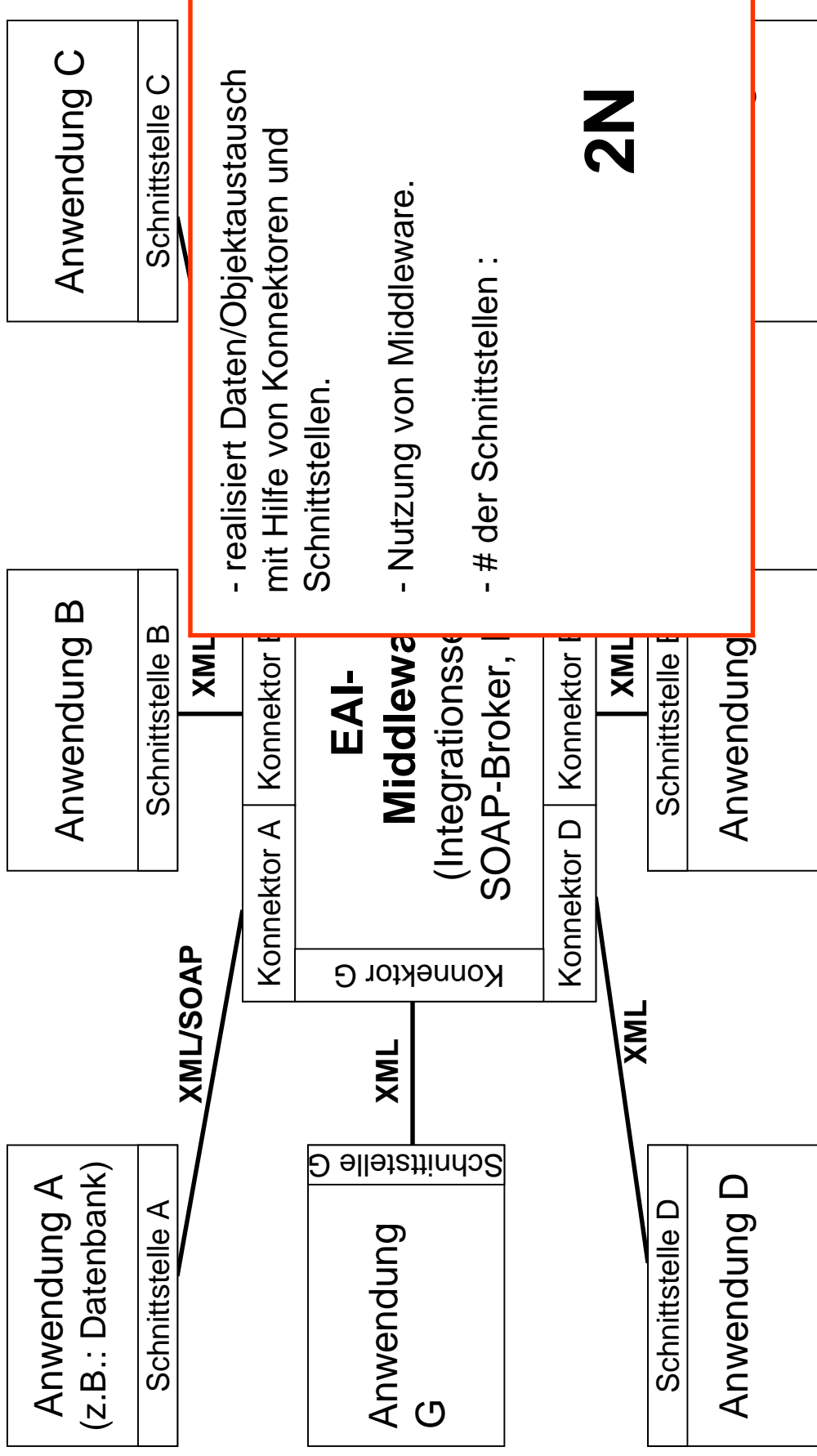
# EAI-Systemansätze II

## HUB-Spoke



# EAI-Systemansätze II

## HUB-Spoke



# mqSeries I

- Integrationsplattform der Firma IBM
- Eng gekoppelt mit Websphere
  - » Websphere MQ
- Nutzt Message-Queue Prinzip (unidirektional)
  - ACID
  - Grosse Flexibilität im Systemdesign (one – to – one, one- to many, many- to –many)
- Realisiert 2 Hauptfunktionalitäten
  - Dequeue- Enqueue - Funktionalität
  - Publish – Subscriber - Funktionalität

# mqSeries II

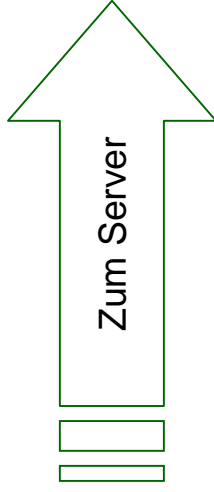
- 5 Teilkomponenten:
  - **Websphere MQ**
    - Realisiert Standard MQ-Funktionalität (De-Envelope)
  - **Websphere Integrator Broker**
    - Publish-Subscriberfunktionalität
  - **Websphere Adapters**
    - Bereitstellung von Konnektoren
  - **Websphere MQ Everywhere**
    - Anbindung ‚mobiler‘ externer Anwendungen
  - **MqSeries Workflow**
    - ‚Business-Management-Flow‘-System

# XML-RPC

- Remote-Procedure-Call-Mechanismus per XML
- Realisiert durch HTTP-POST-Requests
- Definiert Datengrundtypen
  - » `<value>`
  - » `<i4>`, `<int>`, `<boolean>`, `<string>`, `<date>`
- Sowie komplexe Datentypen
  - » `<struct>`, `<array>`
- Spezifiziert Fehlermeldungen
  - » `<fault>`

# XML – RPC

(Beispiel)



```
POST /RPC2 HTTP/1.0
User Agent: frontier/5.1.2 (WiNT)
Host: betty.userland.com
Content-Type: text/html
Content-length: 181

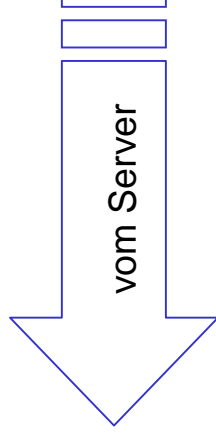
<?xml version="1.0"?>
<methodCall>
  <methodName>examples.gtStateName</methodName>
  <params>
    <param>
      <value><i4>41</i4></value>
    </param>
  </params>
</methodCall>
```

# XML – RPC

(Beispiel)

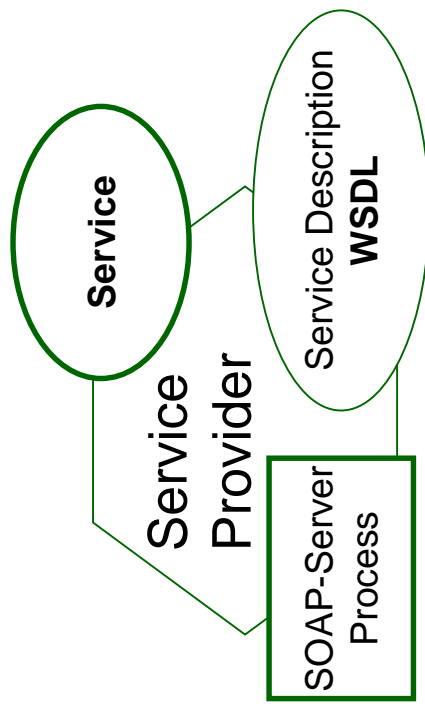
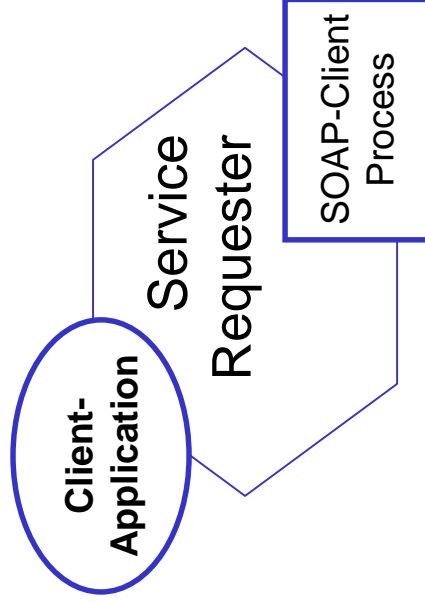
```
HTTP/1.1 200 OK
Connection: close
Content-Length: 158
Content-Type: text/html
Date: fri, 17 Jul 1998 19:55:08 GMT
Server: Userland Frontier/5.1.2-WinNT

<?xml version="1.0"?>
<methodResponse>
  <params>
    <param>
      <value><string>South Dakota</string></value>
    </param>
  </params>
</methodResponse>
```



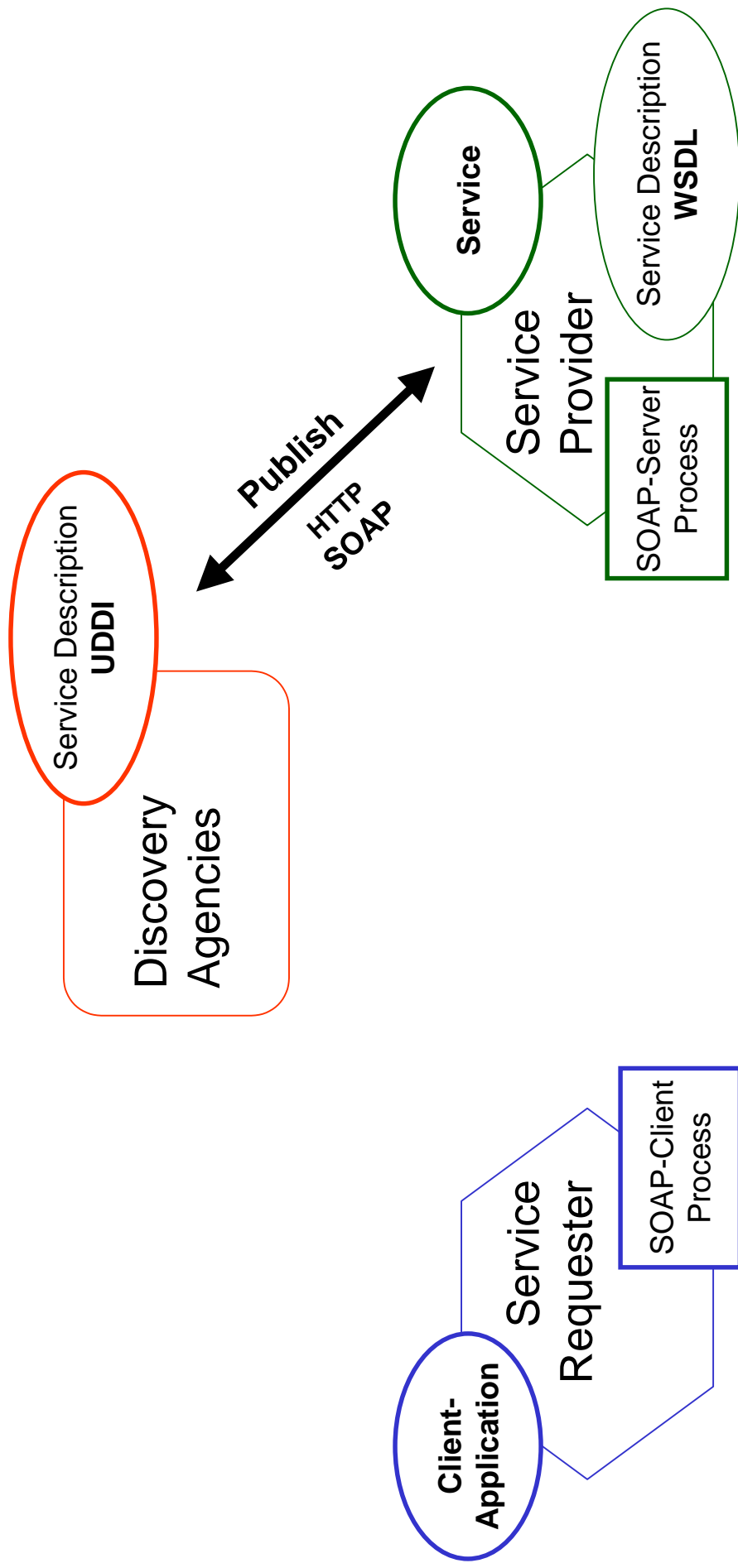
# Web Services

Discovery Agencies

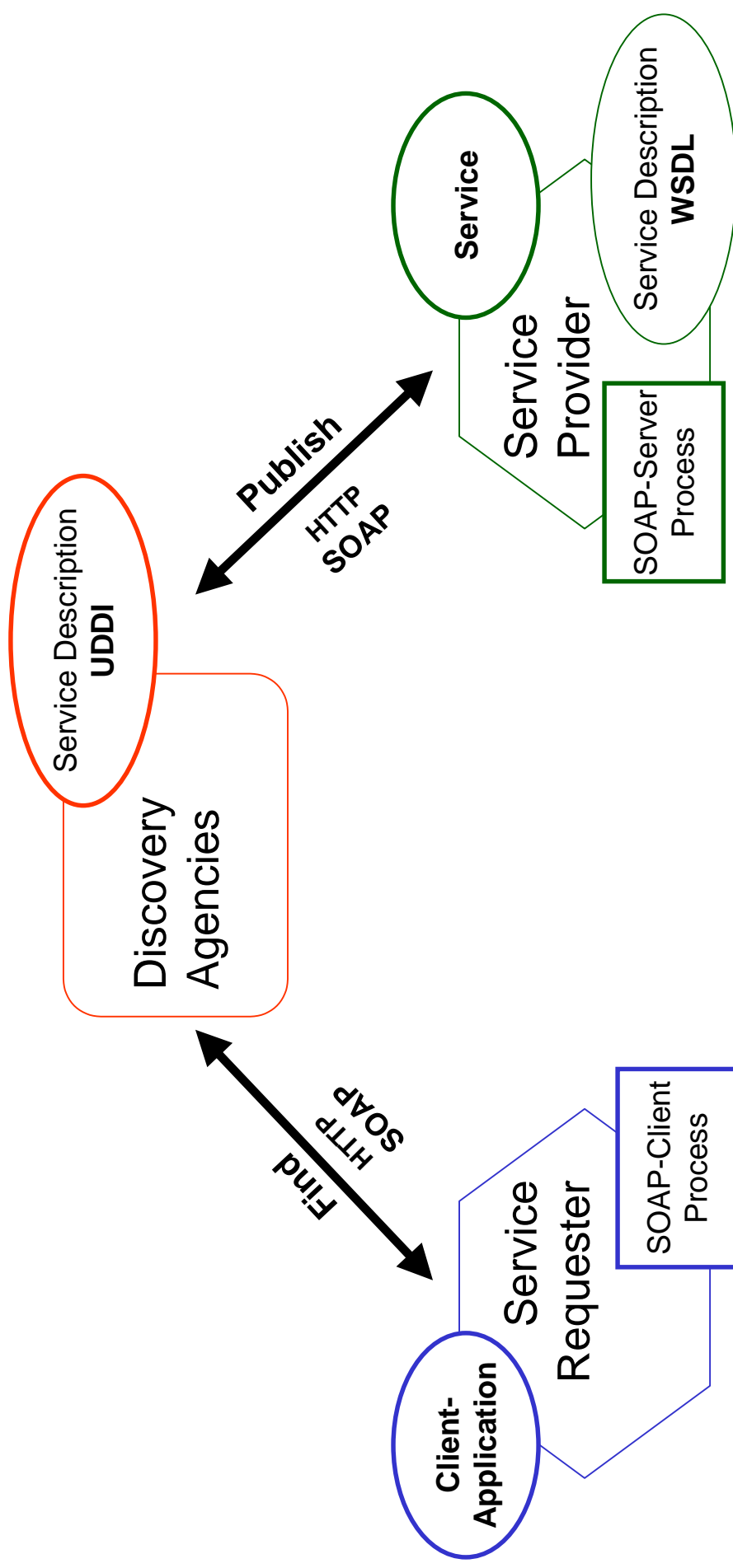




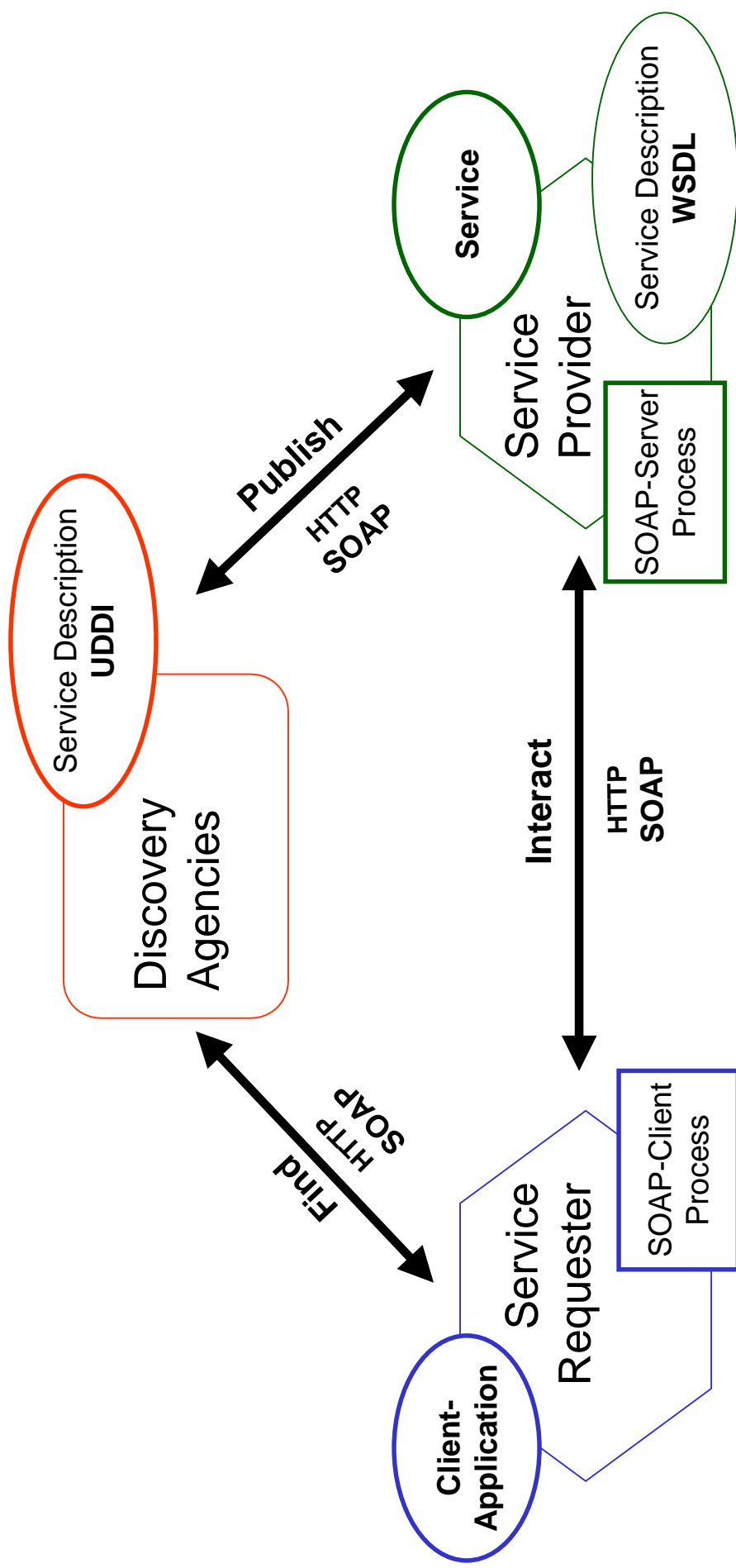
# Web Services



# Web Services



# Web Services



# UDDI

- XML Dokument
  - Wird verwaltet von OASIS (Organisation for the Advancement of Structured Information Standards)
  - Enthält Informationen über Anbieter des Webservices
  - Kategorisierung von Unternehmen
  - Enthält technische Details über Service
  - Definiert Accesspoints, an denen WSDL-Beschreibung zu finden ist.

**Beispiel:**

```
<accessPoint URLType="https">https://uddi.microsoft.com/publish</accessPoint>
```

```
<serviceDetail generic="1.0" operator="Microsoft Corporation"
  truncated="false" xmlns="urn:uddi-org:api">
  <businessService businessKey="0076B468-EB27-42E5-AC09-9955CFF462A3"
    serviceKey="D2BC296A-723B-4C45-9ED4-494F9E53F1D1">
    <name>UDDI Web Services</name>
    <description xml:lang="en">UDDI SOAP/XML message-based programmatic web
  service interfaces.</description>
  <bindingTemplates>
    <bindingTemplate bindingKey="313C2BF0-021D-405C-8149-25FD969F7F0B"
      serviceKey="D2BC296A-723B-4C45-9ED4-494F9E53F1D1">
      <description xml:lang="en">Production UDDI server,
  Publishing interface</description>
    <accessPoint URLType="https">https://uddi.microsoft.com/publish</accessPoint>
  </bindingTemplate>
  </tModelInstanceDetails>
  <tModelInstanceInfo tModelKey="uuid:64C756D1-3374-4E00-AE83-EE12E38FAE63">
    <description xml:lang="en">UDDI SOAP Publication Interface</description>
  </tModelInstanceInfo>
</bindingTemplate>
.....
</serviceDetail>
```

## UDDI – XML - Beispiel

# WSDL

- Ist die maßgebende Beschreibung des Dienstes
- Spezifiziert Datenaustauschformate
  - » Durch Element `<types>`
- Spezifiziert verfügbare Nachrichtentypen

```
<message name="add_publisherAssertions">  
  <part name="body" element="uddi:add_publisherAssertions" />  
</message>
```

- Spezifiziert Funktion dieser Nachrichtentypen
  - Definition von Empfangs und Sendeoperationen, Fehlernachrichten

```
<portType name="Publish">  
  <operation name="add_publisherAssertions">  
    <input message="tns:add_publisherAssertions" />  
    <output message="tns:dispositionReport" />  
    <fault name="error" message="tns:dispositionReport" />  
  </operation>
```

# WSDL II

- Spezifiziert Bindung der Nachrichten zu einem Protokoll z.B.: SOAP , MIME, etc.

```
<binding name="PublishSoap" type="tns:Publish" >
  <soap:binding style="document" transport="http://schemas.xmlsoap.org/soap/http" />
  <operation name="add_publisherAssertions" >
    <soap:operation soapAction="add_publisherAssertions" style="document" />
    <input message="tns:add_publisherAssertions" >
      <soap:body use="literal" parts="body" namespace="urn:uddi-org:api_v2" />
    </input>
  </operation>
```

# SOAP

Simple Object Access Protokoll

```
POST /StockQuote HTTP/1.1
Host: www.stockquotesever.com
Content-Type: text/xml; charset="utf-8"
Content-Length: nnnn
SOAPAction: "Some-URI"

<SOAP-ENV:Envelope
  xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
  SOAP-ENV:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/" />
  <SOAP-ENV:Header>
    <t:Transaction xmlns:t="some-URI" SOAP-ENV:mustUnderstand="1" >
      5
    </t:Transaction>
  </SOAP-ENV:Header>
  <SOAP-ENV:Body>
    <m:GetLastTradePrice xmlns:m="Some-URI" >
      <symbol>DEF</symbol>
    </m:GetLastTradePrice>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

Anfrage



# SOAP

Simple Object Access Protokoll

Anfrage	Antwort
<pre>POST /StockQuote HTTP/1.1 Host: www.stockquotesever.com Content-Type: text/xml; charset="utf-8" Content-Length: 100 SOAPAction: http://schemas.xmlsoap.org/soap/envelope/</pre>	<pre>HTTP/1.1 200 OK Content-Type: text/xml; charset="utf-8" Content-Length: nnnn &lt;SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/" SOAP-ENV:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"&gt; &lt;SOAP-ENV:Body&gt; &lt;m:GetLastTradePriceResponse xmlns:m="Some-URI"&gt; &lt;Price&gt;34.5&lt;/Price&gt; &lt;/m:GetLastTradePriceResponse&gt; &lt;/SOAP-ENV:Body&gt; &lt;/SOAP-ENV:Envelope&gt;</pre>

# Abschließende Bemerkungen

- **EDI**
  - Probleme/Lösungen
  - XML
- **EAI**
  - Rolle von XML
  - EAI ist ein dynamischer Gedanke
  - Unterschiedliche Hersteller mit unterschiedlichen Produkten
    - z.B.: Websphere MQ Family von IBM
    - z.B.: Biztalk von Microsoft
- **WebServices**
  - Rolle von XML

**ENDE**