DaimlerChrysler

Die besondere Rolle von XML im EAI

Mario Jeckle
DaimlerChrysler Forschungszentrum Ulm
mario.jeckle@daimlerchrysler.com
mario@jeckle.de
www.jeckle.de

Gliederung

I Enterprise Application Integration

- Begriffsbildung

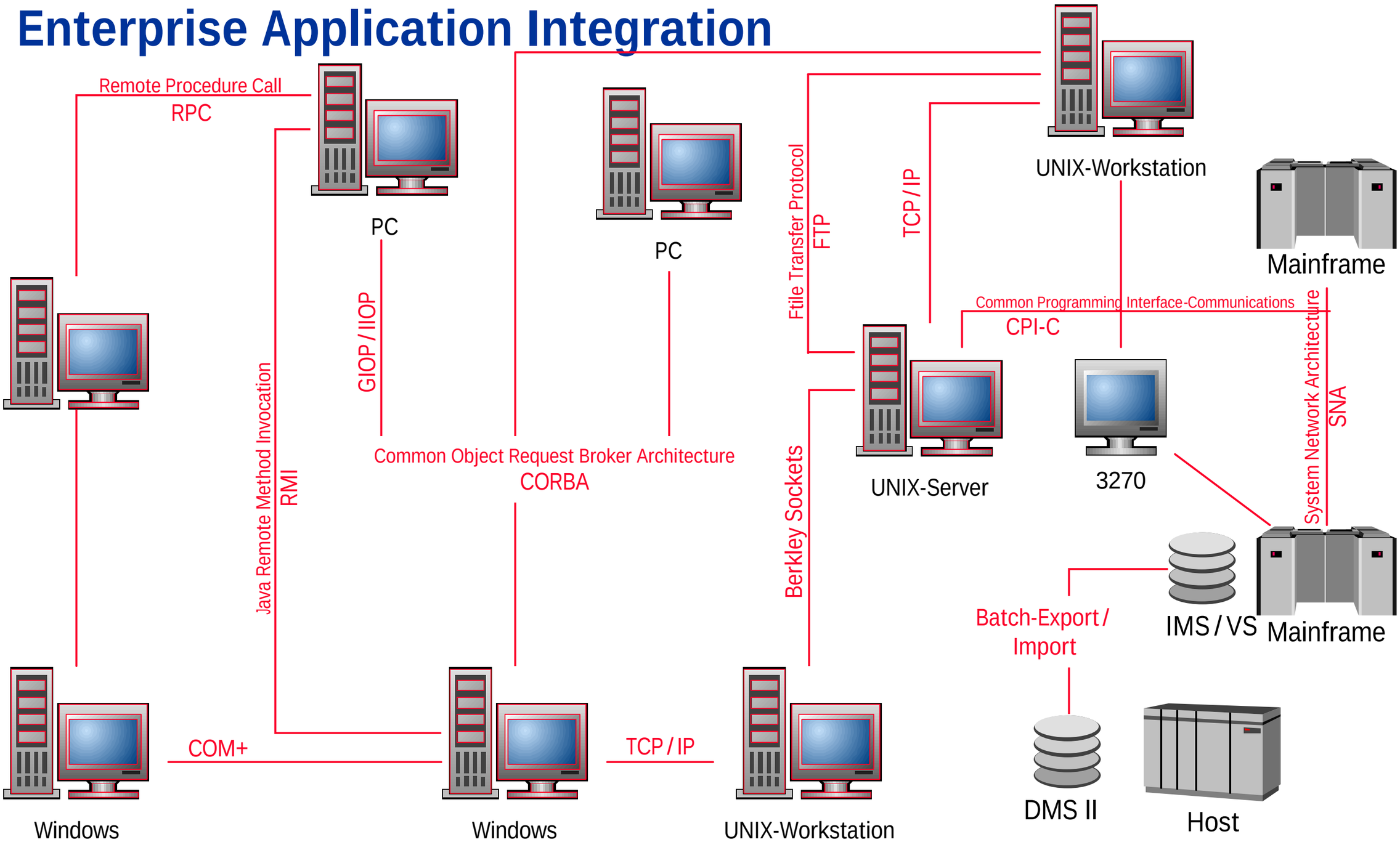
II Herausforderungen und verfügbare Lösungen

- Datenintegration
- Funktionsintegration
- Prozeßintegration

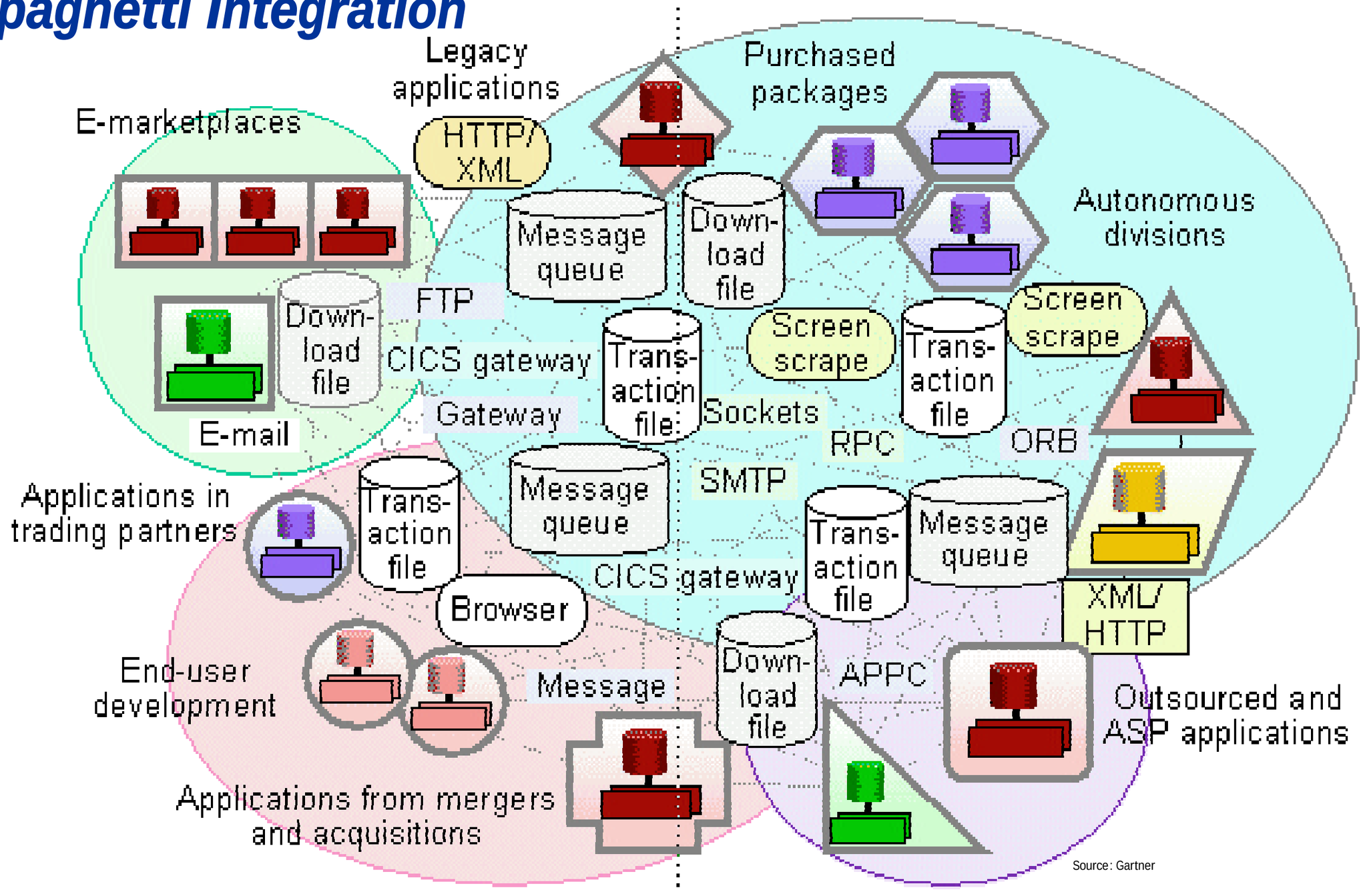
III XML im EAI

- XML-Sprachen, Standards und eigene Datenschemata
- Entfernte Funktionsaufrufe, Kommunikation und Integration
- Prozesse, Transaktionen und verteilte Abläufe

Enterprise Application Integration



Spaghetti Integration



Source: Gartner

Enterprise Application Integration vs. Middleware

Middleware-Eigenschaften

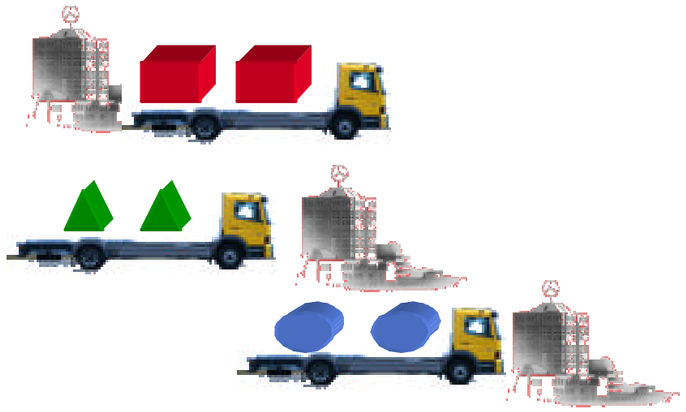
- Applikationsgebunden
  Änderungen (Customizing) an Applikation notwendig
- Technologiegebunden
  (Programmiermodell-, Datenhaltungsparadigmenspezifisch)
- Systemzentriert
  (Hauptfocus Kopplung von technischen Systemen)
- Statische Umsetzung
  (keine oder nur begrenzte dynamische Rekonfigurierbarkeit und Wiederverwendung)

Enterprise Application Integration vs. Middleware

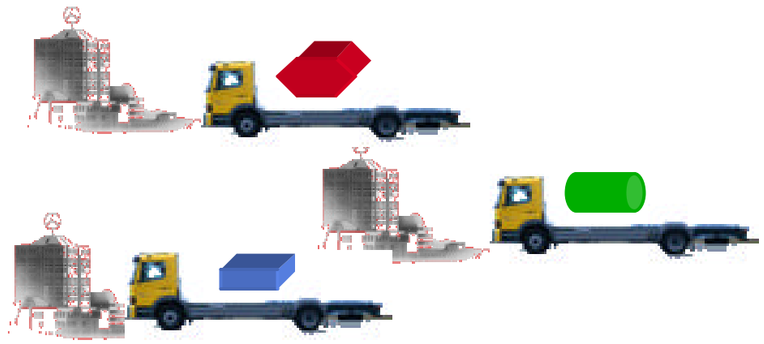
EAI-Eigenschaften

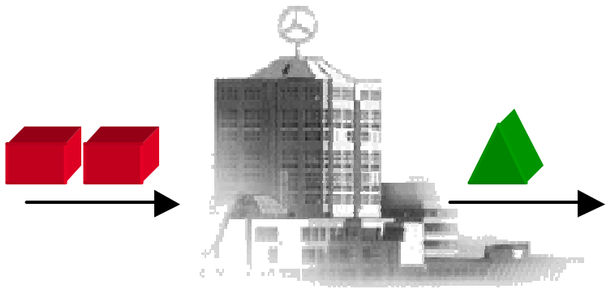
- Anwendungszentriert nicht Applikationsabhängig
  (Prozeß- nicht Systemgetrieben)
- Prozeßorientiert
  (Kopplung von Prozessen, nicht Systemen)
- Dynamische (Re-)Konfigurierbarkeit
  („Integration“ ist eigener Integrationsbaustein)
- Flexible Einsetzbarkeit und Wiederverwendbarkeit
  der verwendeten Techniken und Lösungen

Herausforderungen und verfügbare Lösungen

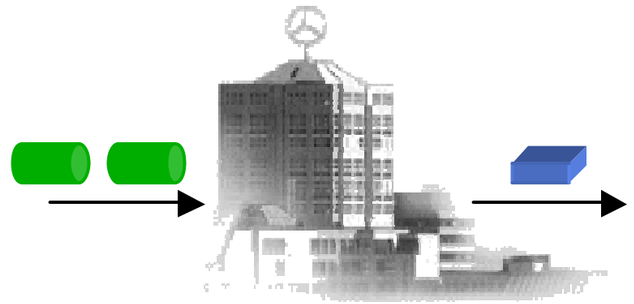


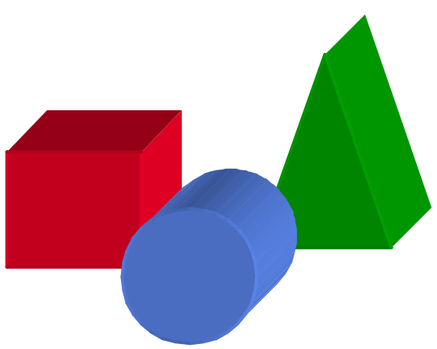
Prozeßintegration



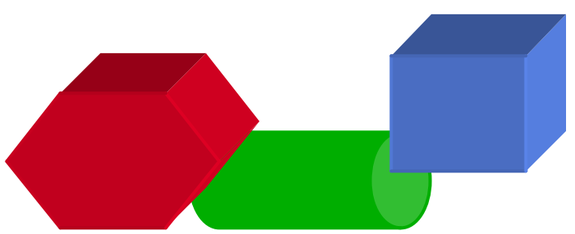


Funktionsintegration





Datenintegration



Herausforderungen und verfügbare Lösungen

Datenintegration

Zielsetzung:
Speicherformunabhängige Informationsdarstellung

Lösungsalternativen:
- Reduktion auf ein einziges Format (*single system strategy*)
- Einführung eines neutralen Austauschformates

(bekannte) verfügbare Umsetzungen:
- Standard for the Exchange of Product Model Data (STEP)
- SWIFT-Format der Society for Interbank Financial Telecommunications
- Electronic Data Interchange (EDI)
- Health Level Seven (HL7)
- Föderierte Datenbanken

Herausforderungen und verfügbare Lösungen

Funktionsintegration

Zielsetzung:
Implementierungsunabhängige Nutzung von Funktionalitäten

Lösungsformen:
- Remote Procedure Calls über System und Plattformgrenzen hinweg
- Middleware-Architekturen

(bekannte und verbreitete) verfügbare Umsetzungen:
- Distributed Computing Environment (DCE) (Open Software Foundation (OSF))
- Common Object Request Broker Architecture (CORBA) (OMG)
- Component Object Model (COM) (Windows, Microsoft)
- Enterprise Java Beans (Java, SUN)

Herausforderungen und verfügbare Lösungen

Prozeßintegration

Zielsetzung:
Nutzung von Funktionalitäten zur Unterstützung von Prozessen

Lösungsformen:
- Message oriented Middleware (MOM)
  - Nachrichten-Beförderung und ggf. Format-Transformation
- Workflow Management Systeme (WfMS)
  - Organisation und Überwachung von Abläufen

(bekannte und verbreitete) verfügbare Umsetzungen:
- ... diverse WfMS verschiedener Hersteller
- ... diverse MOM-Systeme verschiedener Hersteller

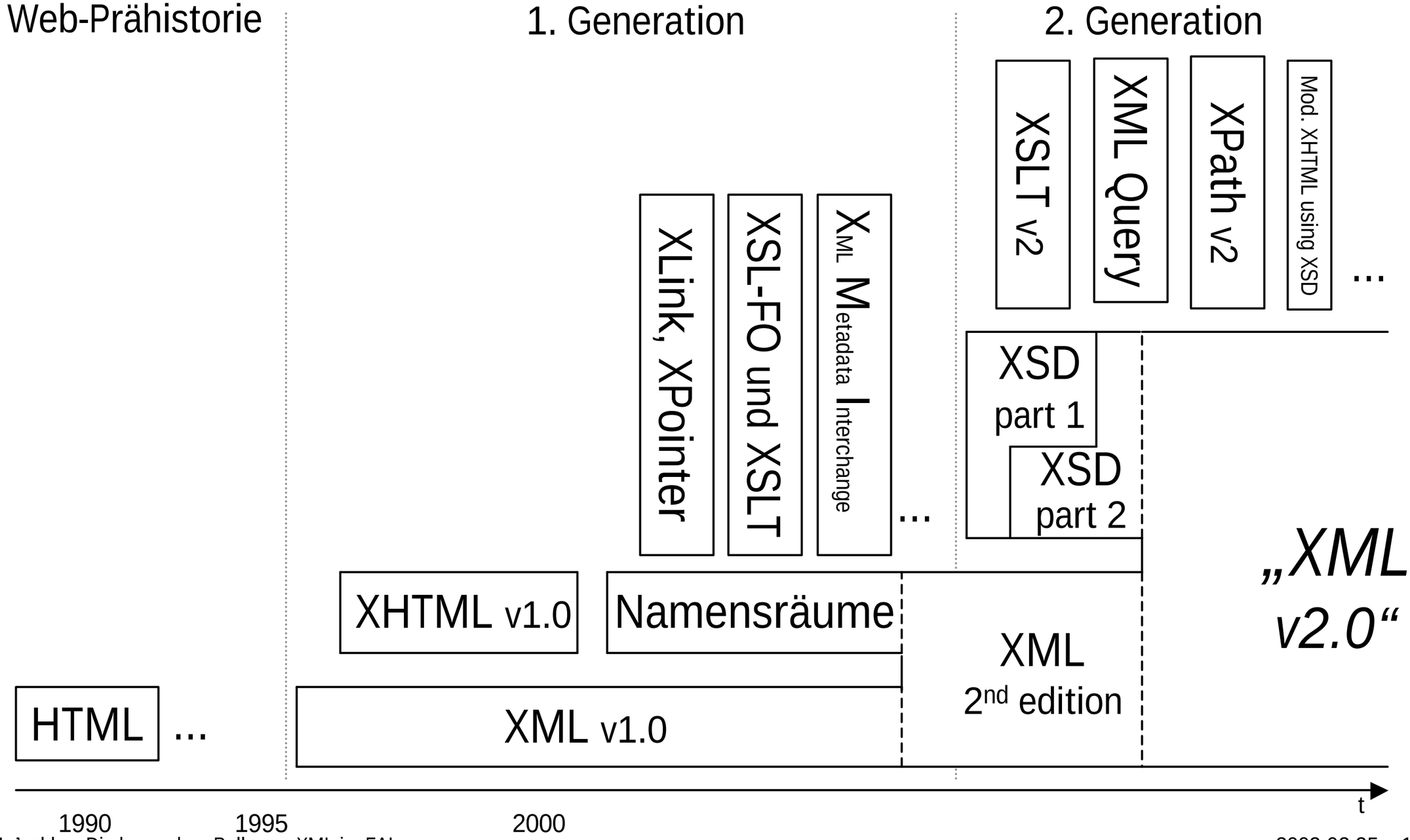
Die Extensible Markup Language (XML)

- XML ist eine Methode, um strukturierte Daten in einer Textdatei darzustellen
- XML ist Text, aber nicht zum Lesen
- XML ist ausführlich, was aber kein Problem darstellt
- XML sieht fast aus wie HTML, ist aber kein HTML
- XML ist eine Familie von Techniken
- XML ist eine Untermenge des ISO-Standards SGML
- XML ist international und unterstützt beliebige Alphabete (Unicode)
- XML ist neu, aber nicht so neu
- XML ist lizenzfrei, plattform- und herstellerunabhängig, und gut unterstützt
- XML ist inzwischen weit verbreitet

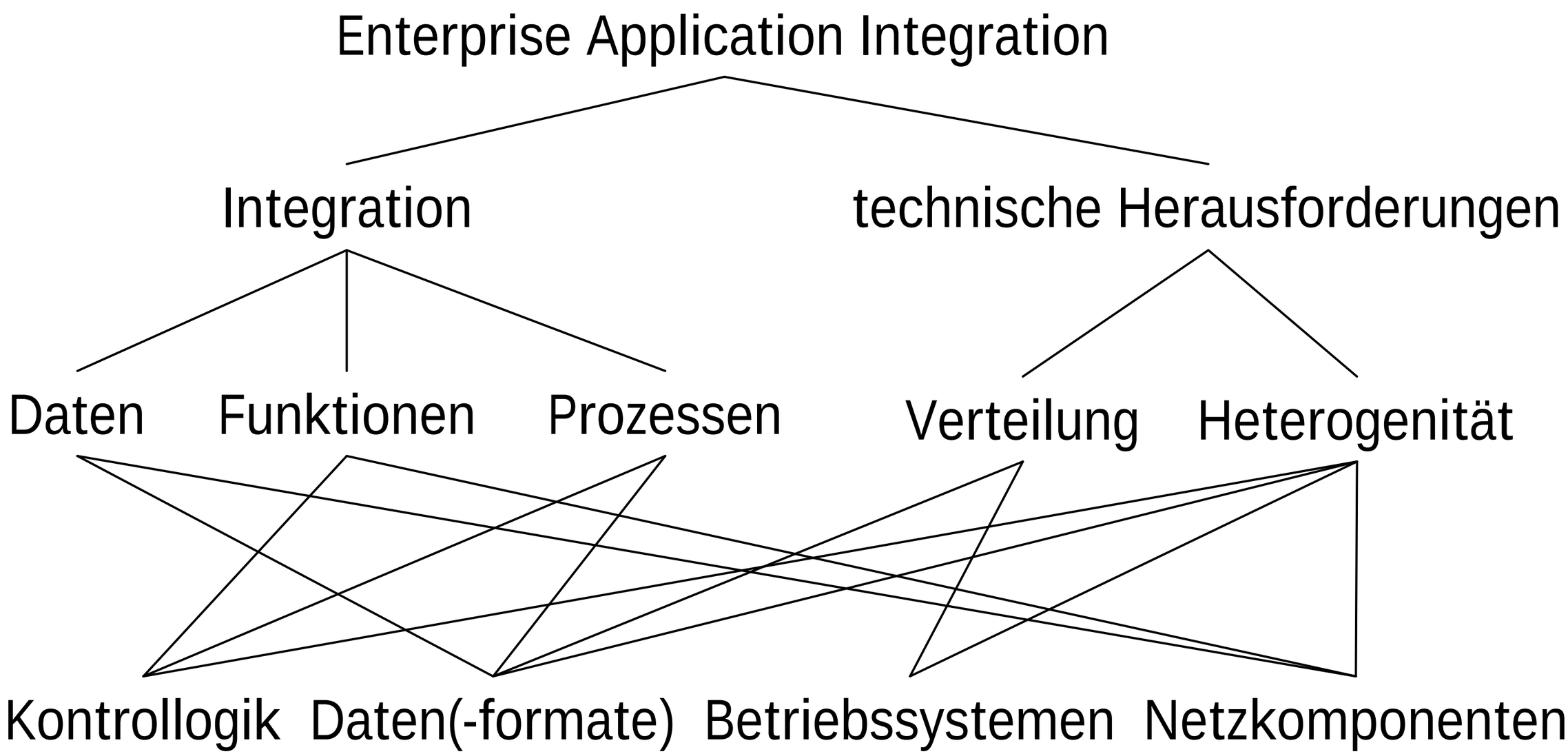
Die Extensible Markup Language (XML)

```
<?xml version="1.0" encoding="UTF-8"?>
<Vortrag>
  <Titel>Die besonderer Rolle von XML im EAI</Titel>
  <Veranstaltung datum="2002-02-25">
    <Name>Euroforum EAI-Kongress</Name>
  </Veranstaltung>
  <Referent>
    <Name>Mario Jeckle</Name>
    <Firma>DaimlerChrysler Research and Technology</Firma>
    <URL>http://www.jeckle.de</URL>
    <E-Mail>mario.jeckle@daimlerchrysler.com</E-Mail>
  </Referent>
</Vortrag>
```

Die Extensible Markup Language (XML)

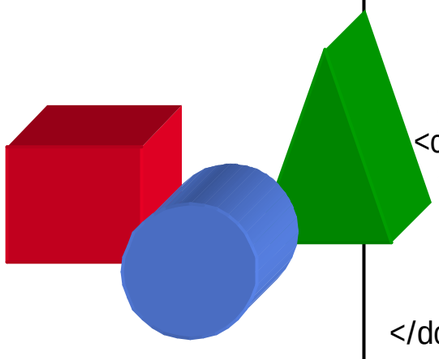


Besondere Bedeutung von XML im EAI



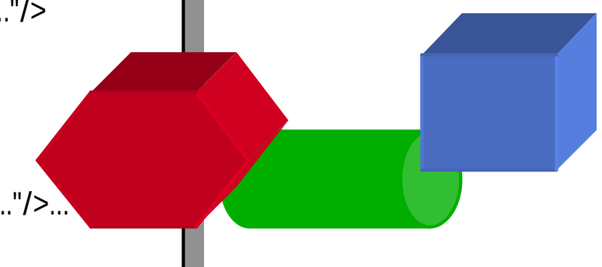
Datenintegration mit XML

- XML erlaubt die Text-basierte Darstellung beliebiger strukturierter Information
- Existierende XML-Formate ermöglichen schnellen Start und Wiederverwendung
- Eigene XML-Sprachen können leicht erstellt werden



```
<?xml version = "1.0" encoding = "UTF-8"?>
<document>
  <invoice issueDate="2001-08-23">
    <customer>
      <id>101</id>
        ...
  </invoice>
  <billOfMaterial>
    <product name="..." quantity="...">
      <product ...
      </product>
    ...
  <catalog>
    <category name="...
      <product id="...">
        <description xml:lang="en">
        ...
</document>
```

```
<?xml version = "1.0" encoding = "UTF-8"?>
<document>
  <invoice>
    <day>23</day>
    <month>August</month>
    <year>2001</year>
    <deliveryTo>
</invoice>
   <BOM>
      <position no="1">
        <record id="42">
          <prdRef qty="..." prd="..."/>
          <prdRef ...
      </position>
<productCat>
  <article>
    <belongsTo><catRef catID="..."/>...
</document>
```



Datenintegration mit XML

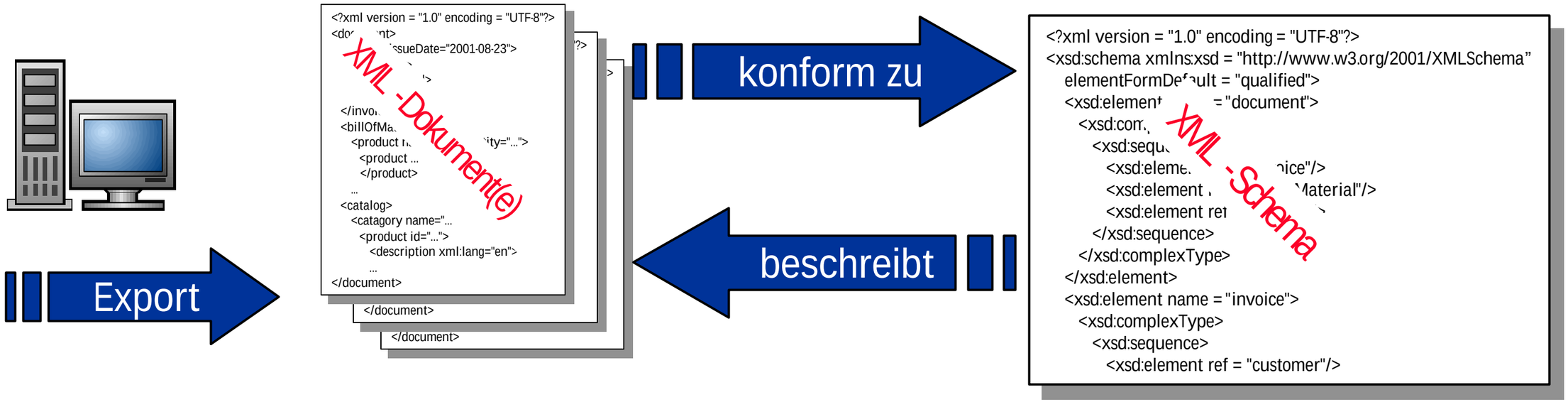
Gemeinsamer Wortschatz und formale Strukturierungsregeln (Syntax)

- ermöglicht Kommunikation und Interaktion auf Basis einheitlich definierter Begriffe
- ermöglicht idealerweise (vergleichsweise) einfache Verarbeitung

(schlanke, eindeutige Definitionen)

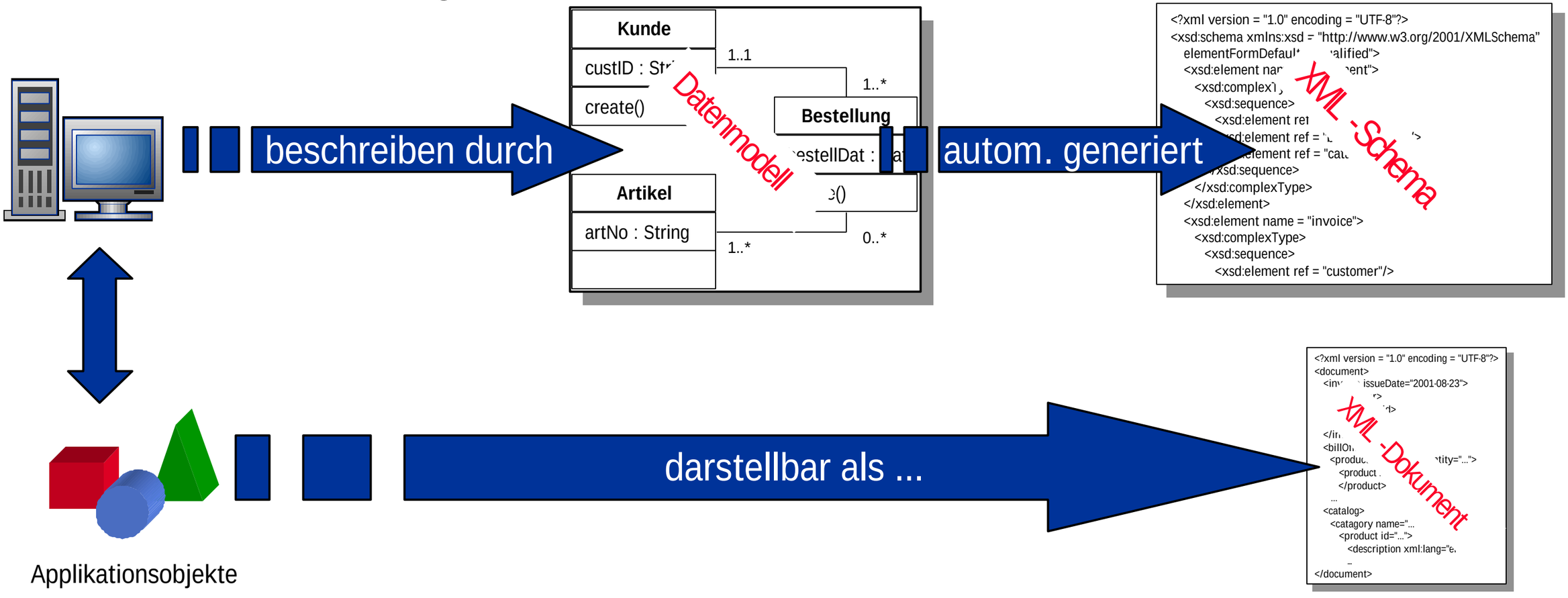
Syntax und Semantik (Vokabular)

- Grundvoraussetzung automatisierter Verarbeitung ohne menschliche Eingriffe



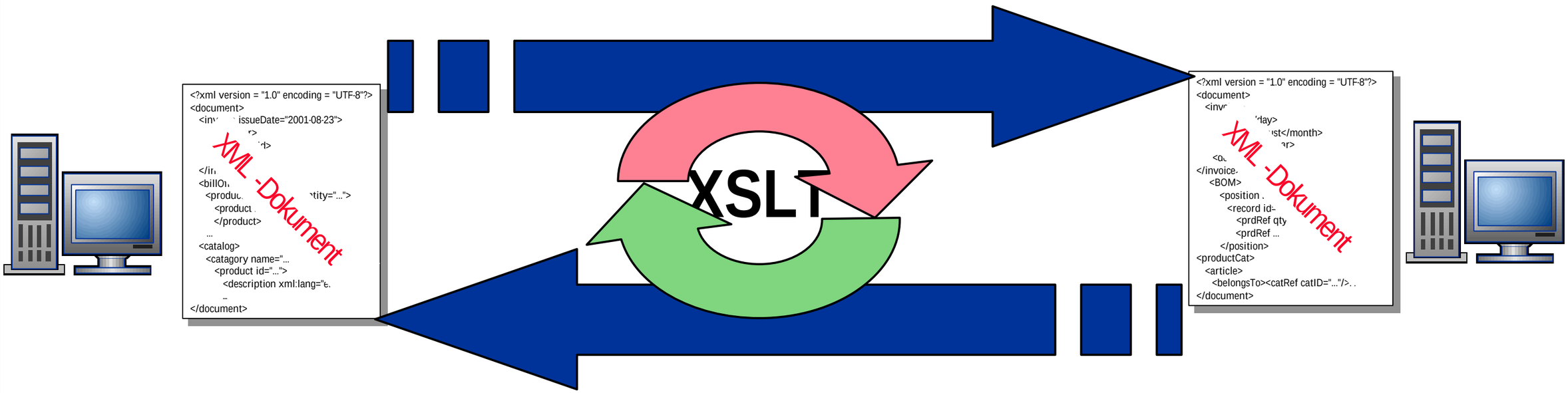
Datenintegration mit XML

- Vokabulare werden durch internationale, nationale und Branchen-spezifische Standardisierungs- und Normierungsgremien erarbeitet (z.B. DIN, VDA, VDMA, ...)
- Vokabulare können aus bestehenden Systembeschreibungen und Datenstrukturen automatisiert erzeugt werden



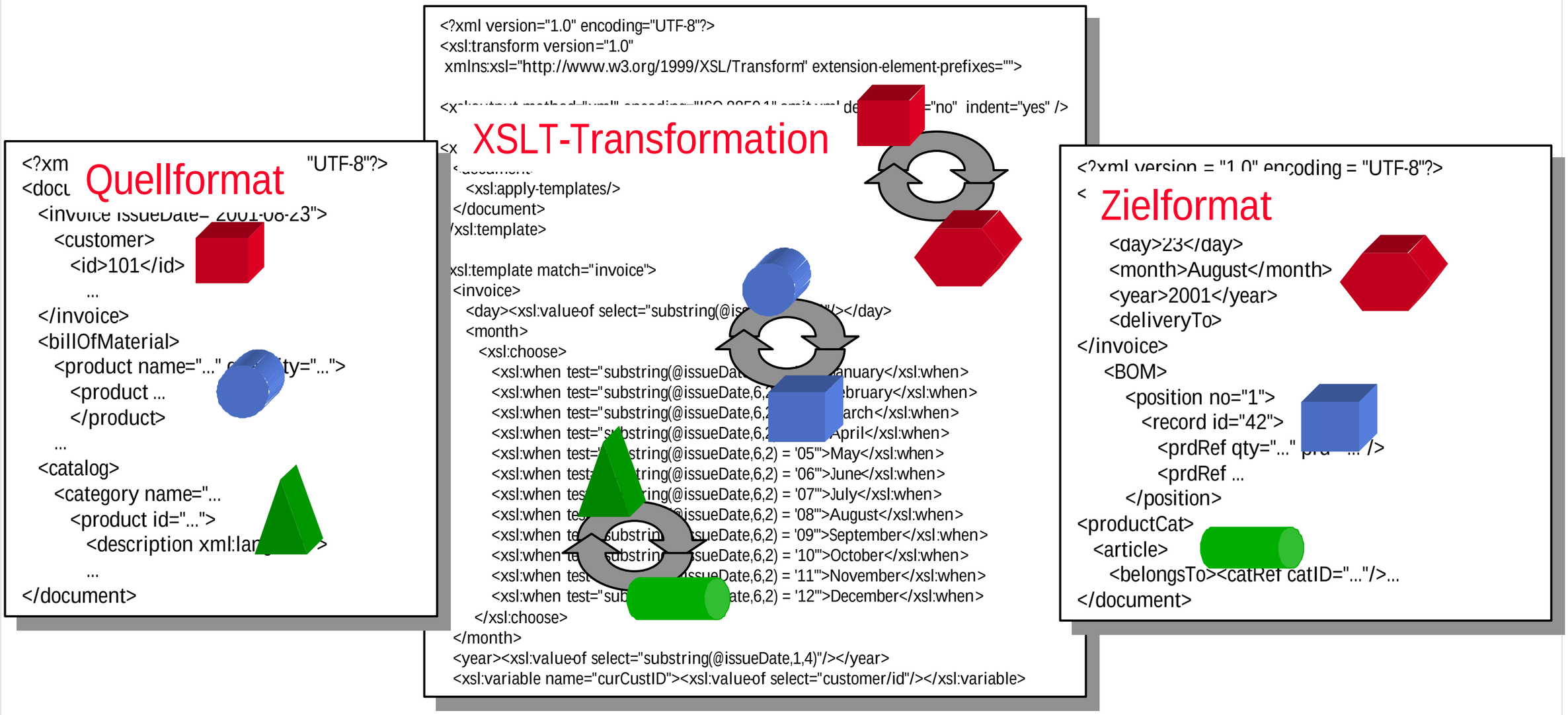
Datenintegration mit XML

- Wandlung zwischen verschiedenen XML-Vokabularen mit der W3C-standardisierten Sprache *XSL Transformations*



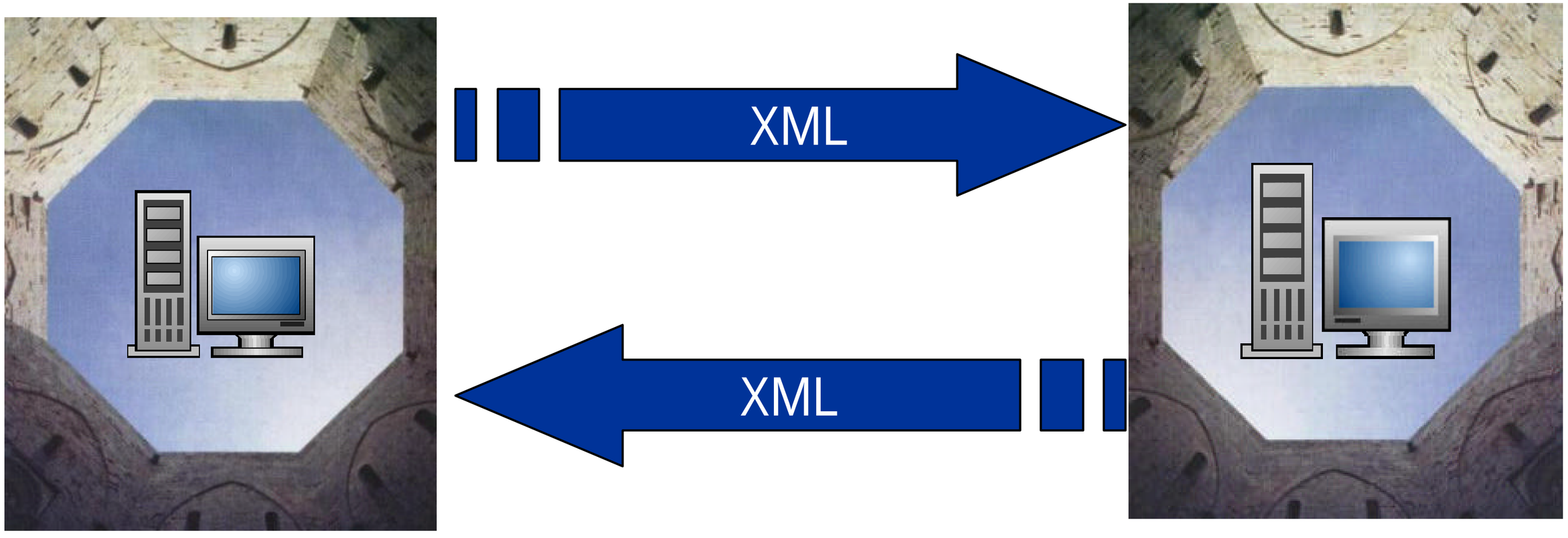
Datenintegration mit XML

- Wandlung zwischen verschiedenen XML-Vokabularen mit der W3C-standardisierten Sprache *XSL Transformations*



Funktionsintegration mit XML

- Datenzugriff oftmals ausschließlich über Systemschnittstelle möglich
- Systeme durch Sicherheitsmaßnahmen (Firewalls) geschützt
- SOAP-Protokoll ermöglicht XML-basierte Funktionsintegration

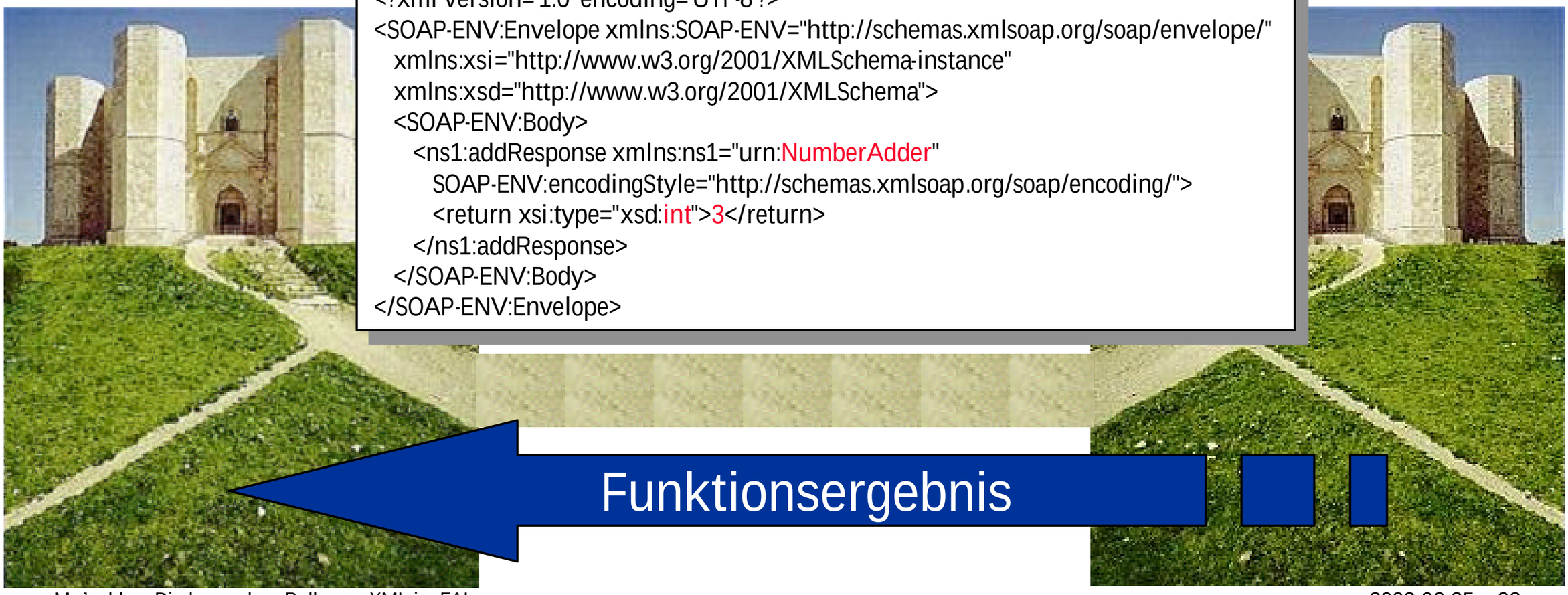


Funktionsintegration mit XML

```
POST /soap/servlet/rpcrouter HTTP/1.0
Host: localhost:8081
Content-Type: text/xml; charset=utf-8
Content-Length: 474
SOAPAction: ""
<?xml version='1.0' encoding='UTF-8'?>
<SOAP-ENV:Envelope
  xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <SOAP-ENV:Body>
    <ns1:add xmlns:ns1="urn:NumberAdder"
      SOAP-ENV:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/">
      <number1 xsi:type="xsd:int">1</number1>
      <number2 xsi:type="xsd:int">2</number2>
    </ns1:add>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

Funktionsintegration mit XML

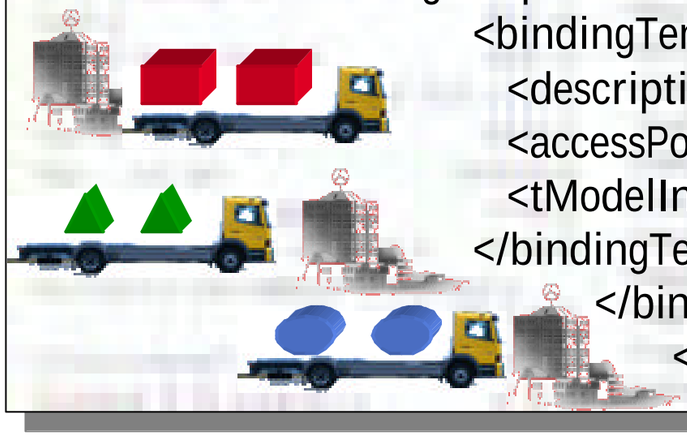
```
HTTP/1.0 200 OK
Content-Type: text/xml; charset=utf-8
Content-Length: 447
Set-Cookie2: JSESSIONID=9ekj42d5b1;Version=1;Discard;Path="/soap"
Set-Cookie: JSESSIONID=9ekj42d5b1;Path=/soap
Servlet-Engine: Tomcat Web Server/3.2.1 (JSP 1.1; Servlet 2.2; Java 1.3.0;
Windows NT 4.0 x86; java.vendor=Sun Microsystems Inc.)

<?xml version='1.0' encoding='UTF-8'?>
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
   xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
   xmlns:xsd="http://www.w3.org/2001/XMLSchema">
   <SOAP-ENV:Body>
      <ns1:addResponse xmlns:ns1="urn:NumberAdder"
         SOAP-ENV:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/">
         <return xsi:type="xsd:int">3</return>
      </ns1:addResponse>
   </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```



Prozeßintegration mit XML

- Publikation extern angebotener Funktionalität durch Dienstverzeichnisse
- Ermittlung geeigneter Dienste aus öffentlich zugänglichen Verzeichnissen
- Dynamisch rekonfigurierbare Kombination verfügbarer Dienste

```
<?xml version = "1.0" encoding = "UTF-8"?>
<businessService xmlns = "urn:uddi-org:api_v2"
xmlns:xsi = "http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation = "urn:uddi-org:api_v2 uddi_v2.xsd"
businessKey = "ae44a975-27ab-4da8-a45a-0e895f588c2f"
serviceKey = "fec47752-2b7e-4541-88a2-770f70b64d7d">
        <name>urn:DCXConfigurationService</name>
        <bindingTemplates>
                <bindingTemplate>
                  <description>Service description ...
                  <accessPoint>http://services.dcx.com/...
                  <tModelInstanceDetails/>
                </bindingTemplate>
                        </bindingTemplates>
                                </businessService>
```



Dienste

UDDI

SOAP

XML

Internet Protokolle (HTTP, TCP / IP)

⋮

Referenzen

XML allgemein:

- http://www.w3.org/TR
- http://www.xml.com
- http://www.oasis-open.org

Datenintegration (XML-Schema, Schema-Erzeugung, XSLT):

- http://www.w3.org/TR
- http://www.omg.org

Funktionsintegration (SOAP):

- http://xml.coverpages.org/soap.html
- http://xml.apache.org/soap/index.html

Prozeßintegration (WSDL, UDDI):

- http://www.uddi.org

Dieser Vortrag und weiterführende Informationen:

- http://www.jeckle.de