



LV8111 - Geschäftsprozessintegration

Eine Vertiefungsveranstaltung
im Master-Studiengang Informatik



Organisatorisches

Platzvergabe, Teambildung
Regeln zur Teilnahme und Scheinvergabe
Zeitplan
Literatur



- Regeln zur Teilnahme und Scheinvergabe
 - Modulhandbuch: „Projektarbeit“
 - Dazu wir auch Ihr Referat zählen, wobei das Thema nicht eng mit Ihrem Projekt verbunden sein muss!
- Referat
 - Dauer: 30 Minuten + max. 15 Minuten Diskussion
 - Wertung: 30%
 - Thema: zu vereinbaren, in Abhängigkeit von Vorkenntnissen und Interessen
- Projekt
 - Abnahme: Während der Klausurwochen
 - Wertung: 70%
 - Thema: Teil eines SOA-Szenarios, sofern nicht anders vereinbart



- Ermittlung der Vorkenntnisse
 - Anonymer Fragebogen – bitte gleich ausfüllen
 - Erläuterungen...
- Platzvergabe, Teambildung
 - 13 Bewerbungen für 15 Plätze → Alle erhalten einen Platz ☺
 - Regelfall: Zweier-Teams für die Projektarbeit
 - Bitte **Teamvorschläge nächste Woche** im Praktikum vorlegen!
- Zeitplan
 - Siehe Tabelle auf der nächsten Folie
 - Noch sehr vorläufig, hängt sehr von Ihren Interessen und Vorkenntnissen ab



Termine im SS 2009

Stand: 25.3.09



Datum (Do)	Vorlesung	Praktikum
19.03.2009	Vorbesprechung, Einl.	Kein Praktikum
26.03.	SOA-Grundlagen	Referatsthemen, Organisatorisches
2.04.	XML-RPC, SOAP	
9.04.	Gründonnerstag - vorlesungsfrei	
16. 04.	SOAP, REST	
23. 04.	WSDL	
30.04.	UDDI	
7.05.	WS Security / Referat	
14.05.	Referate	
21.05.	Christi Himmelfahrt	
28.05.	Referate	
4.06.	Referate	
11.06.	Fronleichnam	
18.06.	Referate	
25.06.	Referate	
29.6.-10.07	Mündliche Prüfungen, Klausuren etc.	



- Bücher
 - **Service-orientierte Architekturen mit Web Services**
Ingo Melzer et al. 3. Auflage. Spektrum Akadem. Verlag, Heidelberg, 2008.
ISBN (GTIN) 978-3-8274-1993-4
 - Programming Web Services with XML-RPC
Simon St. Laurent, Joe Johnston, Edd Dumbill. 1st ed. O'Reilly, Sebastopol, CA, 2001. ISBN 0-596-00119-3
 - Programming Web Services with SOAP
James Snell, Dough Tidwell, Pavel Kulchenko. 1st ed. O'Reilly, Sebastopol, CA, 2002. ISBN 0-596-00095-2
 - Professional XML Web Services
Patrick Cauldwell et al. 1st ed. Wrox Press, Birmigham, UK, 2001.
ISBN 1-861005-09-1
 - HTTP kurz & gut
Clinton Wong. 1. Aufl. O'Reilly, Köln, 2000. ISBN 3-89721-230-7
 - CGI Programmierung mit Perl
Scott Geulich, Shishir Gundavaram, Gunther Birznieks. 2. Aufl. O'Reilly, Köln, 2001. ISBN 3-89721-167-X



- Links
 - Siehe *website* des Kurses, u.a. die dortige Linksammlung:
<http://www.informatik.fh-wiesbaden.de/~werntges/lv/bpi05.html>
- E-Mail-Verteiler
 - Liste „im-lv8110“
 - <https://mail.informatik.fh-wiesbaden.de/cgi-bin/mailman/listinfo/im-lv8110>



Geschäftsprozessintegration

Einige grundsätzliche Gedanken zu Beginn



Geschäftsprozesse

unternehmens-
intern

unternehmens-
übergreifend

SOA

Web Services

EDI UN/EDIFACT

ESB

openTRANS

ebXML

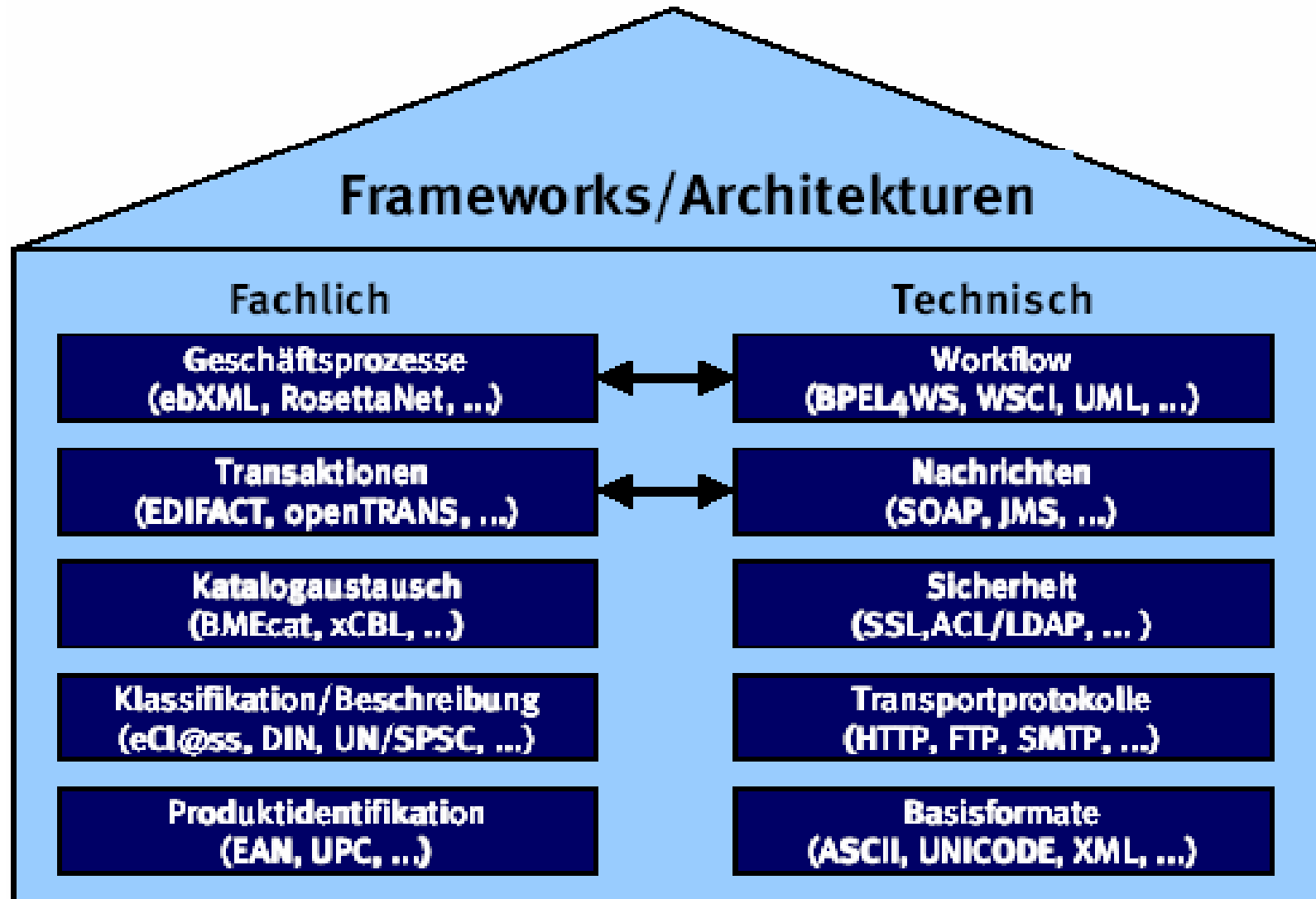
BPM

SAP XO

RosettaNet

ECR

EPCIS



Berlecon Research 2003



Einordnung in die Systematik

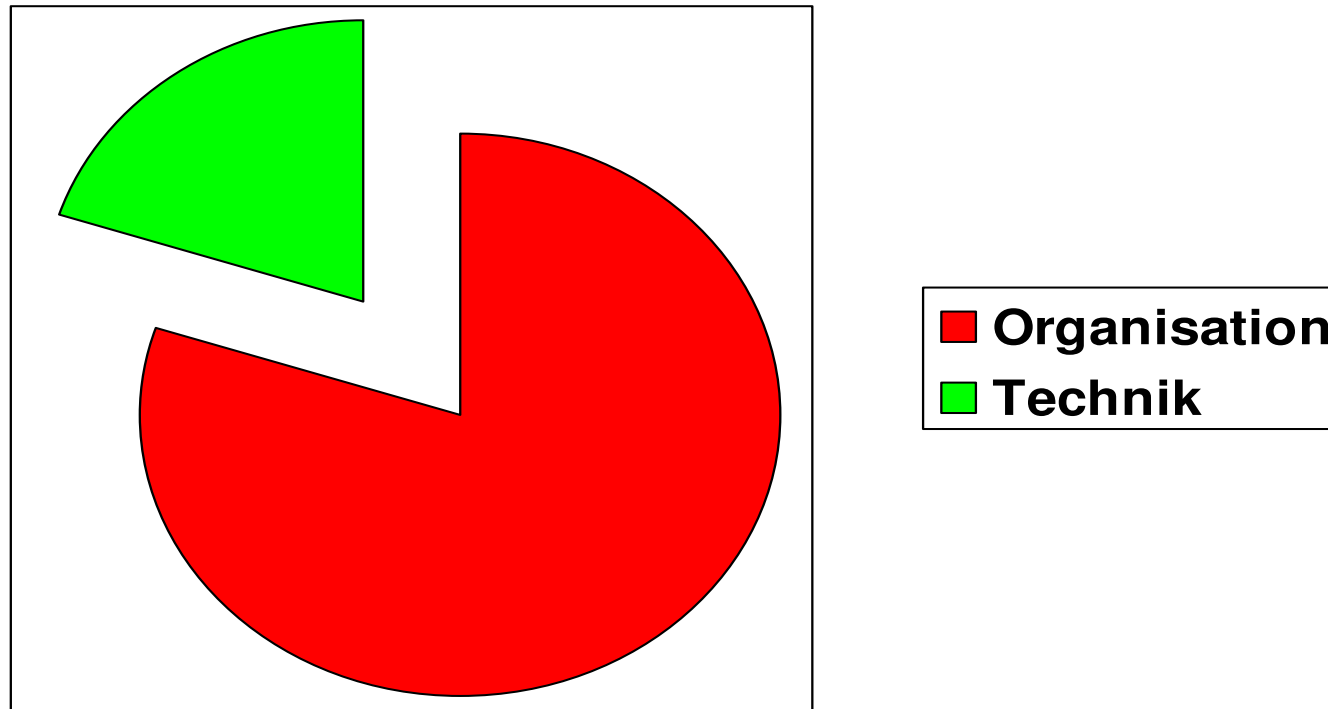


Standardname	Identifikation	Klassifikation	Katalogdaten	Transaktionen	Prozesse
BMEcat			■		
DATANORM, ELDANORM, etc.			■		
EAN•UCC	■				
ebXML					■
EDIFACT, EANCOM, etc.			■	■	
ETIM		■			
eClass		■			
openTRANS				■	
proficlass		■			
RosettaNet			■	■	■
UBL				■	■
UN/SPSC		■			
X12			■	■	
xCBL			■	■	

Berlecon Research 2003

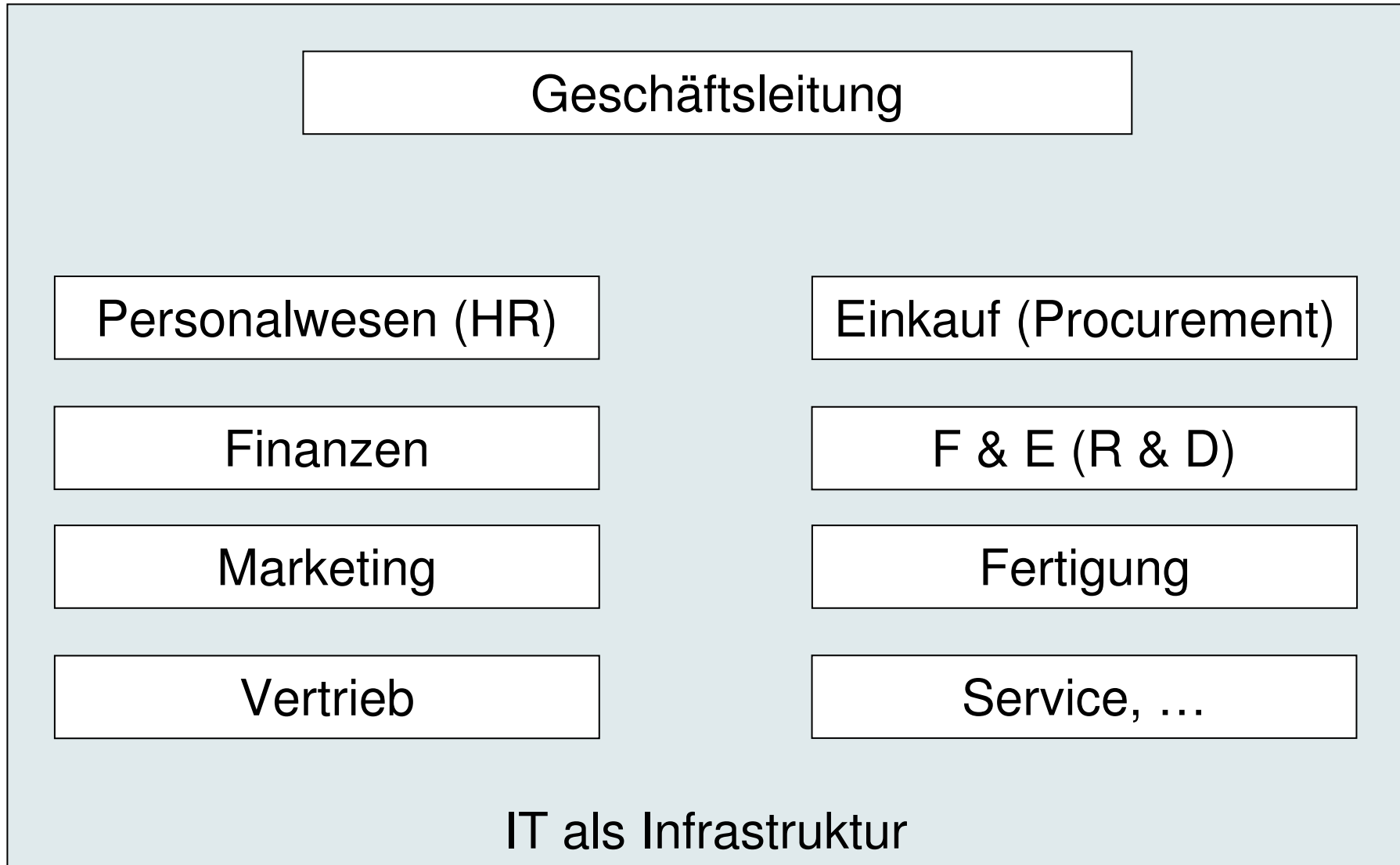


- Bekannte Missverständnisse / Die 80:20-Regel:
 - “Geschäftsprozessintegration ist i.w. ein technisches Thema”





Gedanken zur unternehmensinternen Geschäftsprozessintegration





Auswirkungen der funktionalen Struktur



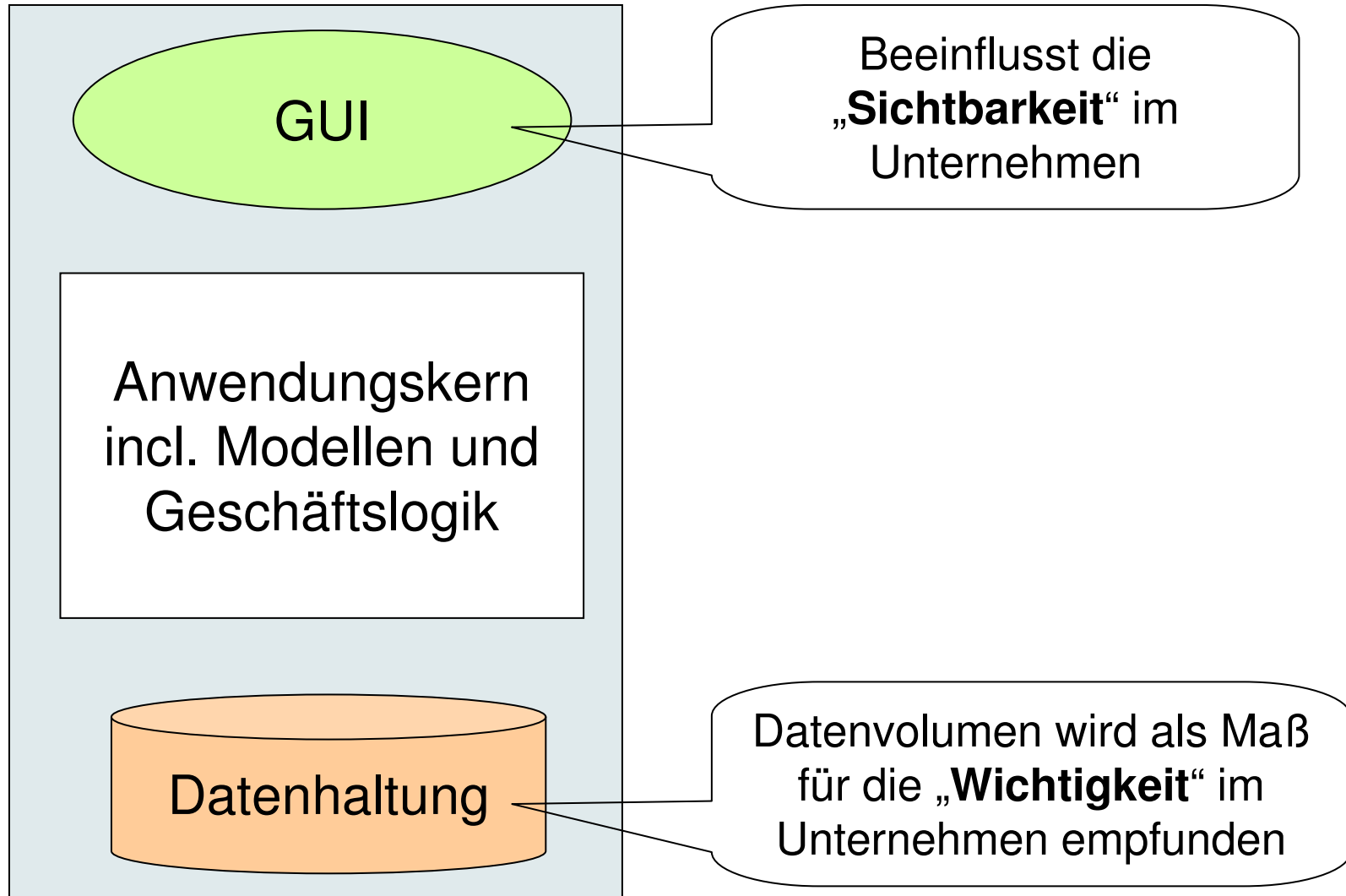
- Lange Zeit das vorherrschende Modell
 - IT-Systeme entstanden entlang der Grenzen dieser funktionalen Anforderungen
 - Komplexe und ausgereifte Leistungen mit eigenen UI *innerhalb* einer Funktionssäule, aber Schwächen bei der Interaktion
 - Langlebigkeit von IT-Systemen
 - Einsatzdauer einer Unternehmensanwendung > 30 Jahre ist nicht selten!
 - Beharrungseffekt der Funktionsträger („Fürstentümer“)
- Auch heute sind sehr viele IT-Systeme so geprägt!
 - Auch SAP R/3, nämlich in Form der Module wie HR, PP, SD, MM, ..



Auswirkungen der funktionalen Struktur



- Ein typisches IT-System zum Funktions-“Silo“ oder –“Schacht“





- 1990er Jahre: „*Business Process Re-engineering*“
 - Man erkannte, dass Unternehmen in Form von Geschäftsprozessen funktionieren
 - Firmen (einer Branche) konkurrieren
 - um die besseren Geschäftsprozesse
 - um die bessere Implementierung gleichartiger G.
 - Gute Unternehmen zeichnen sich aus durch
 - überlegene Geschäftsprozesse und
 - deren konsequente Implementierung
 - Die Folgen
 - Viele Unternehmen stellten ihre internen Abläufe um
 - Die IT-Infrastruktur musste aus diesem Anlass grundlegend verändert werden. Oft geschah dies durch Ablösung von Altsystemen durch SAP!
 - Leider wurde oft R/3 den (alten) Prozessen angepasst statt umgekehrt...



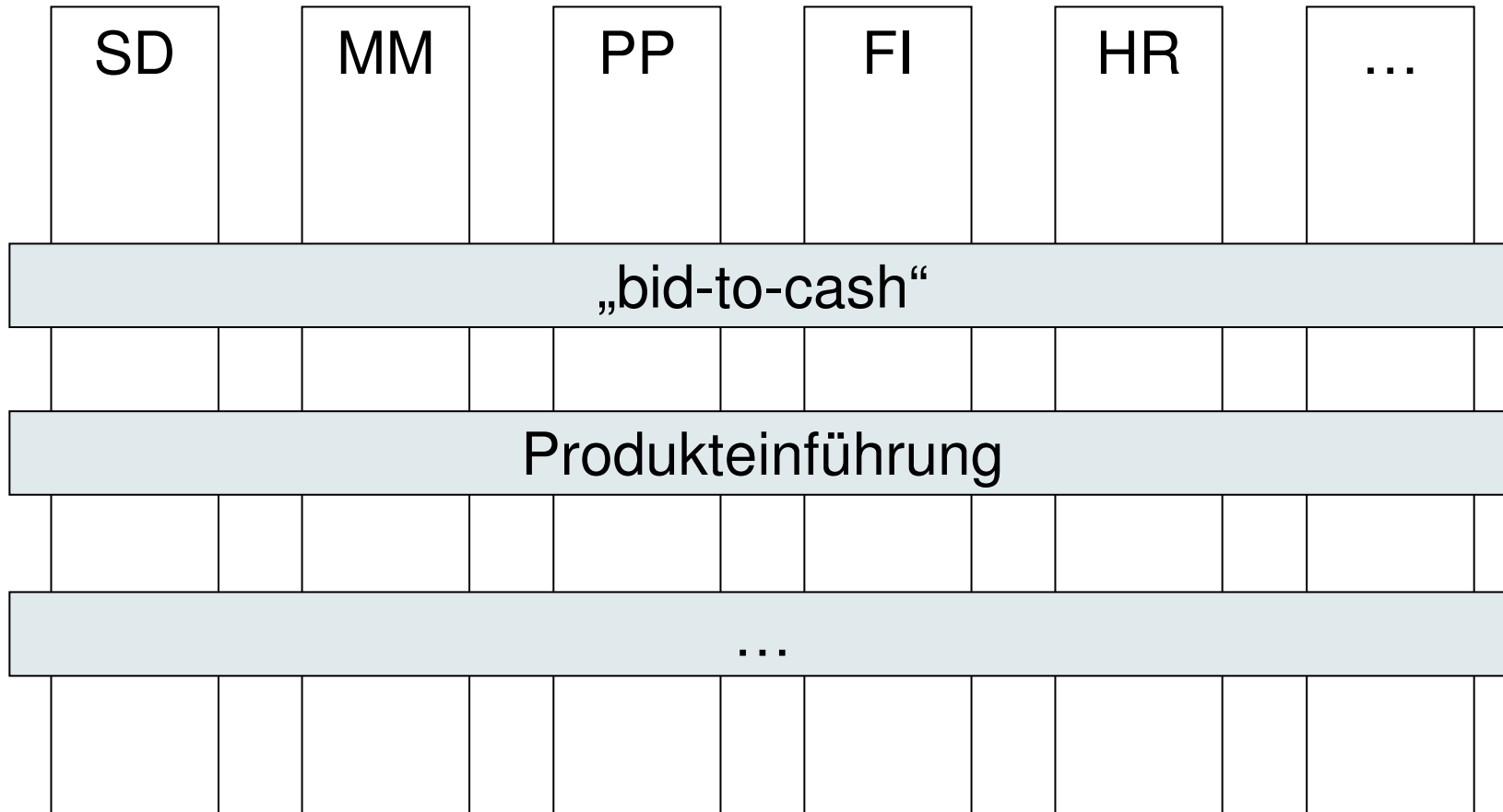
- Beispiel: PC-Fertigung
 - Klassischer Prozess:
 - Modellpalette entwickeln, auf Lager produzieren, vermarkten / abverkaufen
 - Alternative von Dell u.a.: „build-to-order“
 - Systeme werden aus Standardkomponenten frei wählbar zusammengestellt
 - Einzelfertigung statt Serienfertigung: Jeder Rechner wird erst auf Kundenauftrag hin gefertigt
 - Vorteile: Insb. bei schnellem technologischem Wandel - keine veralteten Geräte im Lager, geringe Kapitalbindung, große Vielfalt
 - Nachteile: Komplexität des neuen GP
- „build-to-order“ beherrscht inzwischen sogar die Automobilindustrie! (Farbe, Ausstattung)



Funktionssäulen und Prozesstunnel



SAP-Modulkürzel stellvertretend für Legacy-Systeme mit entsprechender Funktion





- Organisatorische Konsequenzen
 - Einführung von prozess-orientierten Positionen, z.B.
 - Produktmanager
 - SCM-Beauftragte (SCM = Supply Chain Management)
 - Matrix-Struktur
 - Funktionale Berichtslinie
 - Prozessorientierte Mitarbeit
 - Leitungsgruppen zur Konfliktauflösung
- Probleme
 - Neue GP sind schnell skizziert, aber die beteiligten Menschen folgen nicht (so schnell): Gewohnheit, lokale Nachteile, schlechte Kommunikation, ...
 - IT-Systeme folgen gar nicht, sondern müssen mit hohem Aufwand angepasst werden
 - **Folge: Forderung der Fachabteilungen nach Flexibilisierung der IT**



- Service-orientierte Architekturen (SOA) als Antwort
 - Legacy-Systeme erhalten **standardisierte Schnittstellen**, mit denen sie ihre **Kernfunktionen als Dienste** anbieten
 - Isolierte Front-Ends dieser Systeme werden ersetzt durch ein **einheitliches virtuelles „GUI“** (häufig auch per Unternehmensportal)
 - Kommunikation zwischen den Diensten über eine gemeinsame Infrastruktur (**Enterprise Service Bus, ESB**)
 - Dienste werden **formal beschrieben**
 - Schnittstellen bleiben langfristig **stabil**
 - Änderungen erfordern z.B. Genehmigung der Geschäftsleitung
 - Dienste werden **zentral erfasst**
 - Prozesse werden durch Kombination von solchen Diensten abgebildet, möglichst automatisiert und standardisiert

-
- Offene Fragen
 - Granularität der Dienste??
 - Chance der Wiederverwendung von Diensten?
 - Stabilität der neuen Gesamtlösungen?
 - „Eine Kette ist nur so stark wie ihr schwächstes Glied“
 - Regelung der Verantwortlichkeiten?
 - Antwortzeiten? Systemlast? Engpässe?
 - „Ein bisschen SOA“ lohnt selten, alles umzustellen ist zu teuer – was tun?

 - Folgen
 - Organisatorische Herausforderungen
 - Neue Aufgaben, z.B. BPM
 - Neue Risiken
 - Nur selten Kostenersparnis



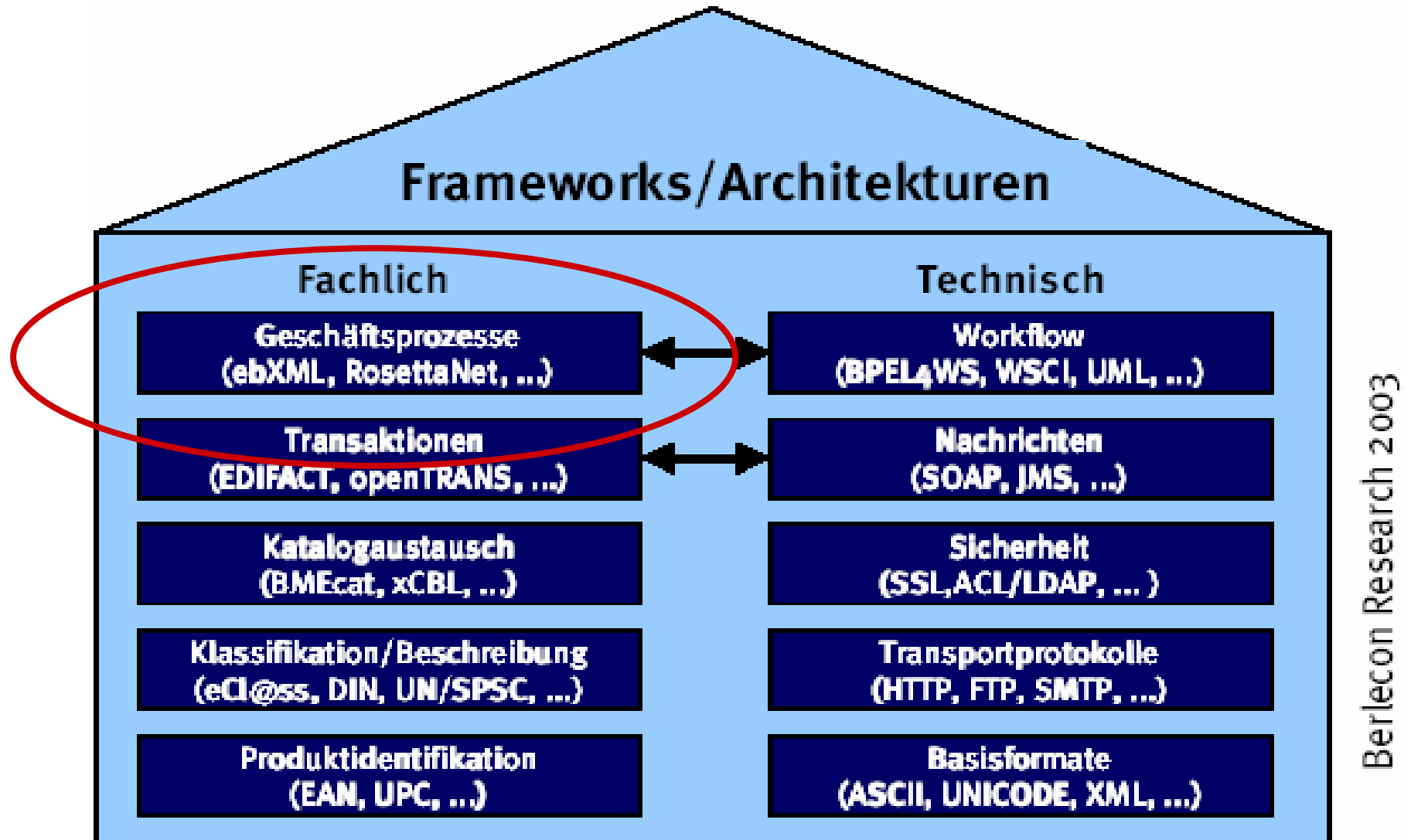
Situation bei unternehmensübergreifender Geschäftsprozessintegration



Geschäftsprozesse

Beispiel-Szenario: “*bid-to-cash*”

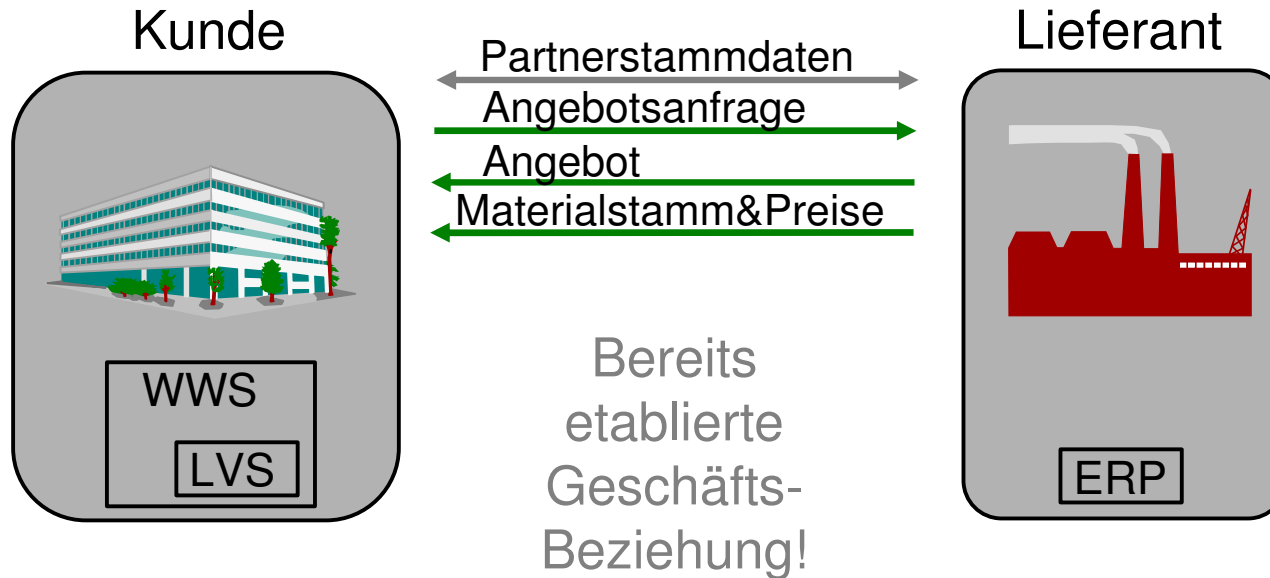
Ein Blick von "oben" auf den *stack*



Berlecon Research 2003



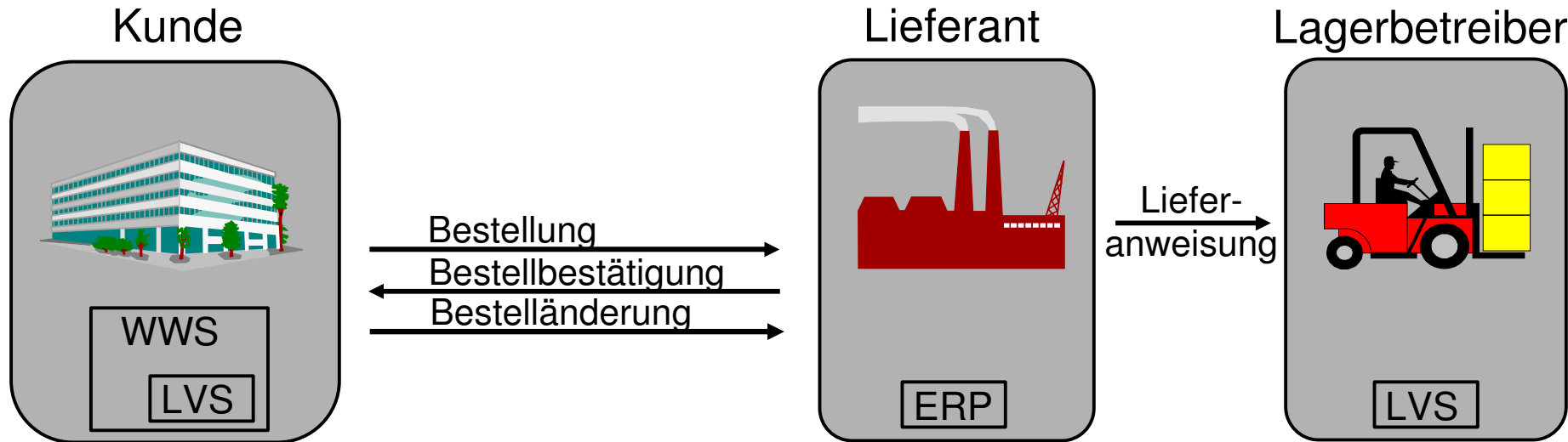
Weiter mit: Bestellabwicklung



WWS: Warenwirtschaftssystem
LVS: Lagerverwaltungssystem
ERP: Enterprise Resource
Planning system

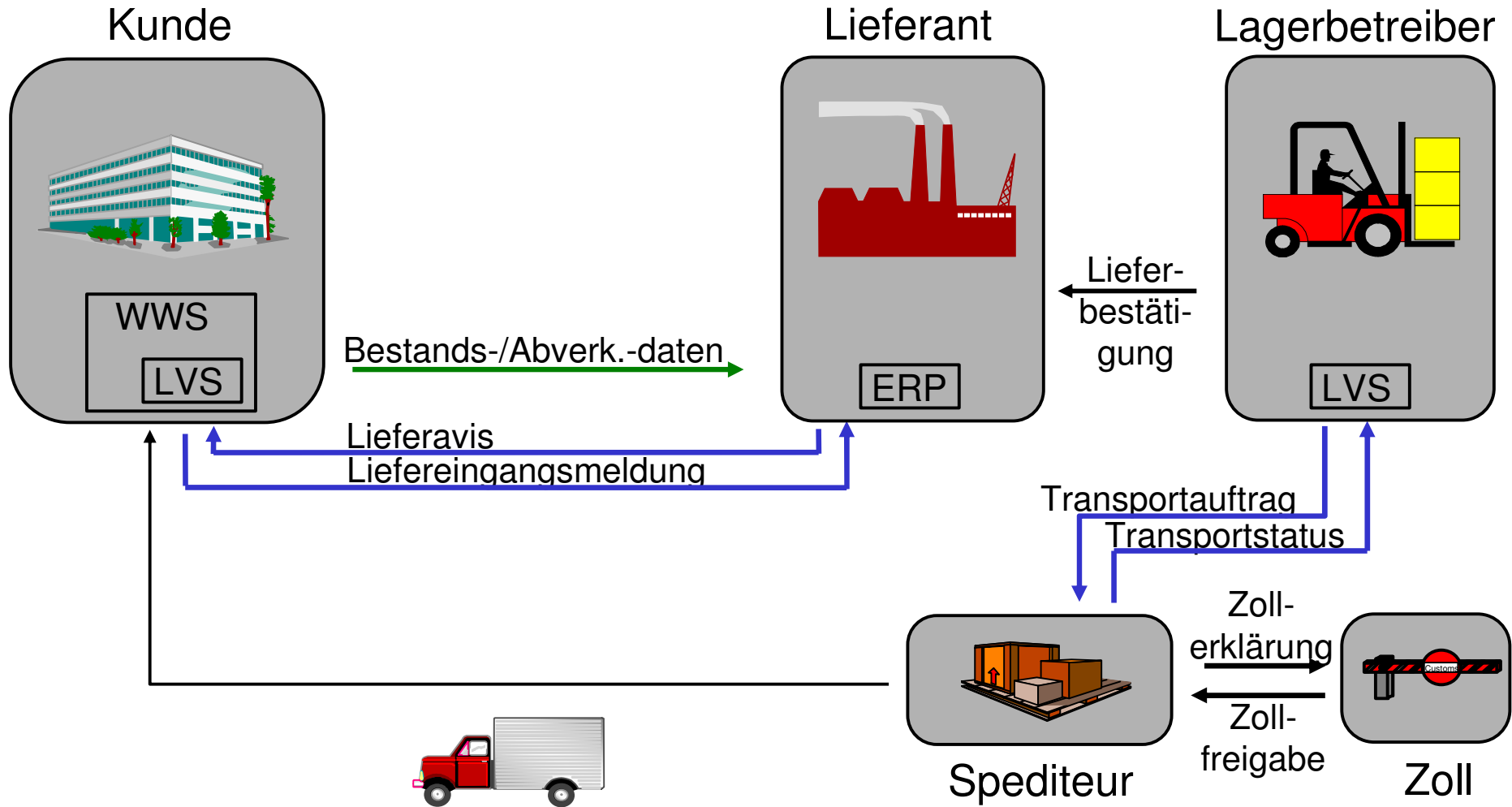


Weiter mit: Warenfluss



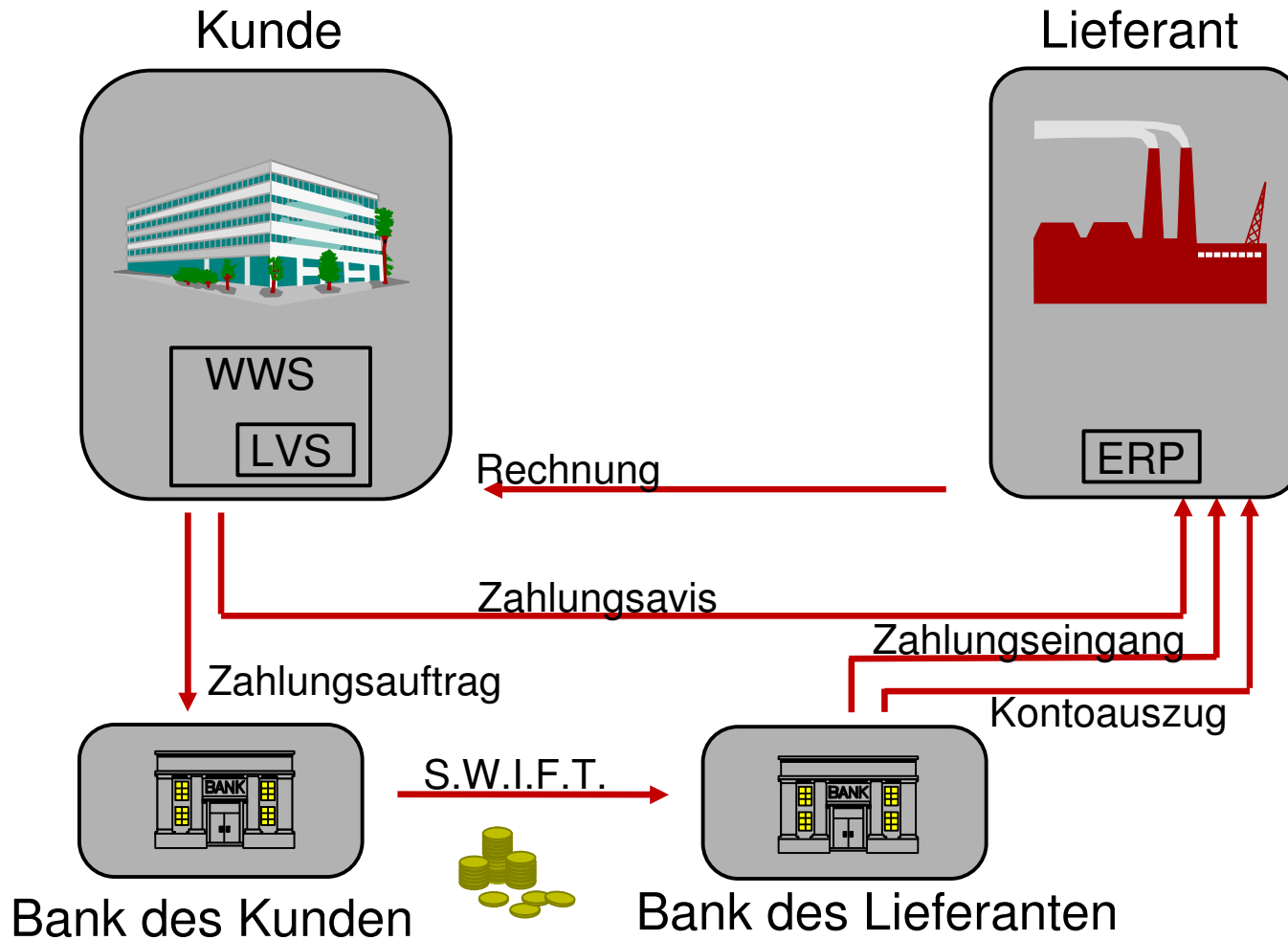


Weiter mit: Geldfluss





Weiter mit: Gesamtszenario

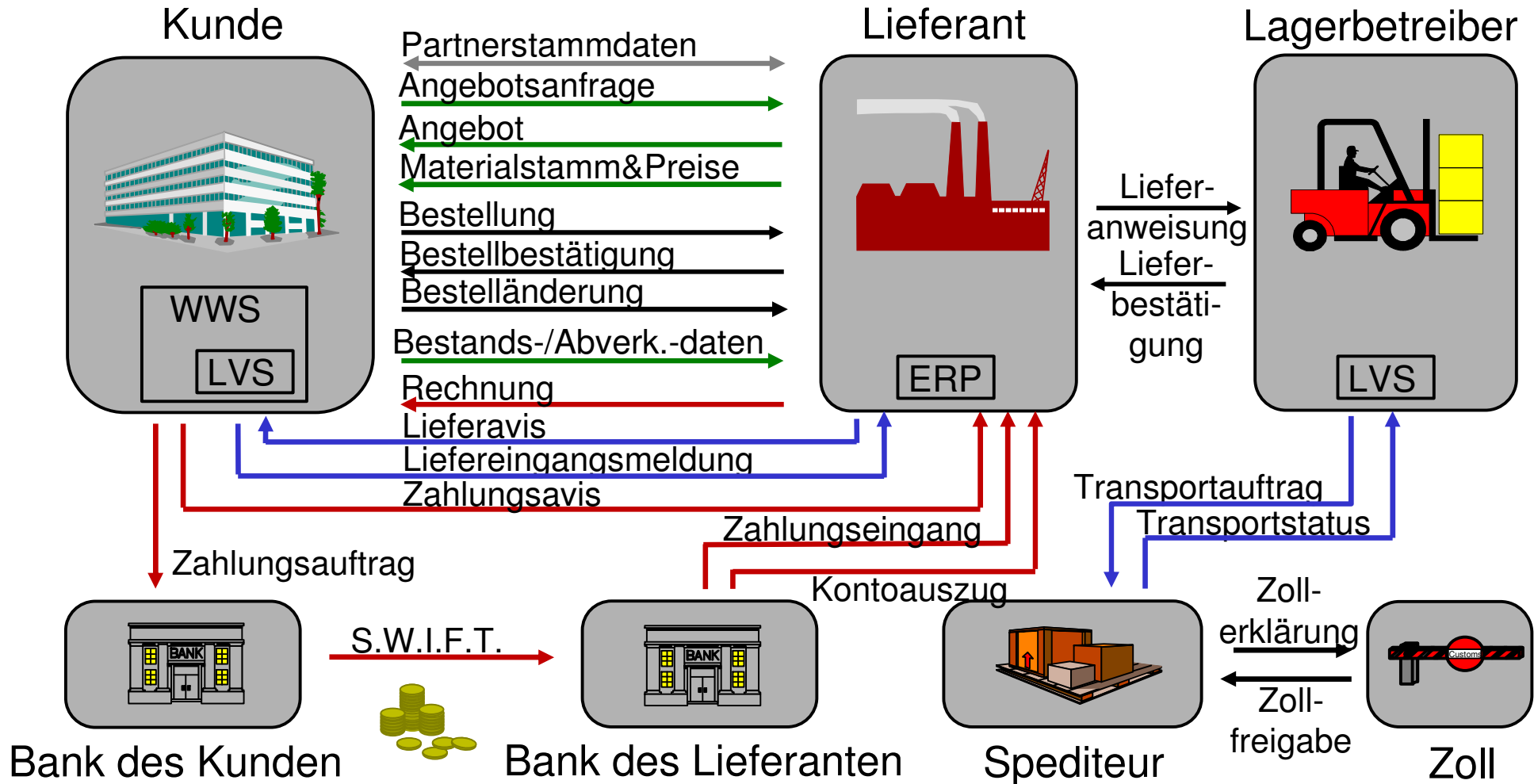




Gesamtszenario



BPI (nachrichten-orientiert)





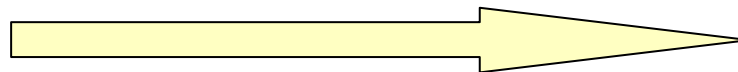
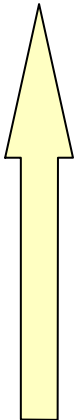
- Bedeutung von Standards
 - ... oder worauf sonst sollten sich gleichrangige Geschäftspartner einigen?
- Entwicklung von Standards
 - Ein echter IT-Dauerbrenner!
 - Siehe EDI-Geschichte, seit den 1970er Jahren
- Benötigt
 - (Semantische) Standards zur Beschreibung von Geschäftsdokumenten
 - Viel später: Auch von G.-Prozessen
 - Technische Standards zur Abbildung dieser Dokumente in Dateiform
 - Datenaustausch-Standards
 - Inhaltliche Unterstützung dieser Standards durch die Firmenanwendungen
 - Datenschnittstellen für den Import und Export im jeweiligen App.-Format
 - Konverter
 - Messaging-Komponenten



Der Weg zu UN/EDIFACT



	Proprietär	Branchenspezifisch	Branchen- übergreifend
International		EANCOM (subset): Handel+ S.W.I.F.T: Banks	UN/EDIFACT EANCOM (subset)
Regional		ODETTE (Auto, EU) RINET (Versicherung, EU)	ASC X.12
National		VDA (Auto, DE) SEDAS (Handel, DE / AT) GENCOD (Handel, FR)	TRADACOMS (UK)
Bilateral	BAV (Siemens) VW Formate		





- **1977: SEDAS**
 - “Standardregelungen einheitlicher Datenaustauschsysteme”
 - Industrie und Handel, DE und AT
 - CCG - Centrale für Coorganisation, Köln (heute GS1 Germany)
- **1978: VDA-Norm**
 - VDA: Verband der deutschen Automobilindustrie e.V.
- **1981: GTDI**
 - Als TDID Teil 4 (Syntaxregeln) von UN/ECE veröffentlicht
 - GTDI: *Guidelines for Trade Data Interchange*
 - TDID: *Trade Data Interchange Directory*
- **1982: ANSI ASC-X12**
 - *Accredited Standard Committee X12 for Electronic Business Data Interchange (EBDI) by the Am. Ntl. Standards Institute*



- **1987: EDIFACT Syntaxregeln**
 - Überarbeitung / Aktualisierung durch die UN/JEDI Group
 - Synthese aus GTDI und ANSI X.12
 - Juli: UN/JEDI Group verabschiedet
 - *Message Design Guidelines*
 - Erste Nachricht: INVOIC
 - September: Übernahme der UN/ECE-Empfehlungen der EDIFACT-Syntaxregeln durch ISO, CEN, DIN:
 - International: **ISO 9735** (15. Juli 1988)
 - EU-Ebene: EN 27 372
 - Deutsche Norm: DIN 16559
- **1990: EANCOM** (wichtiger Subset)
- **Mitte 2007: D.07A – 195+17** Nachrichtentypen



Standardisierung

Warum Standards?
Zu standardisierende Ebenen



Warum Standards?

Das Skalierungsproblem
Alternativen



Phase 1: Inselbildung



Bi-laterale
Absprachen:



Definitions-
aufwand $\sim N$

Installations-
aufwand $\sim N$



Nur effizient
für kleine
Insellösungen





Phase 2: Wachstum

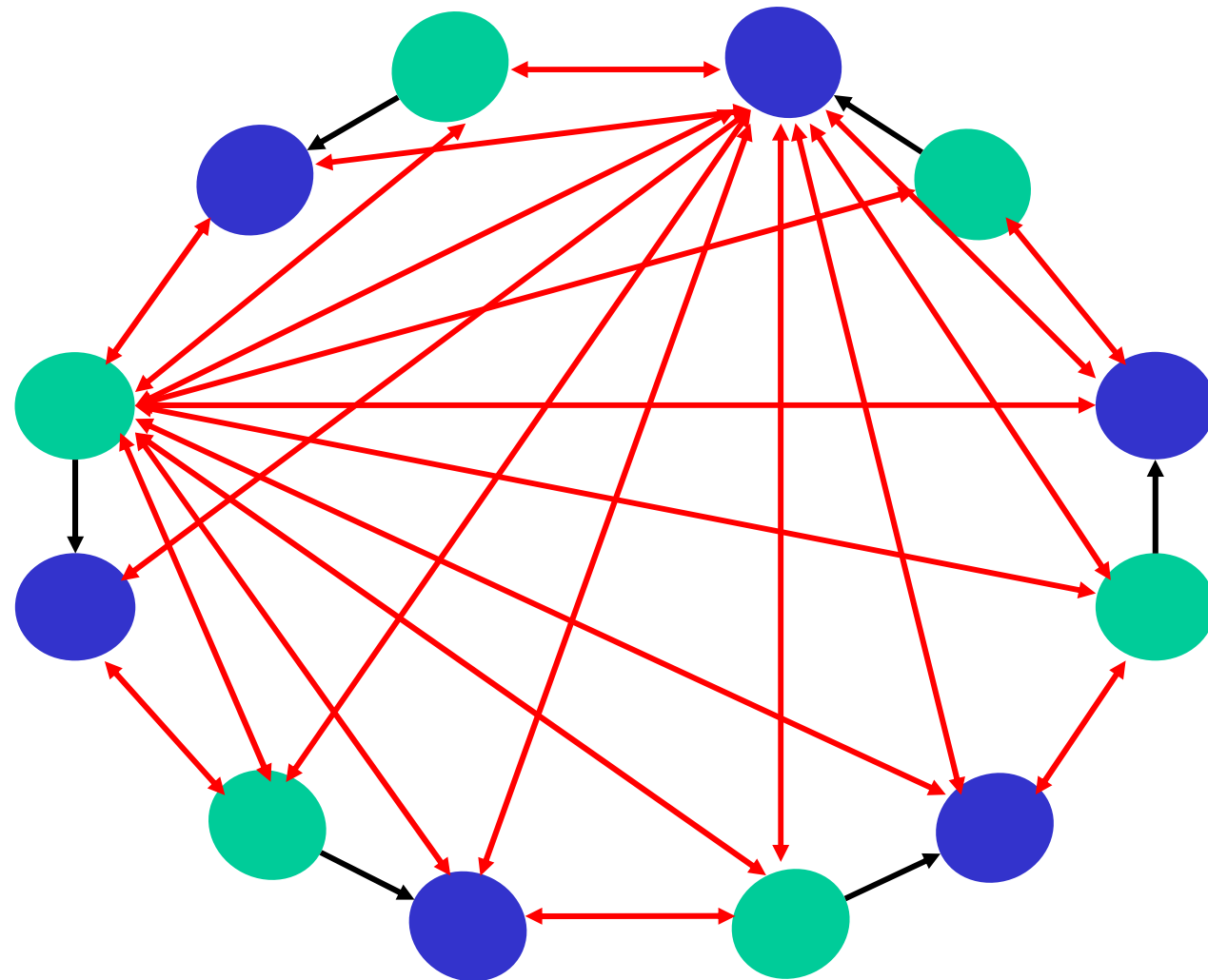


Jeder verbunden
mit vielen anderen:

Definitions-
aufwand $\sim N^2$

Installations-
aufwand $\sim N^2$

Teuer und
zeitraubend!



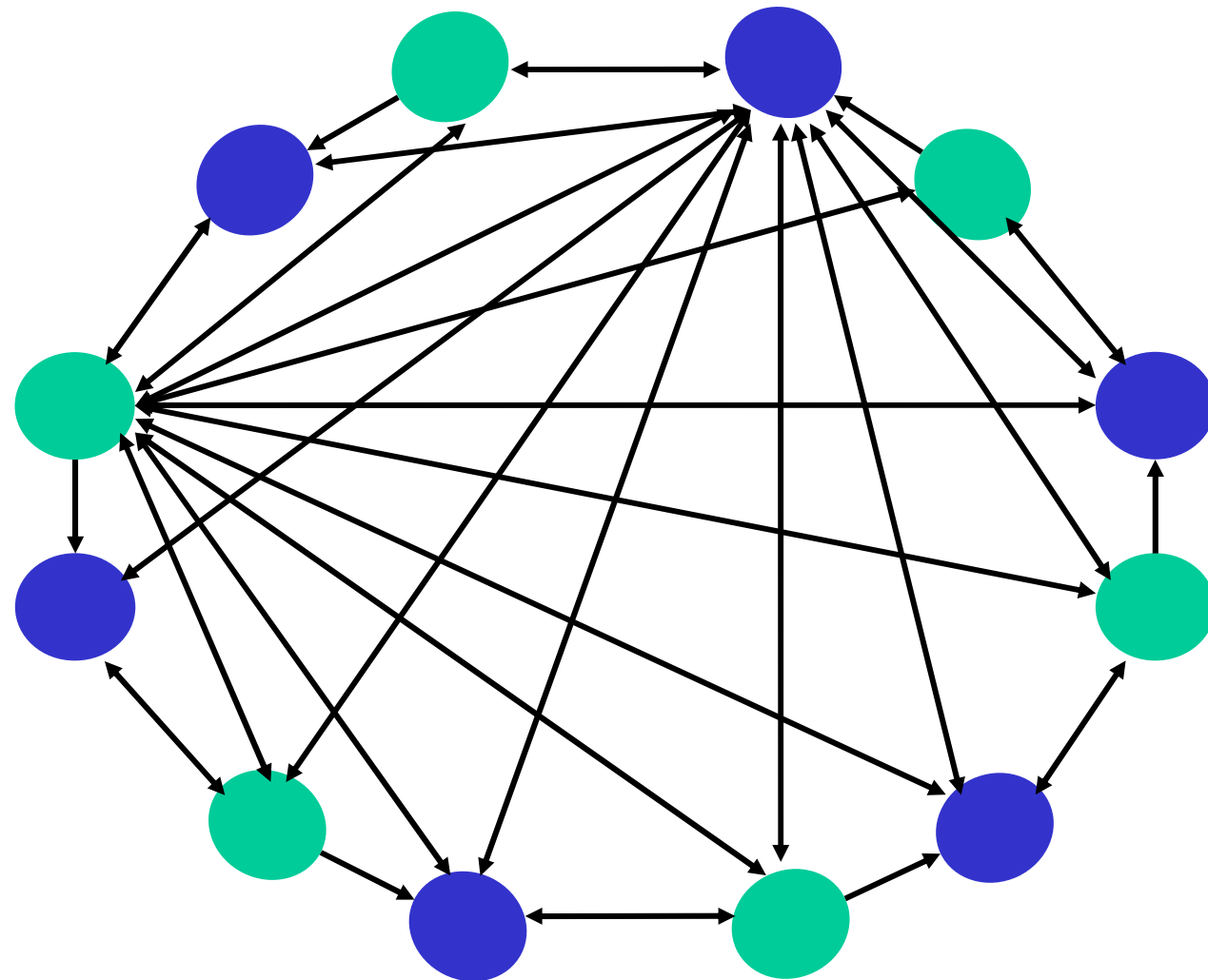


Ein gemeinsamer
Standard!

Definitions-
aufwand $\sim \text{const.}$

Installations-
aufwand $\sim N$

Nur so sind große
EDI-Netzwerke
effizient erreichbar!





- Insellösungen
 - Skalierungsproblem wird umgangen
 - Inseln mit unterschiedlichen Einzellösungen wachsen nicht zusammen
 - Akzeptabel oder durchsetzbar, wenn Mehrfachaufwand nur von wenigen Teilnehmern zu tragen ist.
- Zentralistischer Ansatz
 - Ein dominierender Geschäftspartner schreibt seine Verfahren den anderen Teilnehmern verbindlich vor.
 - Historisches Beispiel: Automobilindustrie
 - Nachteile
 - Nicht skalierbar,
 - nicht mit Globalisierungstrend kompatibel,
 - nicht praktikabel bei Gemeinschaften ähnlich starker Partner



- **Geschäftsprozesse und -praktiken**
 - Beispiel-Initiative: ECR
 - (Efficient Consumer Response), <http://www.ecrnet.org>
- **Ident-Systeme**
 - Beispiel: EAN
 - <http://www.ean-int.org>, <http://www.gs1.org>
- **Datenstrukturen**
 - Syntax, z.B. UN/EDIFACT
 - Belegarten, -aufbau, z.B. ORDERS, EANCOM-Subset
- **Datenaustausch-Verfahren**
 - Beispiele: VANS, X.400, http, AS/2
- **Strukturierter und ausführlicher:**
 - **Der Berlecon-Stack!**

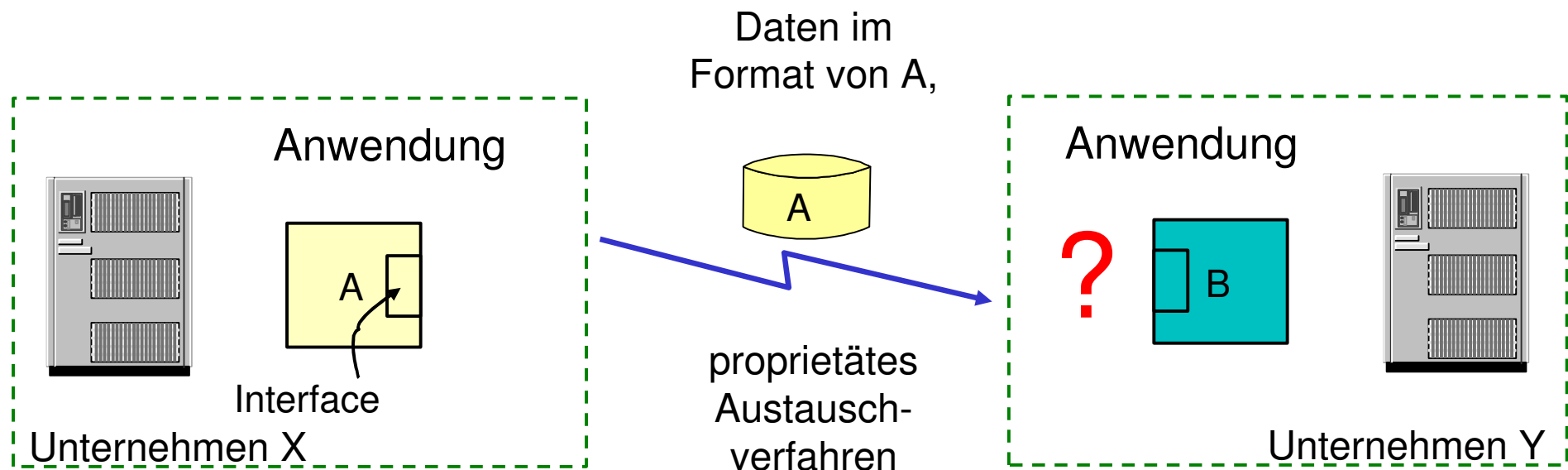


EDI-Kernkomponenten

Eine eher technische Sicht



Anwendungen mit Schnittstellen ...



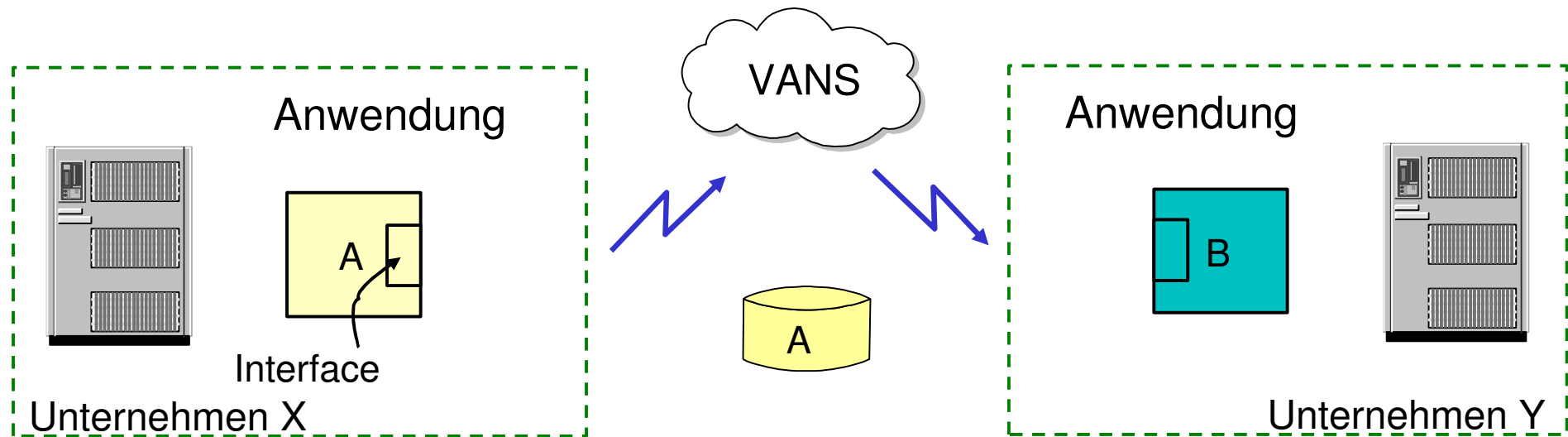
- ✓ Anwendbar für bilaterale Projekte
- ✓ B richtet sich nach A



...



... und VAN-Unterstützung ...

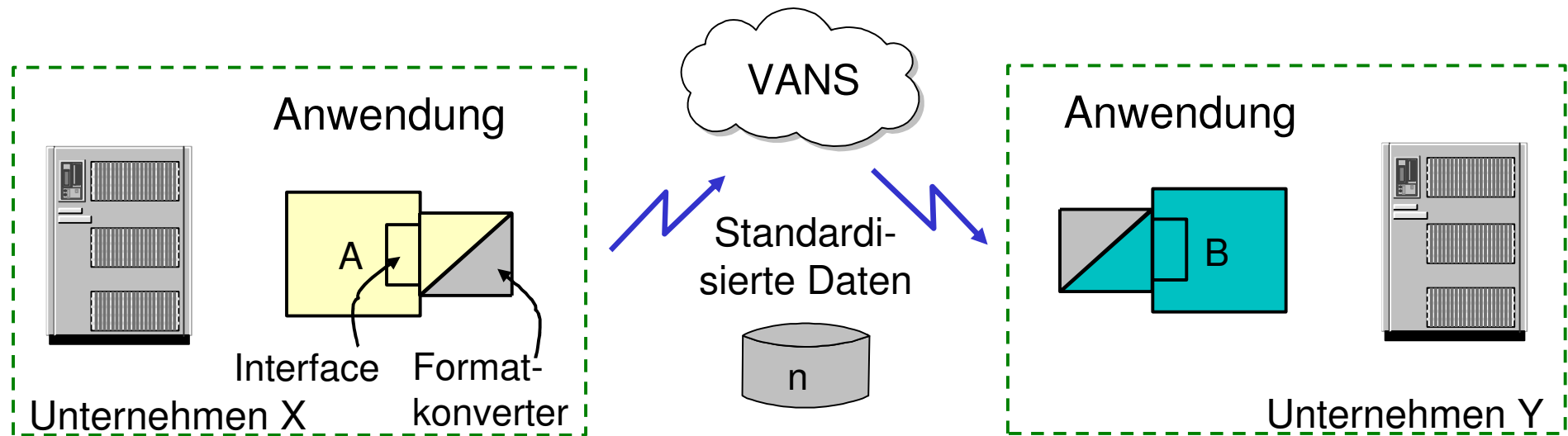


- ✓ Viele Kommunikationsverbindungen
- ✓ Alle Empfänger richten sich nach A



Veraltet!

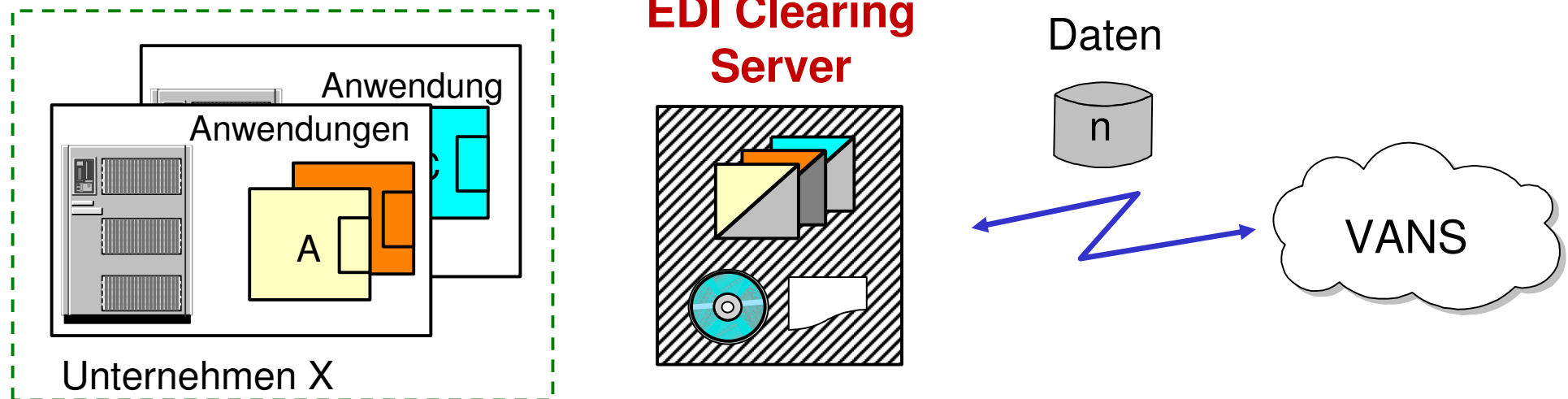
Front-End Konzept



- ✓ Viele Kommunikationsverbindungen
- ✓ Viele EDI-Partner



Der EDI-Server - Eine zentrale IT-Ressource



- ✓ Viele Kommunikationsverbindungen
- ✓ Viele EDI-Partner
- ✓ Verschiedene Anwendungen
- ✓ Verschiedene Geschäftseinheiten



Die Kernkomponenten von EDI



1. Anwendungsschnittstellen und -Formate
2. EDI-Standardaustauschformate
3. *Mapping*
4. *Reliable Messaging / File Transfer*
5. Extras
 - *Routing*
 - Archivierung
 - *Reporting*
 - Alarmierung
 - *Tracking & Tracing*
6. Organisatorische Voraussetzungen (!!!)

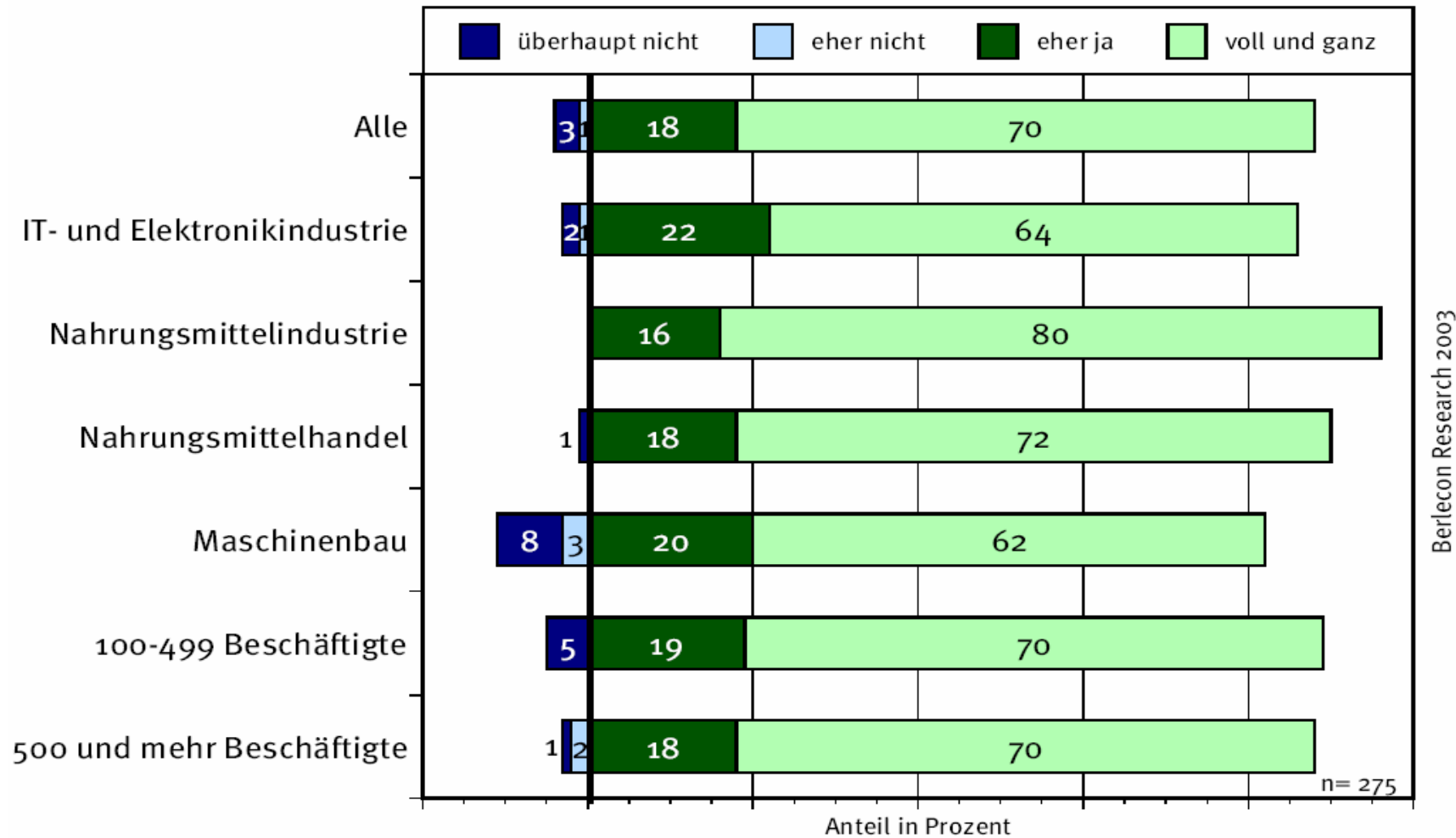


Berlecon: Nutzerbefragung

Einige (subjektiv) ausgewählte Ergebnisse
mit Schwerpunkt EDI



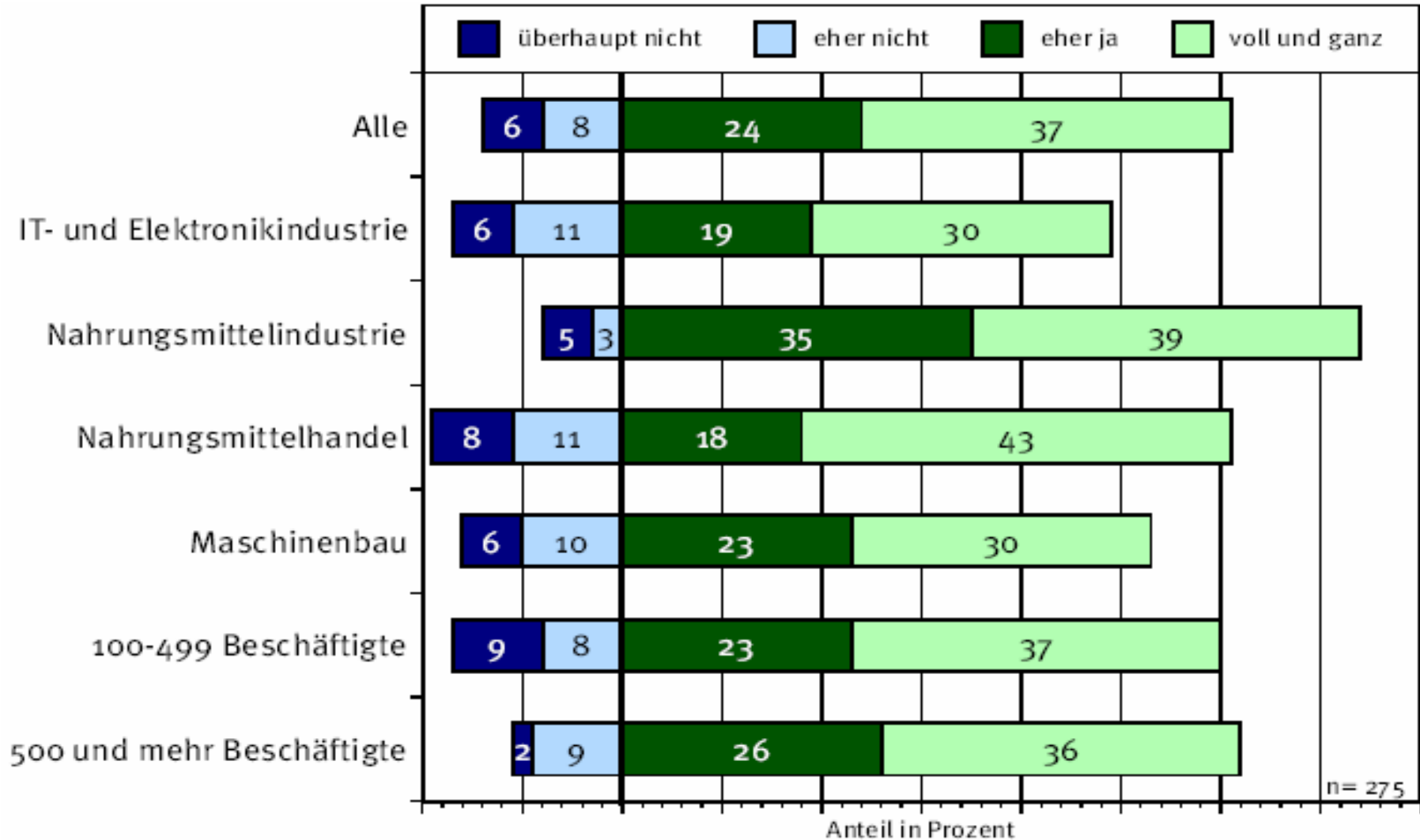
EDI wird für uns auch in einigen Jahren eine Rolle spielen

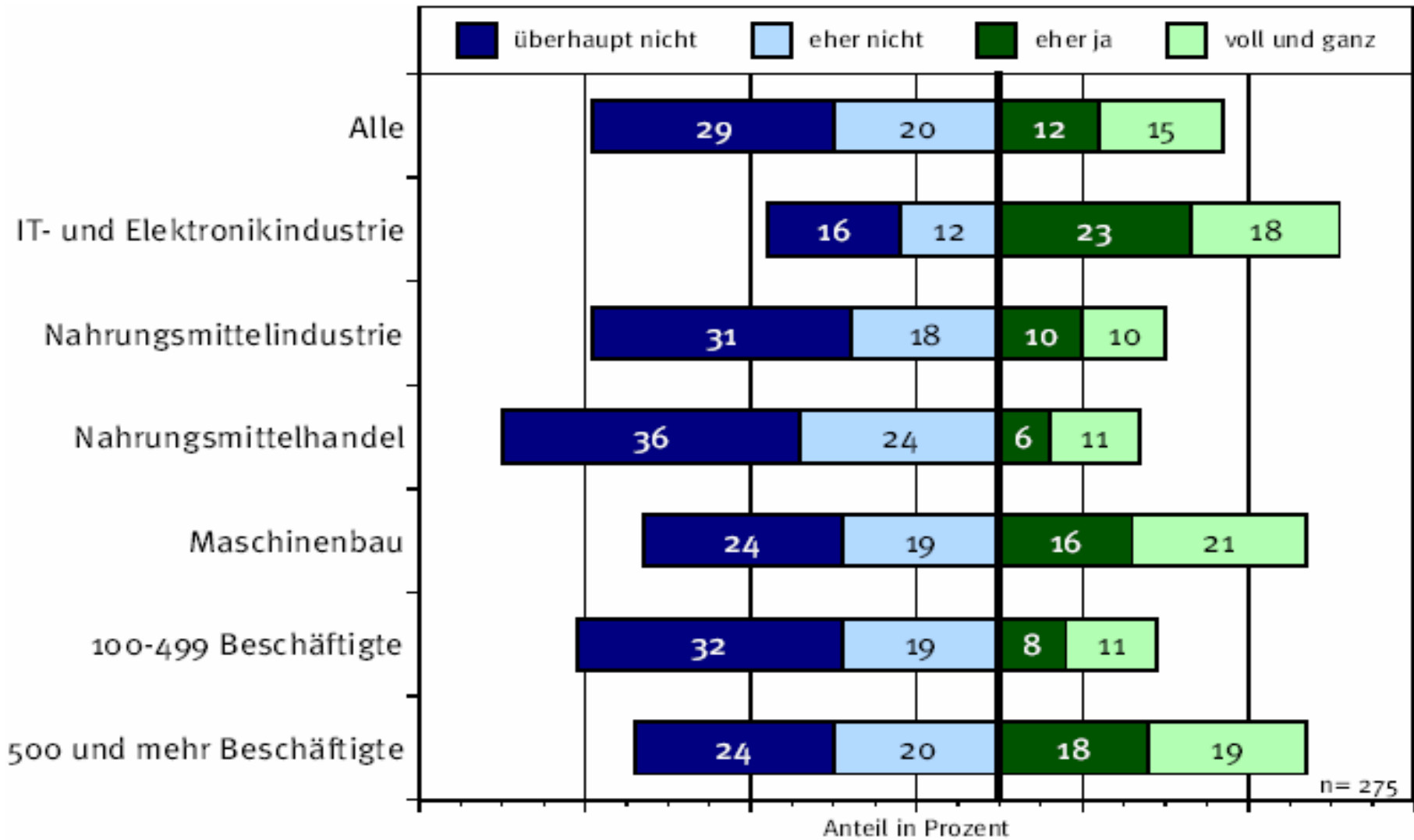


Grundgesamtheit: Daten austauschende Unternehmen, die EDI nutzen.



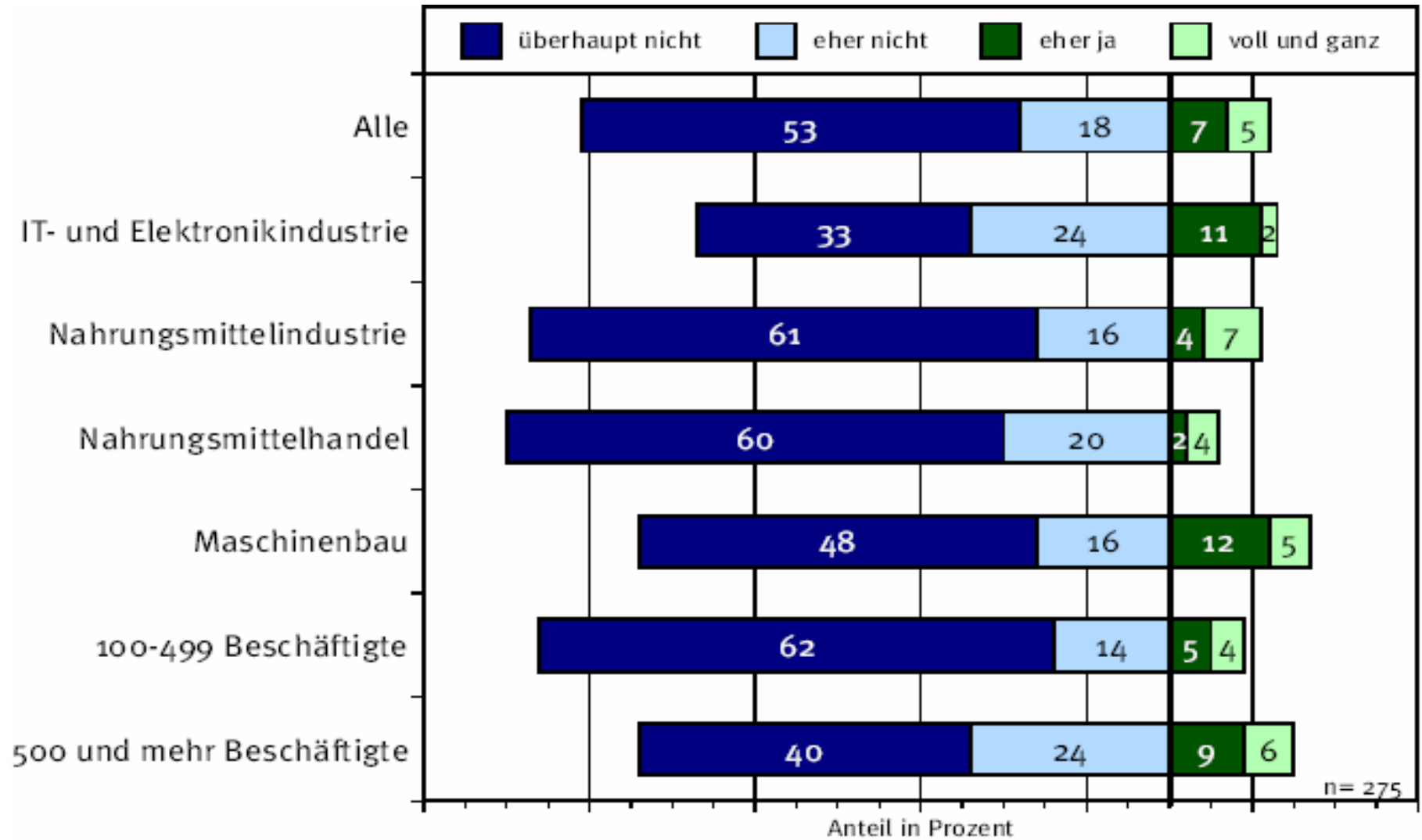
Kompaktes Datenformat von EDI ist für uns wesentlicher Vorzug





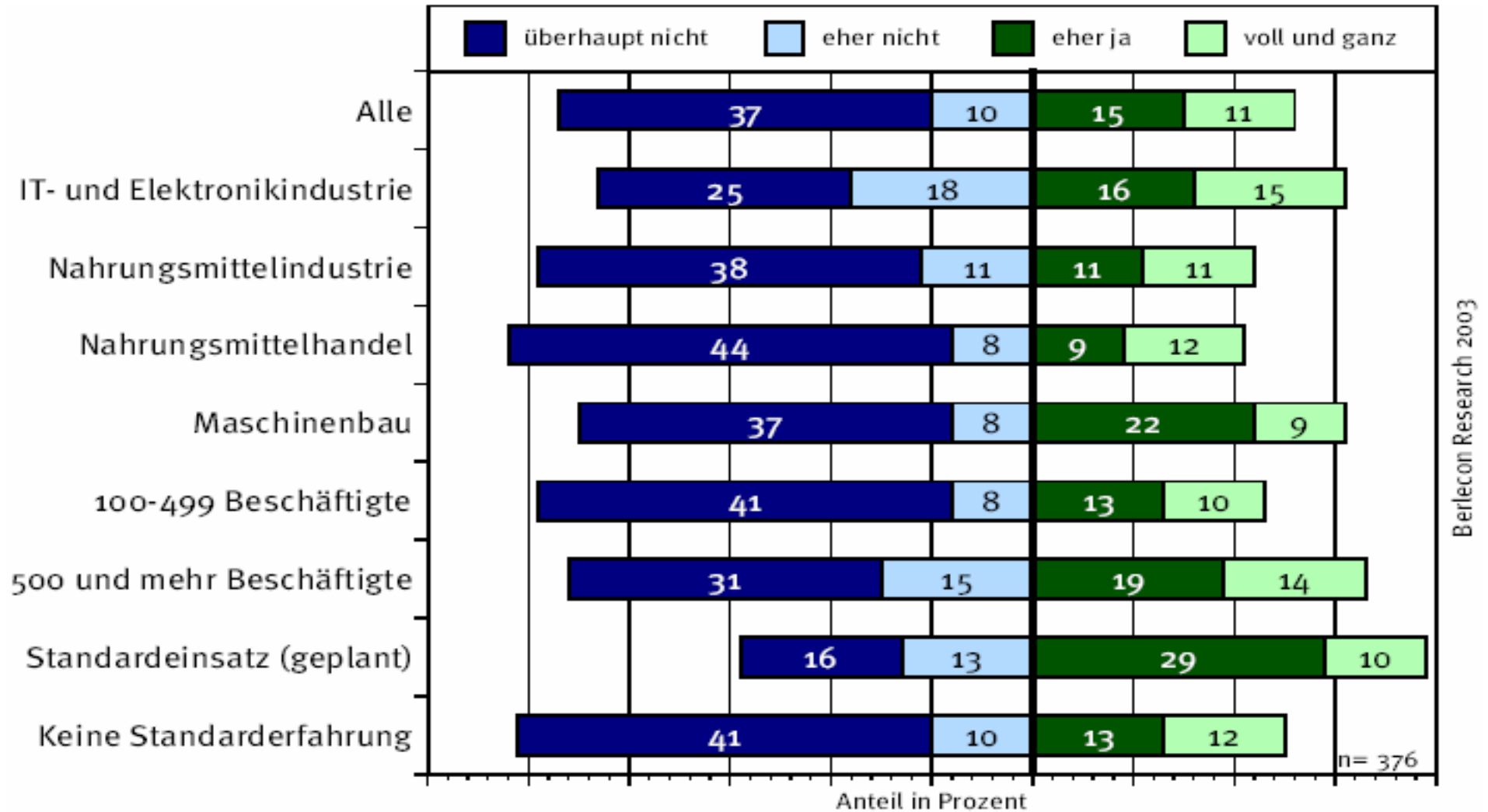


Ersatz von EDI-Lösungen durch XML-basierte Lösungen



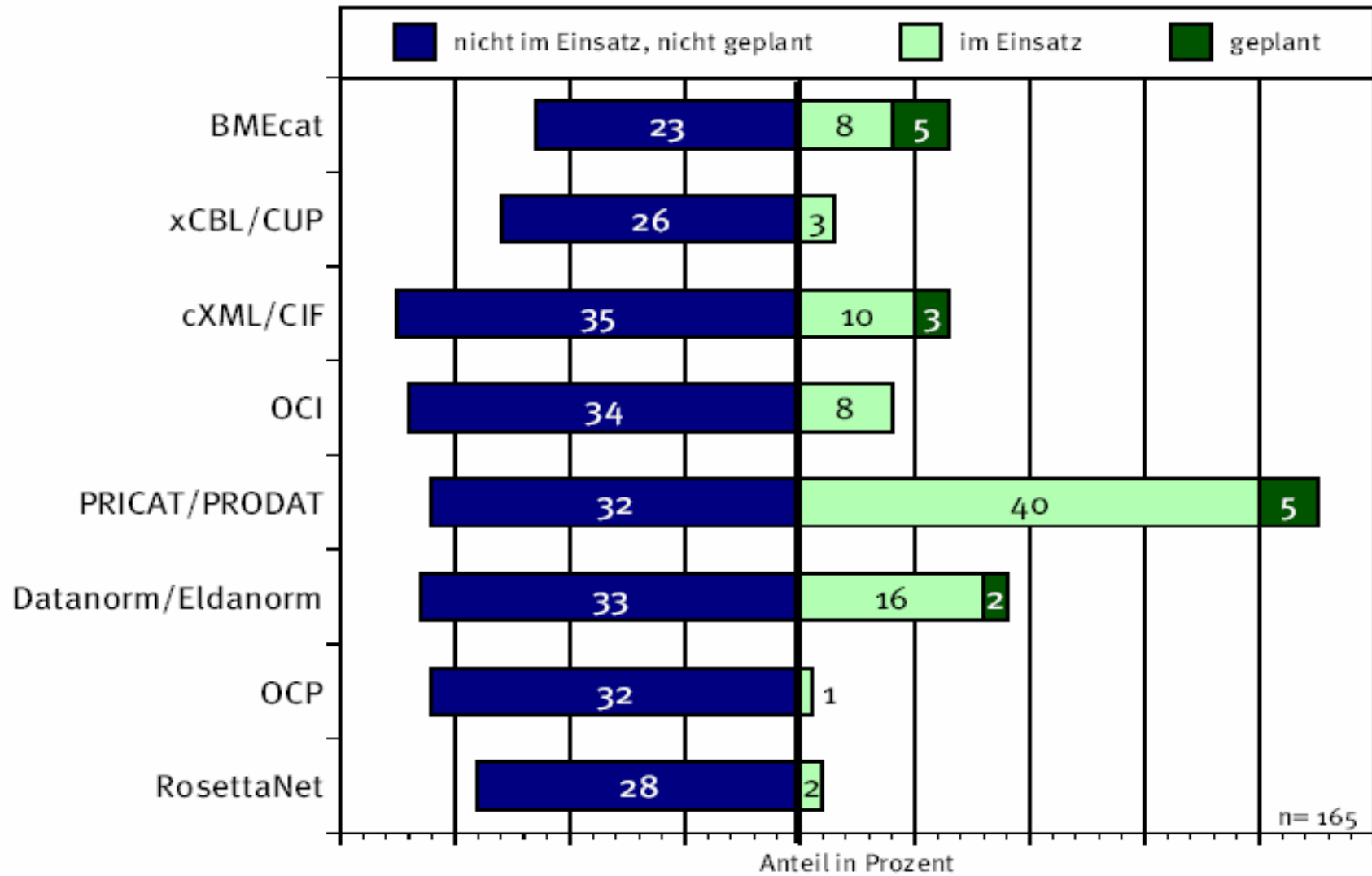


Nebeneinander verschiedener Klassifikationsstandards ist Problem für uns



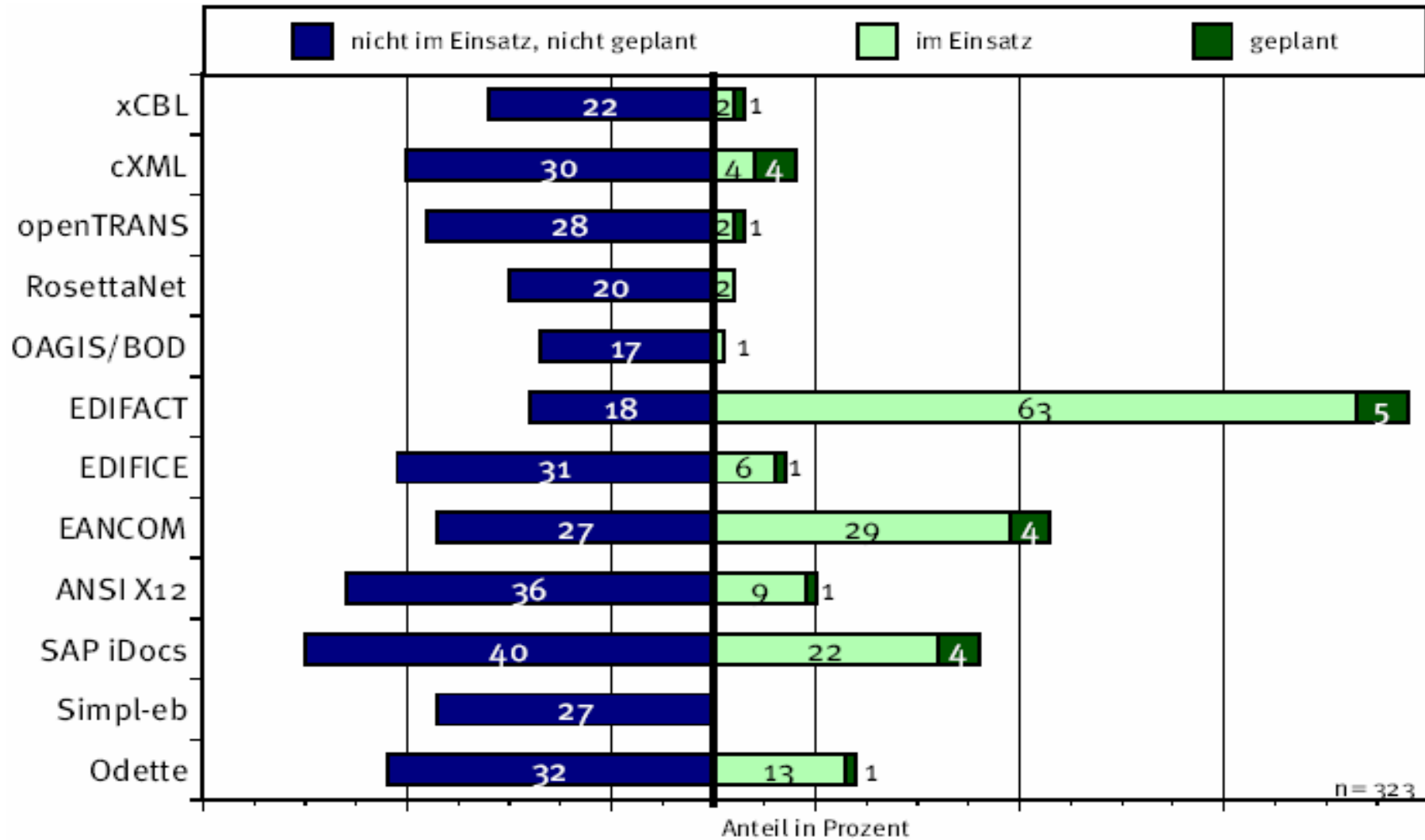


Aktuelle und geplante Nutzung von Katalogaustauschformaten



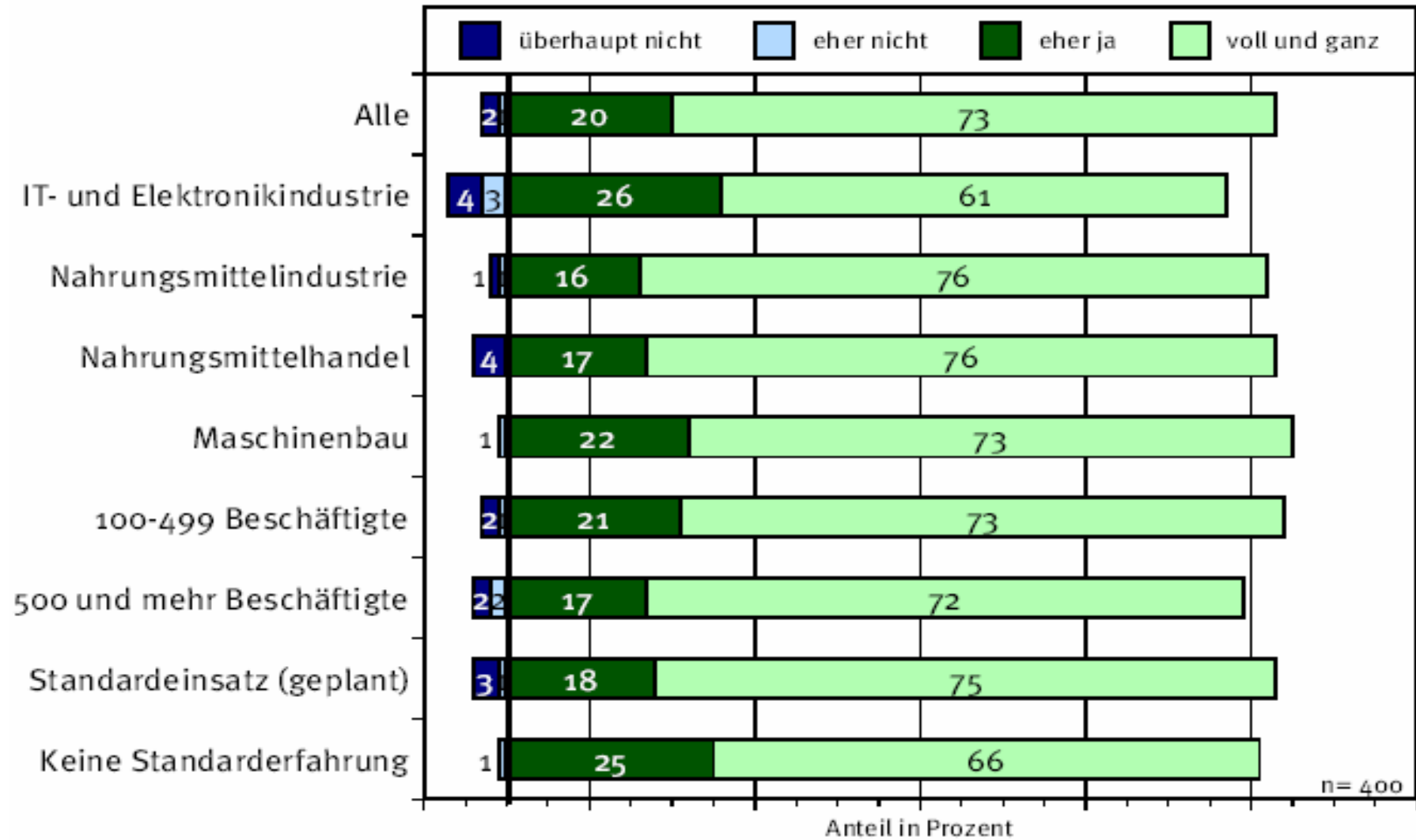


Aktuelle und geplante Nutzung von Transaktionsstandards



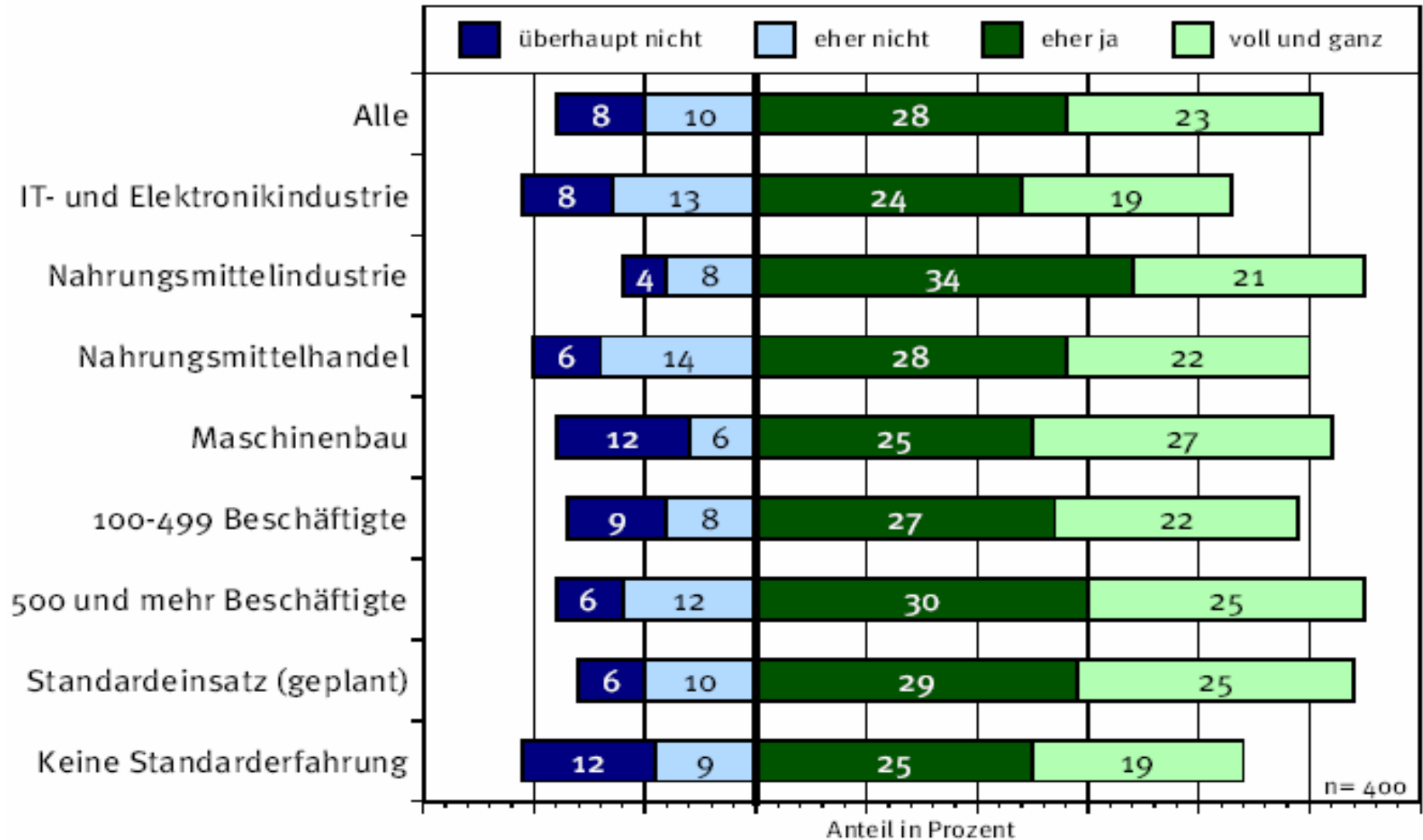


E-Business-Standards sollten weltweit einheitlich sein





Standards lösen unsere Integrationsprobleme



Berlecon Research 2003



Standardisierungsgremien



- W3C (World Wide Web Consortium)
 - NICHT direkt von Firmen abhängig (aber mit vielen Firmenvertretern als Mitwirkenden)
 - Grundlagen: HTML, XML, XML Schema, ...
 - WS-Standards: SOAP, WSDL u.v.a., auch Strategie- u. Architekturkonzepte

<http://www.w3.org>
- OASIS (Org. for the Advancement of Structured Information Standards)
 - Non-profit Organisation, bestehend aus Firmenvertretern
 - Ursprünglich „SGML-Open“; DocBook u.a. DTDs
 - Neben WS-Standards auch UDDI, ebXML, WS-BPEL, ...

<http://www.oasis-open.org>
- IETF (Internet Engineering Task Force)
 - In unserem Kontext relevant für SSL/TLS, LDAP, IPv6

<http://www.ietf.org>



- UN/CEFACT
 - Zentrum für Handelserleichterungen und E-Business, Teil der Vereinten Nationen!
 - Mammut-Aufgabe, u.a. für UN/EDIFACT und ebXML (mit)verantwortlich

<http://www.unece.org>

- WS-I (Web Services Interoperability Organization)
 - Gibt selbst keine Standards heraus, sondern definiert Profile mit Beschreibungen zur Benutzung von konkreten Implementierungen von Standards, um interoperabel zu sein

<http://www.ws-i.org>



SOA

Material zur Diskussion & Vorstellung



Wer "redet" mit wem?

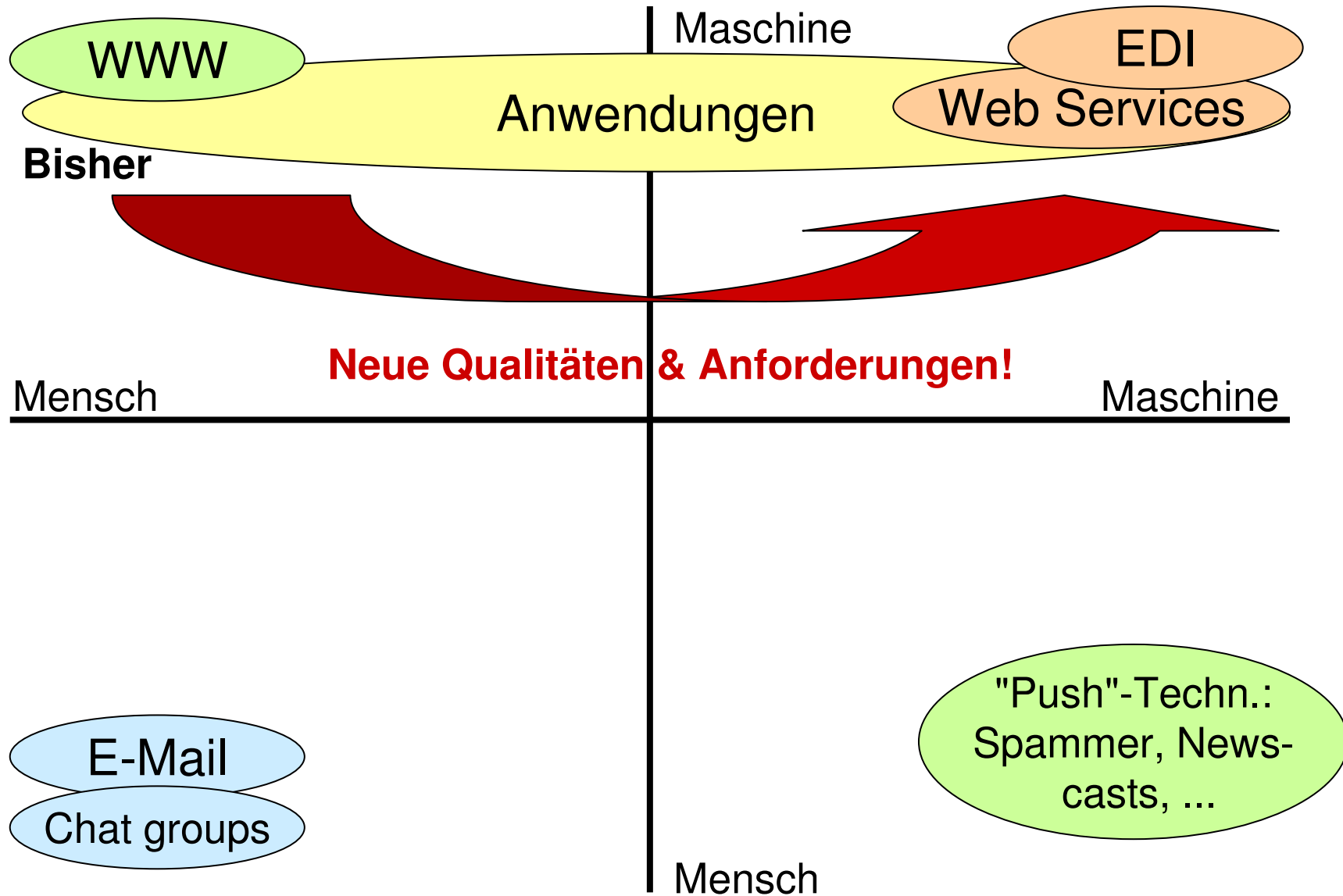
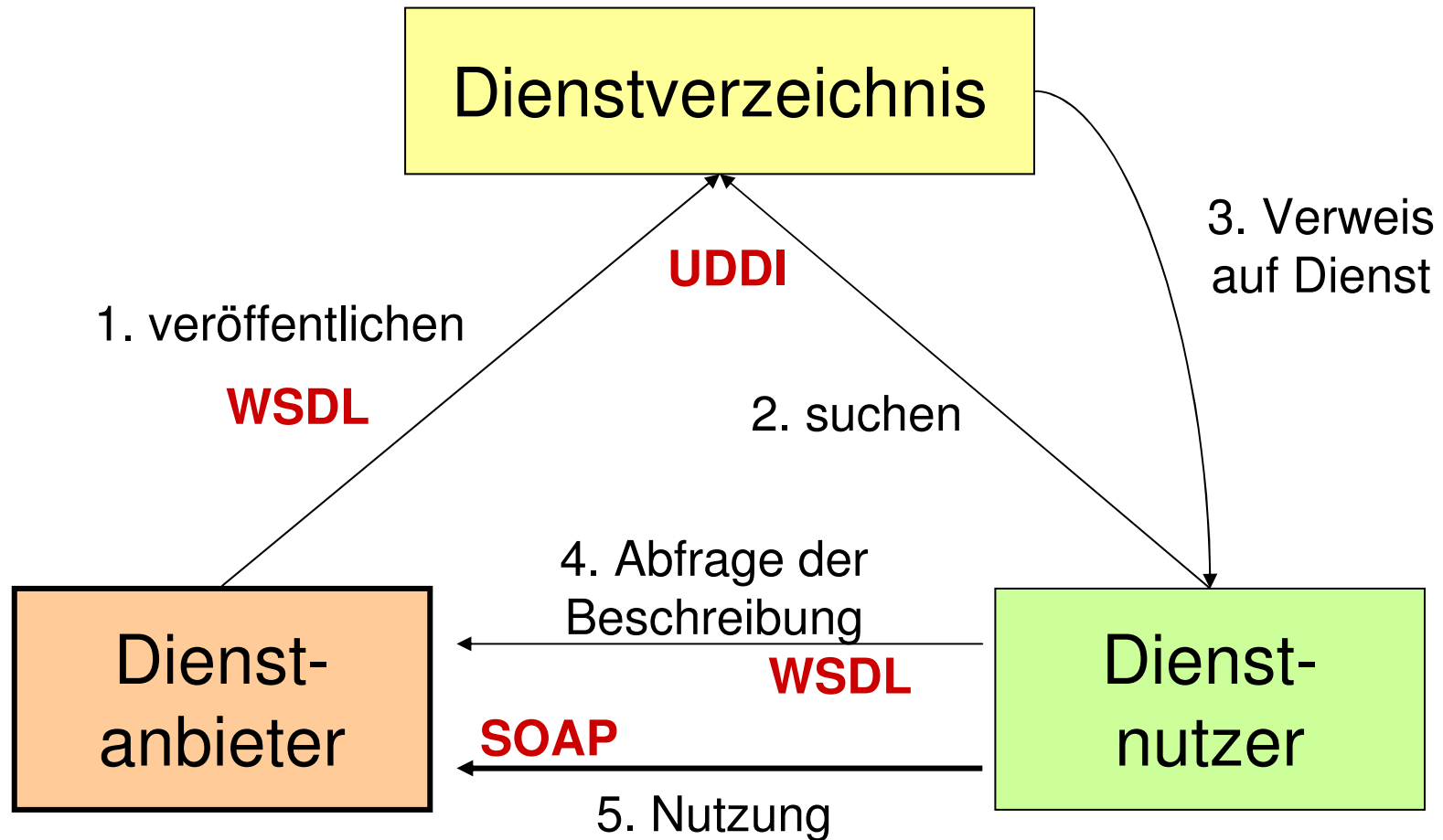


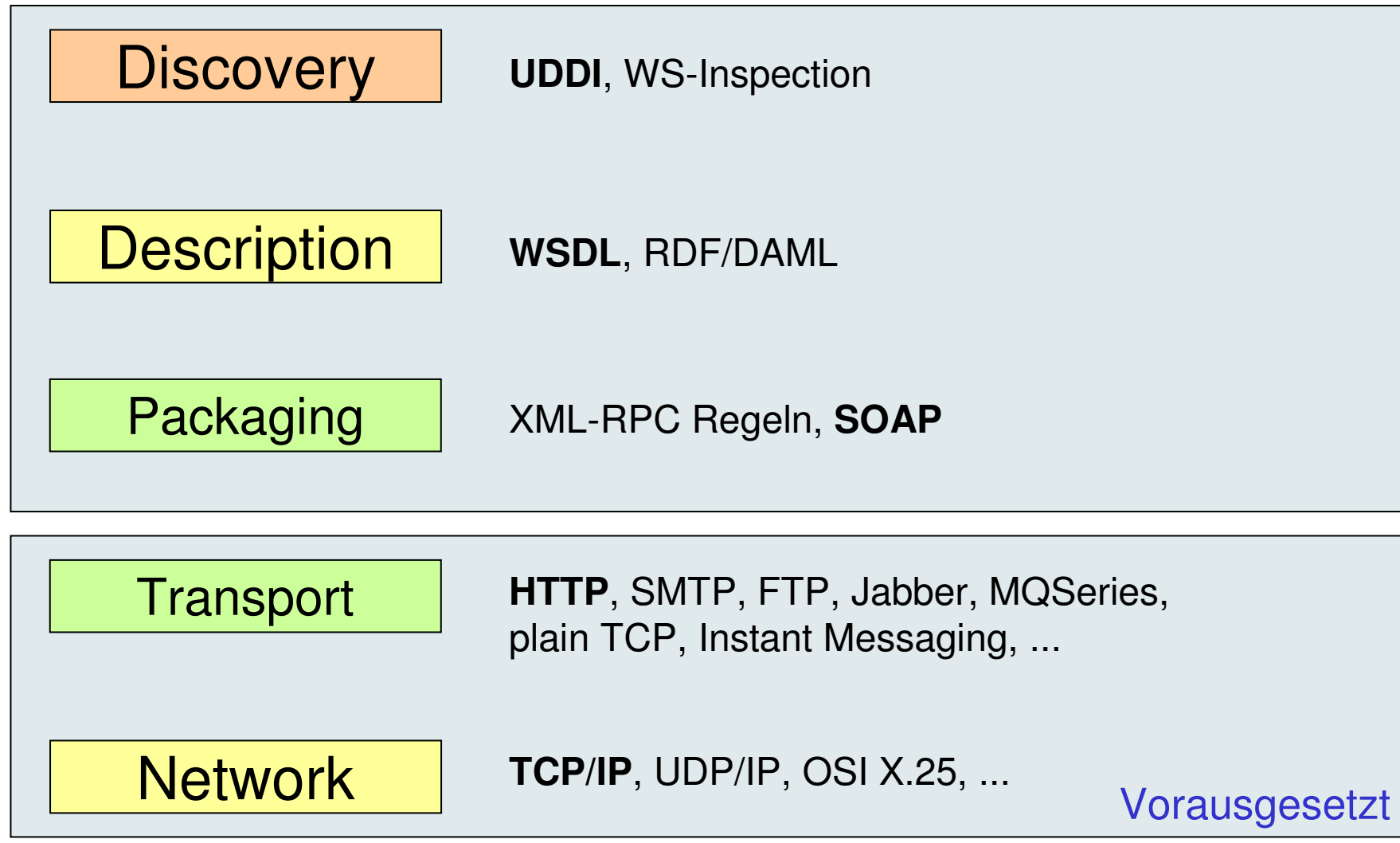


Abbildung von SOA auf Web Services





Der Technologie-Stack von Web Services (Ausschnitt)





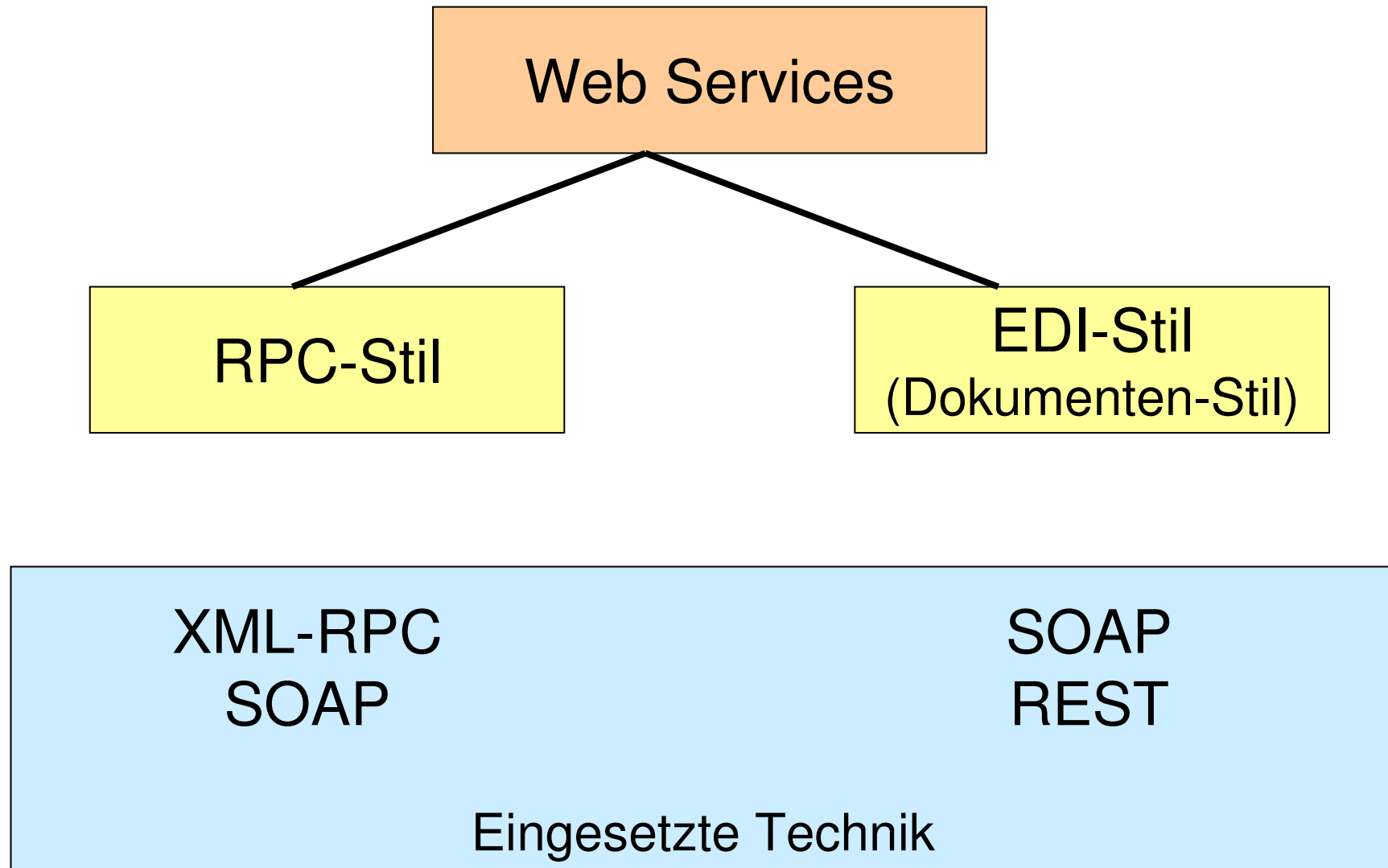
„Web Services“?



- „Service“? – Wer dient hier wem?
 - Bisher: WWW - Dialog zwischen Mensch und Anwendung
 - Nun: WS - Dialog zwischen Anwendungen
- Konsequenzen
 - Präzision und Strukturierung
 - Automatisierung
- Wieso der Name „Web“ Service?
 - Dienste sind nicht auf das WWW beschränkt!
 - Verschiedenste Transportprotokolle sind insbesondere bei SOAP möglich
 - Besserer Name: Net Services
 - Hat sich nicht durchgesetzt – WS war bereits zu etabliert
 - Anekdote: Bill Gates brachte „Web Services“ ins Spiel
 - Merke: WS = Eher ein Marketingname als ein technisch präziser Begriff



- Web Services: Die zwei Betriebsarten
 - **RPC-Stil** (RPC = *Remote Procedure Call*)
 - Übergabe einiger einfacher Datentypen
 - Rückgabe einfacher Datentypen, evtl. Fehlerbedingungen
 - Verarbeitung primär *on-line* (*Client* wartet auf Antwort)
 - **EDI- bzw. Dokumenten-Stil**
 - Austausch komplexer Datenstrukturen
 - Nur Dokumentenaustausch *on-line*, Verarbeitung (auch) *off-line*



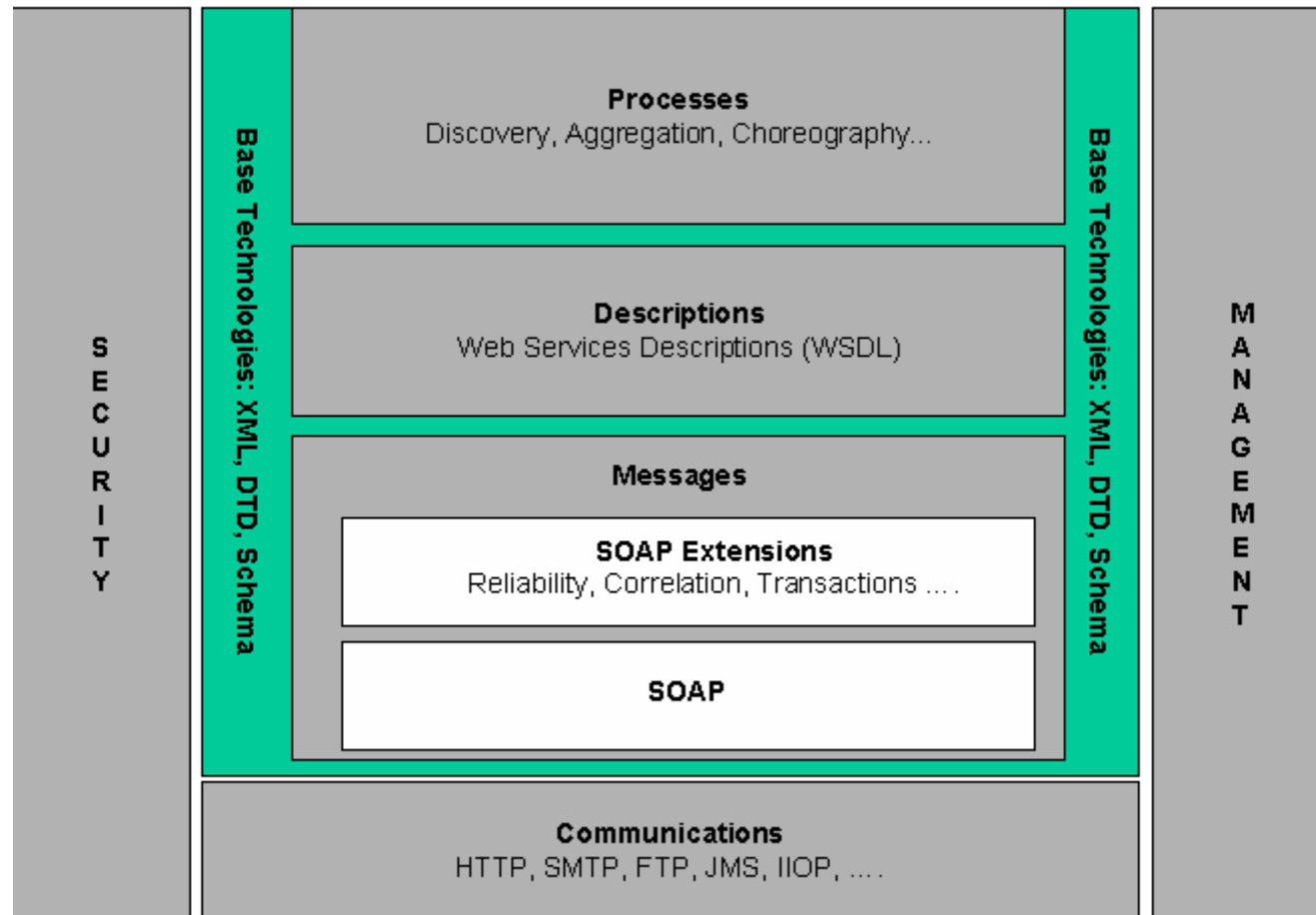


- **Vorteile von XML-RPC**
 - Einfach
 - Optimiert für RPC-Zwecke
 - Gut für *Inhouse*-Anwendungen
 - Kostengünstig realisierbar

- **Vorteile von SOAP**
 - Standardisiert (W3C)
 - Geeignet für RPC- und EDI-Stil
 - Grundlage weiterführender Standards (WSDL, UDDI)
 - Auch geeignet für internetweiten Einsatz
 - Erweiterbar



Ein WS-Architekturstapel



Quelle: Booth et al., W3C, 2004 (<http://www.w3.org/TR/2004/NOTE-ws-arch-20040211>)



Quelle: Melzer et al., Service-orientierte Architekturen mit Web Services, 3. Auflage, Spektrum Akad. V., 2008

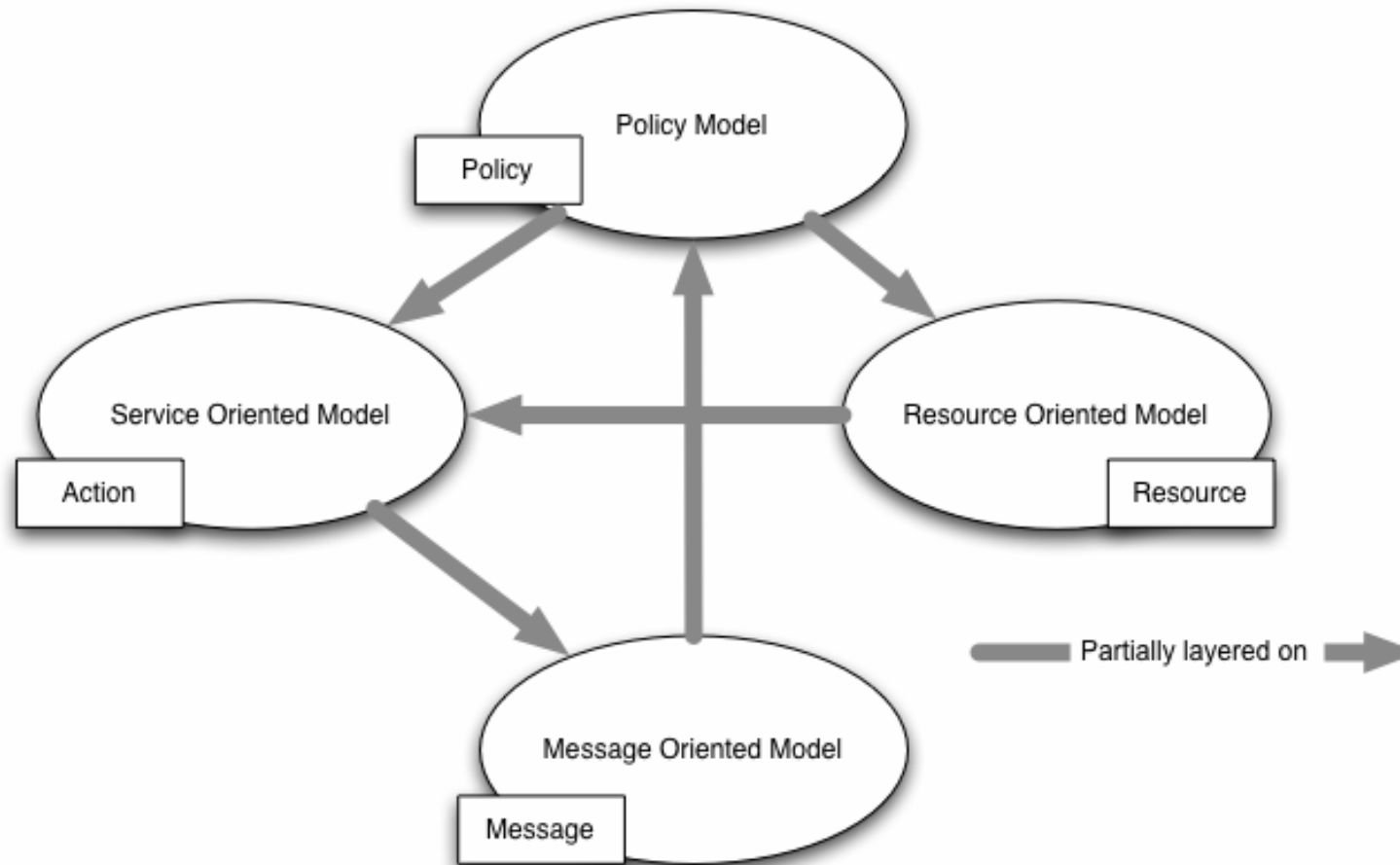


Entwicklungstrend

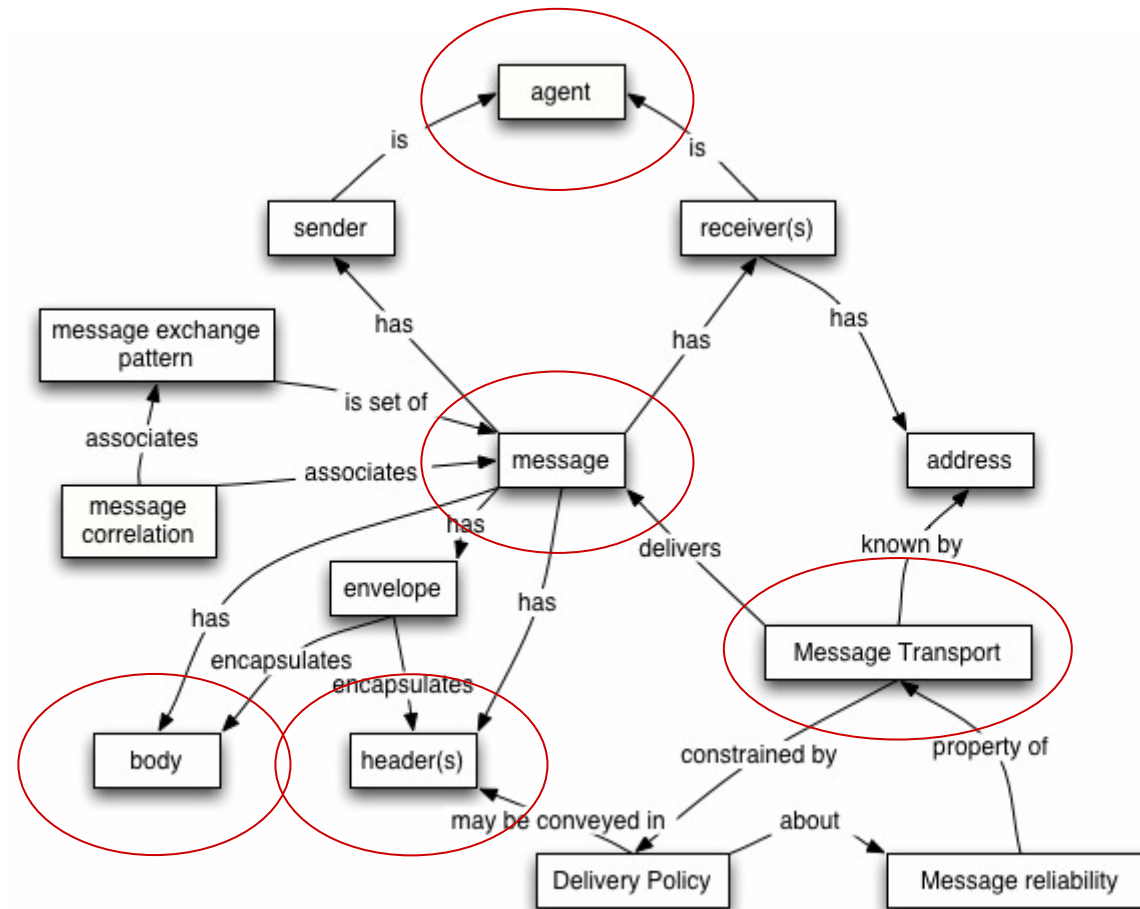
1. Enge Kopplung zwischen Anwendungen
 - Tiefe Integration; Ideal: Die Möglichkeiten integrierter Anwendungen
 - Ausdehnung des Konzepts "Prozedur-Aufruf" auf Anwendungs- und Rechnernetze
 - Techniken: CORBA, COM/DCOM, Suns RMI

2. Übergang
 - Beibehaltung des Konzepts "Prozedur-Aufruf" bei lockerer Kopplung
 - Ausgliederung des *Messaging* an separate Schicht (etwa HTTP)
 - Verpackung ("*marshalling*") der Aufrufs- und Rückgabeparameter mit Standardmethoden (XML)
 - Techniken: XML-RPC, SOAP (RPC-Stil)

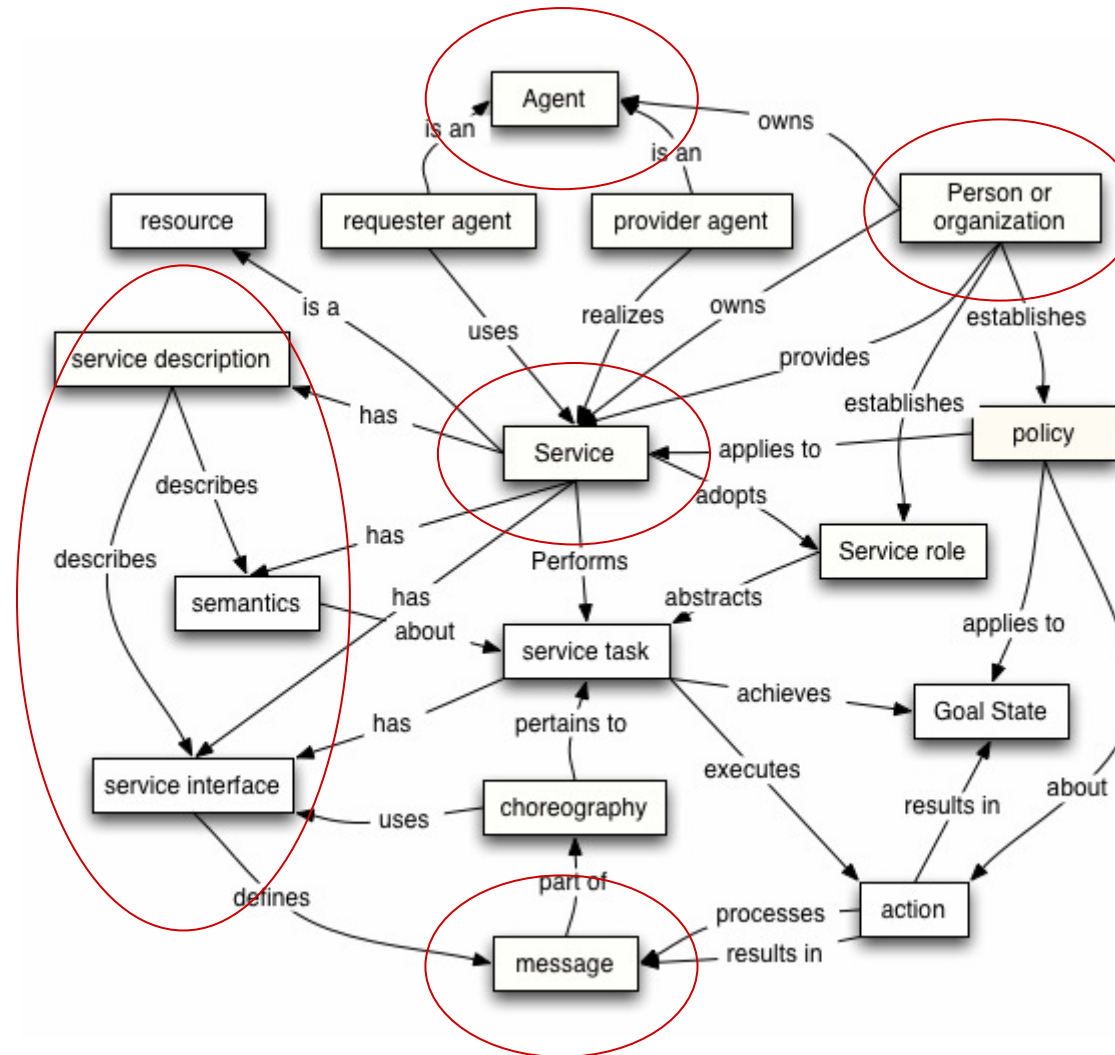
3. Lockere Kopplung zwischen Anwendungen
 - Robuste, fehlertolerante Anwendungsnetze
 - Das Konzept "Dokumentenaustausch" herrscht vor
 - "Aufruf" und Verarbeitung bzw. "Antwort" erfolgen meist asynchron
 - Techniken: SOAP (Dokumenten-Stil), REST



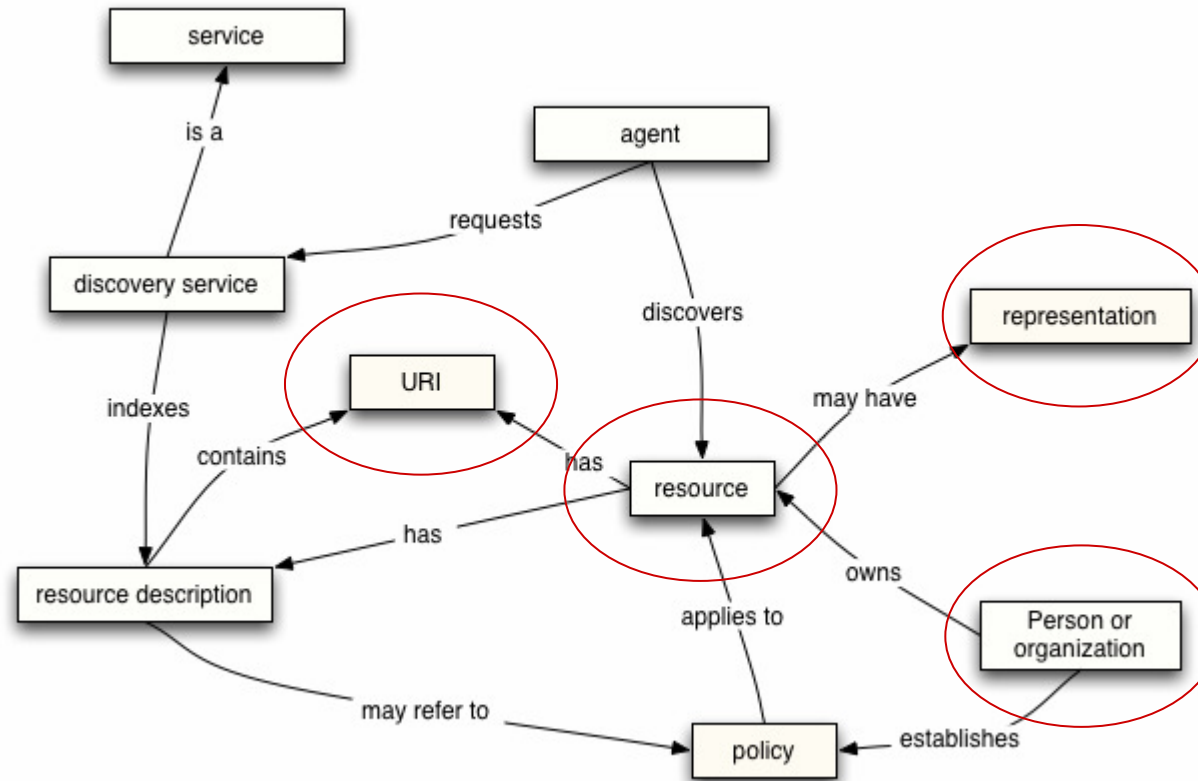
Quelle: Booth et al., W3C, 2004 (<http://www.w3.org/TR/2004/NOTE-ws-arch-20040211>)



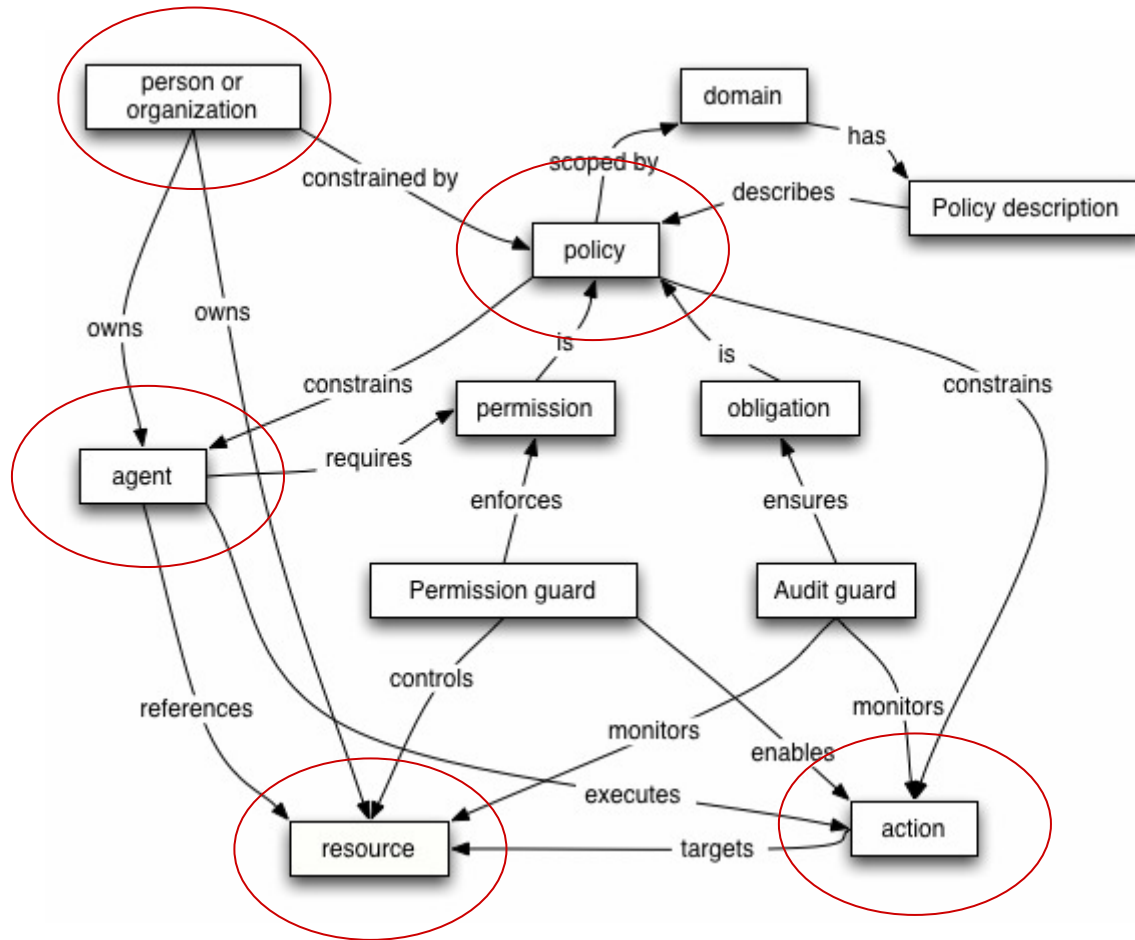
Quelle: Booth et al., W3C, 2004 (<http://www.w3.org/TR/2004/NOTE-ws-arch-20040211>)



Quelle: Booth et al., W3C, 2004 (<http://www.w3.org/TR/2004/NOTE-ws-arch-20040211>)



Quelle: Booth et al., W3C, 2004 (<http://www.w3.org/TR/2004/NOTE-ws-arch-20040211>)



Quelle: Booth et al., W3C, 2004 (<http://www.w3.org/TR/2004/NOTE-ws-arch-20040211>)