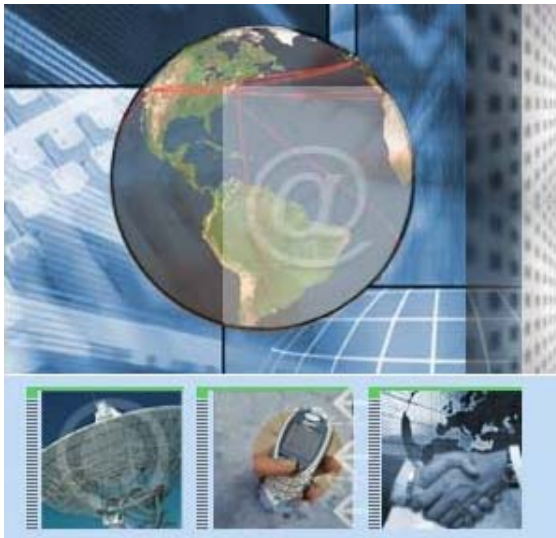


Ulrich M. Löwer und Tobias Hauser

Grundlagen und Standards für Web Services



DINI Jahrestagung 2004
am 29.09.2004
in Heilbronn



Überblick

- Grundlagen und Standards für Web Services
 - Ausgangslage
 - Service Oriented Architecture
 - Web Services Modell und Stapel
 - Basis-Standards
 - Weitere Standards
 - Bewertung
- Web Services für Portale
 - Definition eines Portals
 - Architektur – der Anknüpfungspunkt
 - WSRP – Ein Standard
 - Lösungen heute
 - Flexible Kopplung – das Ideal



Inhalt

- Ausgangslage
- Service Oriented Architecture
- Web Services Modell und Stapel
- Basis-Standards
- Weitere Standards
- Bewertung



Ausgangslage: Koppelung von Software

- Situation
 - Zunehmend viele Abläufe von Software unterstützt
 - Z.B. Office, Online Shopping, ERP, Verwaltung (Bücher, Studenten, Noten), etc.
- Komplikationen
 - Oft benötigte Daten oder Funktionen bereits in einer anderen Software (evtl. in anderer Organisation) vorhanden
 - Verknüpfung/Integration oft sehr aufwändig oder unmöglich (z. B. Cobol-Programm auf Mainframe an Webseite)
- Lösungsansätze

Grundidee: flexible Koppelung von Software über standardisierte Schnittstellen

 - Remoting (JRMII, DCOM – jetzt .NET Remoting)
 - CORBA
 - XML-RPC
 - Web Services



„Hype“ und Verwirrung rund um Web Services

Unzählige TLAs, 4LAs, etc.

- .NET
- BPEL
- ebXML
- NetWeaver
- OASIS
- RosettaNet
- SOAP
- UDDI
- W3C
- WebSphere
- WS-I
- WS-Security
- WSDL
- WSRP
- XML
- ...



Web Services bei großen Unternehmen

- Die drei größten Softwareanbieter richten ihre Softwarearchitekturen auf Web Services aus:

 Microsoft

– Microsoft: „Longhorns“ WinFX, die Zusammenführung von Win32 und .NET, beinhaltet eine Kommunikationsarchitektur mit dem Codenamen „Indigo“. Diese basiert vollständig auf Web Services

 IBM

– IBM WebSphere: Java-Applikationsserver mit Web Services

 SAP

– SAP: NetWeaver stellt die zentrale Plattform neuerer und aller künftiger SAP-Produkte dar. Über Web Services können alle Produkte und Komponenten miteinander gekoppelt werden

- Große „Dot-Com“-Gewinner bieten ihre Leistungen als Web Services an:

 amazon.com

– Amazon bietet von Kataloginfos bis zur Bestellabwicklung die wichtigsten Funktionalitäten Dritten zur Koppelung an

 Google

– Google bietet einen direkten Zugriff auf seine Suchfunktionen via Web Services zur Koppelung in andere Angebote

 A9

– Beispiel: A9 kombiniert Googles Suche im Web mit Amazons Suche in Büchern



Inhalt

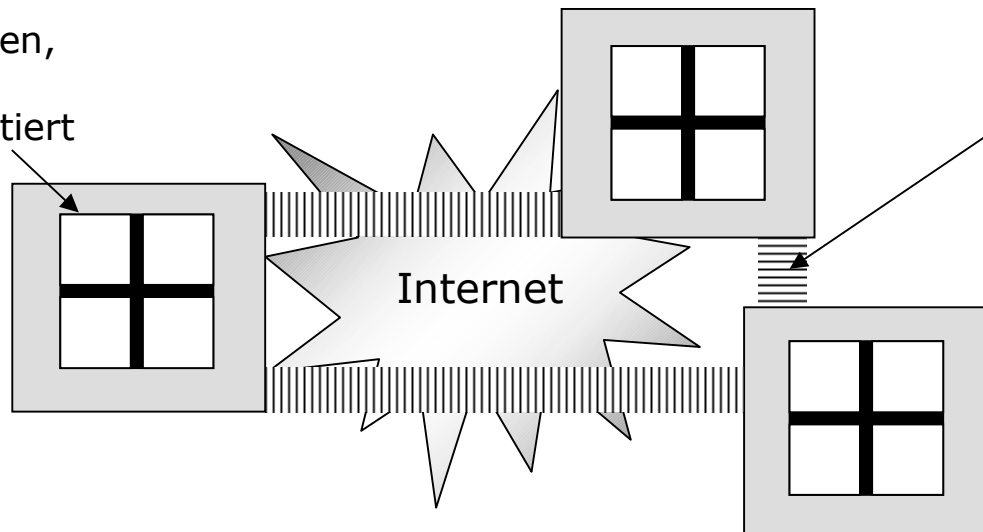
- Ausgangslage
- Service Oriented Architecture
- Web Services Modell und Stapel
- Basis-Standards
- Weitere Standards
- Bewertung



Object vs. Service Oriented Architecture

- **Object Oriented Architecture (OOA):**
Architektur für die enge Koppelung von Objekten, insbesondere für die interne Funktionalität von Softwarekomponenten
 - **Service Oriented Architecture (SOA):**
Architektur für die lose Koppelung von Komponenten, insbesondere für die verteilte Kommunikation von Komponenten
- Keine Ablösung, sondern Koexistenz beider Architekturen

Software-
komponenten,
meist
objektorientiert
auf einem
Rechner



Services als mehr
oder weniger
komplexe Bündel
von Methoden, die
über ein Netzwerk
angeboten werden

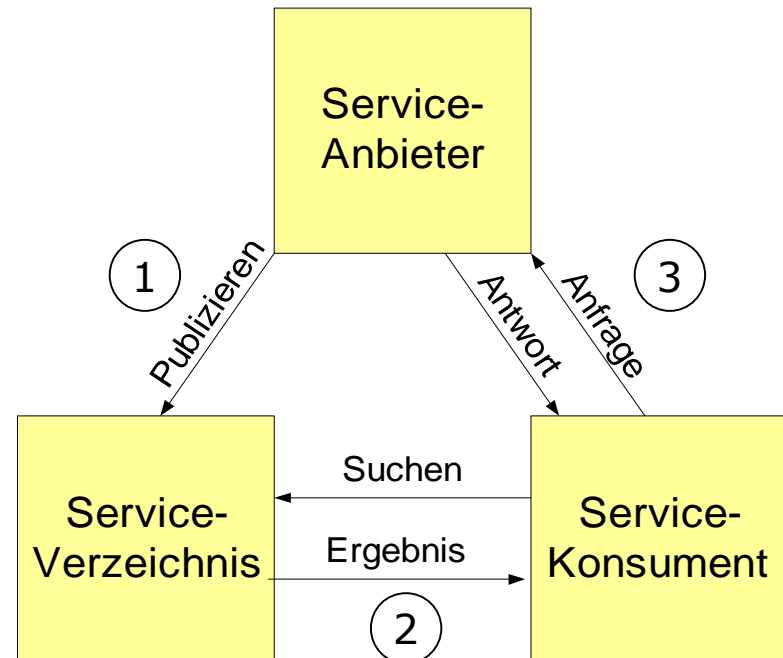
Architektur	Koppelung
OOA	Eng
SOA	Lose



Grundkonzept der SOA

Grundidee: weitgehend automatisiertes „Plug&Play“ von verteilten Softwarekomponenten

1. Anbieter publiziert seinen Service in einem Verzeichnis
2. Konsument sucht im Verzeichnis und findet Service des Anbieters
3. Konsument bindet den Service des Anbieters direkt in seine Software ein





Inhalt

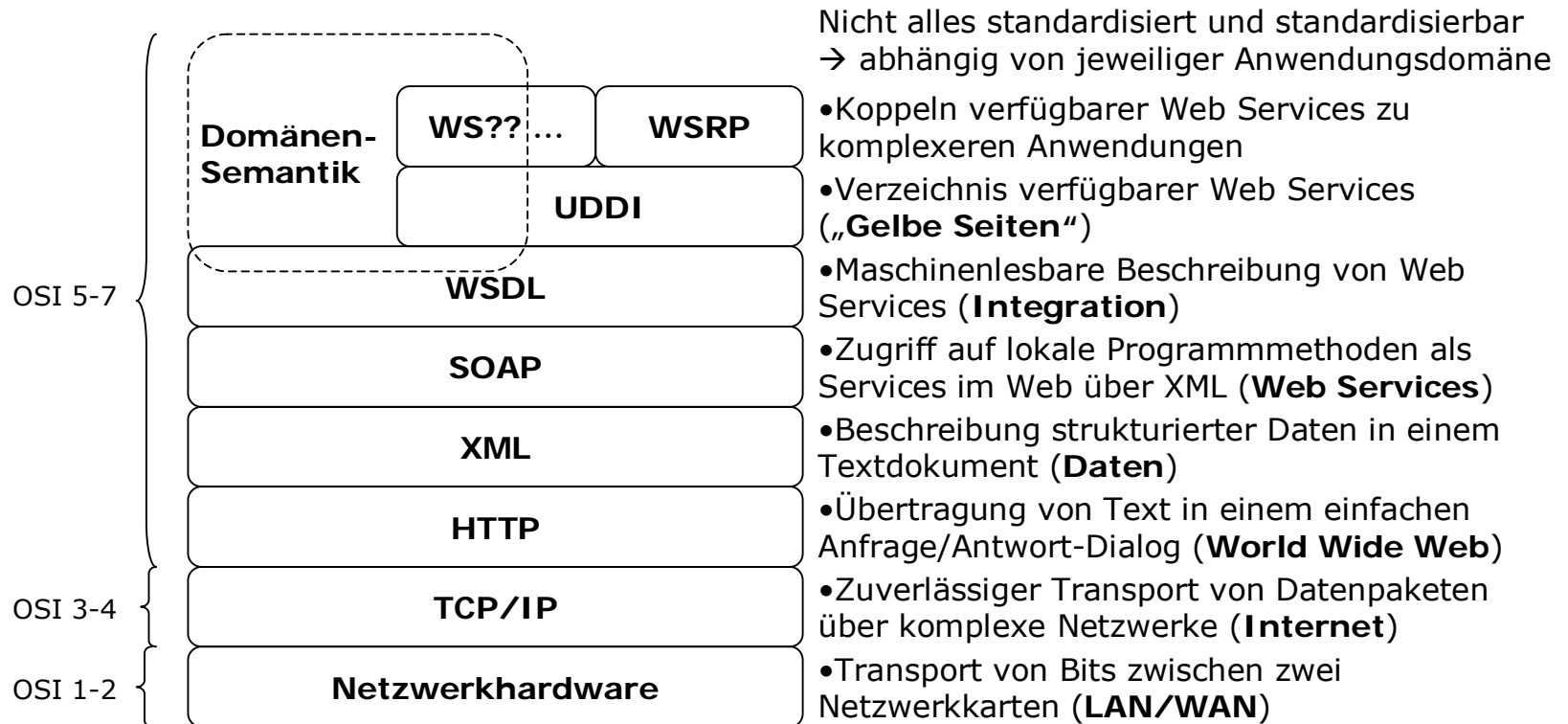
- Ausgangslage
- Service Oriented Architecture
- Web Services Modell und Stapel
- Basis-Standards
- Weitere Standards
- Bewertung



Web Services als konkrete SOA

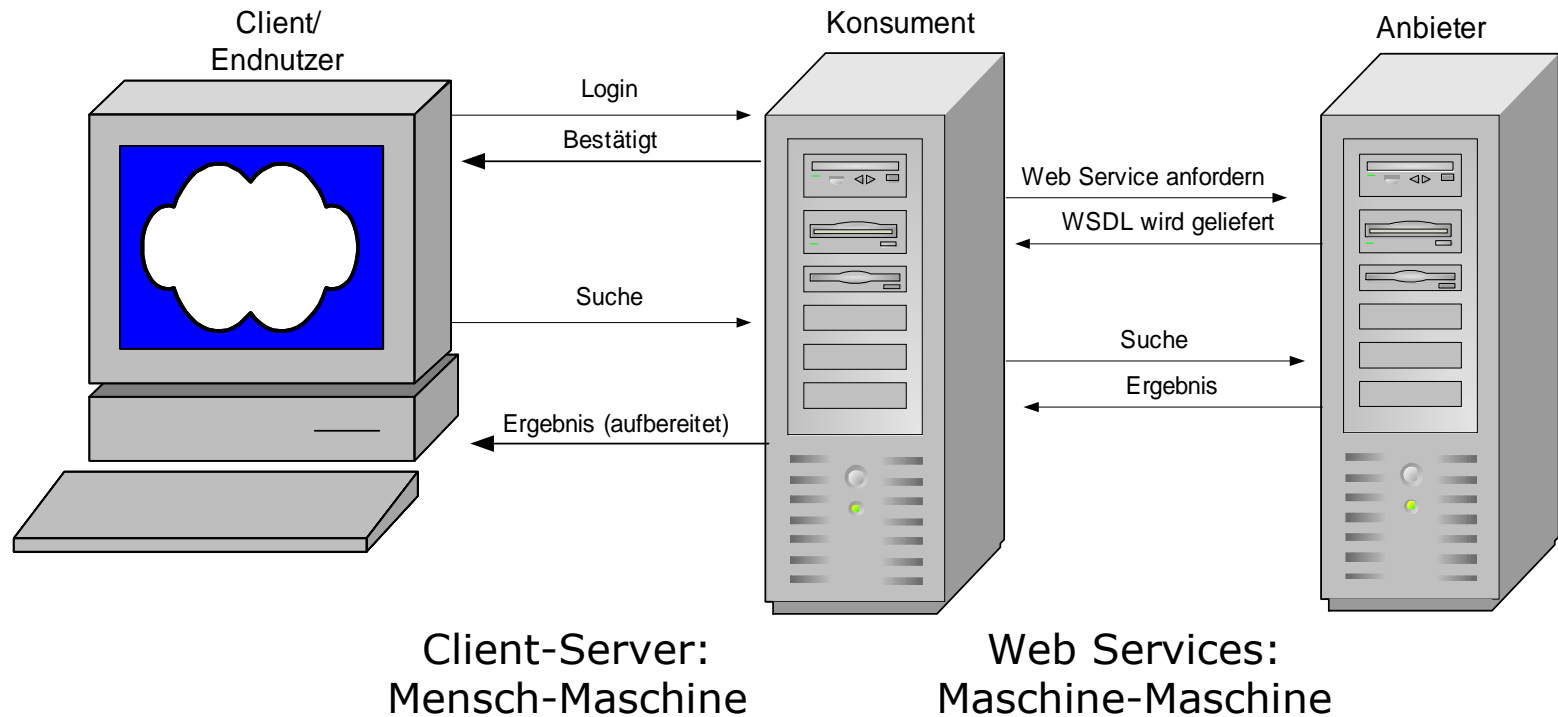
- Web Services sind eine konkrete Implementierung der Service Oriented Architecture
- Definitionen
 - “Program integration across application and organizational boundaries.” (Berners-Lee 2003)
 - Web Services sind ein Satz von Standards, der flexible Kommunikation zwischen Maschinen erlaubt
- Zentrale Spezifikationen
 - SOAP
 - WSDL
 - UDDI
- Viele weitere bauen darauf auf
 - SAML
 - WS-Security
 - BPEL
 - WSRP
 - ...

Der Web Services Stapel





Web Service-Modell mit Endnutzer





Inhalt

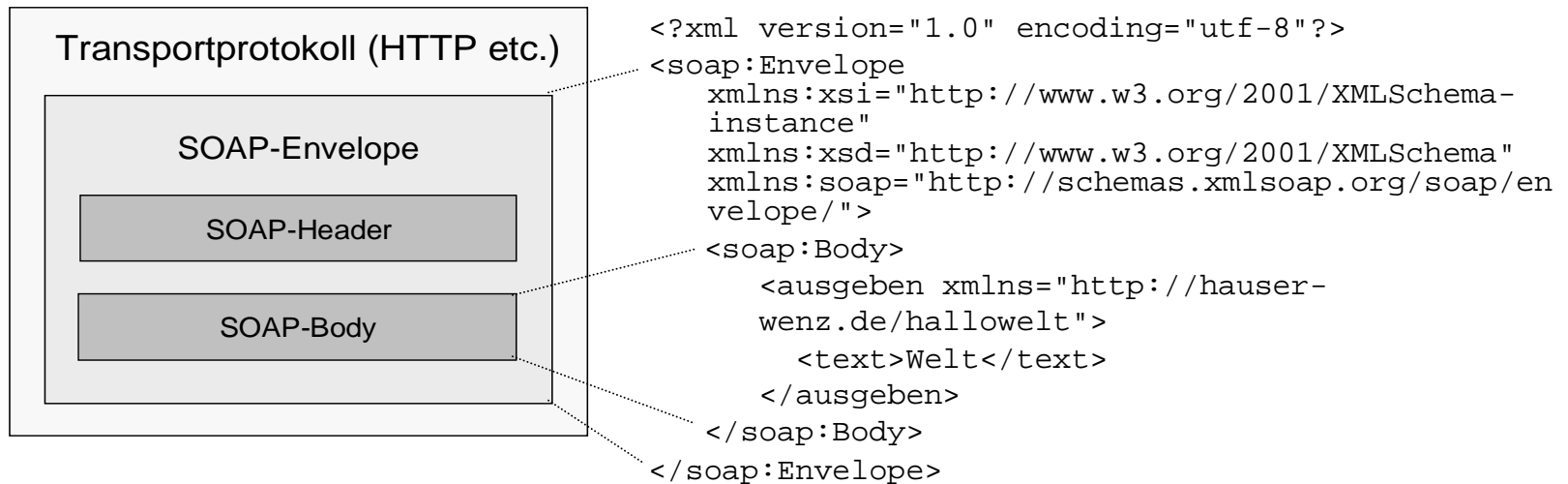
- Ausgangslage
- Service Oriented Architecture
- Web Services Modell und Stapel
- **Basis-Standards**
- Weitere Standards
- Bewertung



SOAP

- SOAP
 - SOAP hieß ursprünglich Simple Object Access Protocol
 - In SOAP 1.2 heißt SOAP einfach nur SOAP
- SOAP wird vom W3C verwaltet und wurde dort (hauptsächlich) von Microsoft und IBM gefördert
- Spezifikationen
 - Aktuelle Version ist SOAP 1.2 (<http://www.w3.org/TR/soap/>)
 - Meist implementiert ist aktuell 1.1 (<http://www.w3.org/TR/2000/NOTE-SOAP-20000508/>)
- Transport
 - SOAP wird meist über HTTP transportiert
 - Denkbar sind außerdem andere Protokolle wie z.B. SMTP

SOAP - Aufbau



- Der SOAP-Body enthält die eigentliche Nachricht mit Methodenaufruf bzw. Antwort
- Der SOAP-Header enthält zusätzliche Informationen beispielsweise zur Authentifizierung
- Der SOAP-Envelope (-Umschlag) schließt beide ein
- Er wird wiederum im Transportprotokoll eingeschlossen
- „Hallo Welt“-Beispiel ohne Header und Transportprotokoll



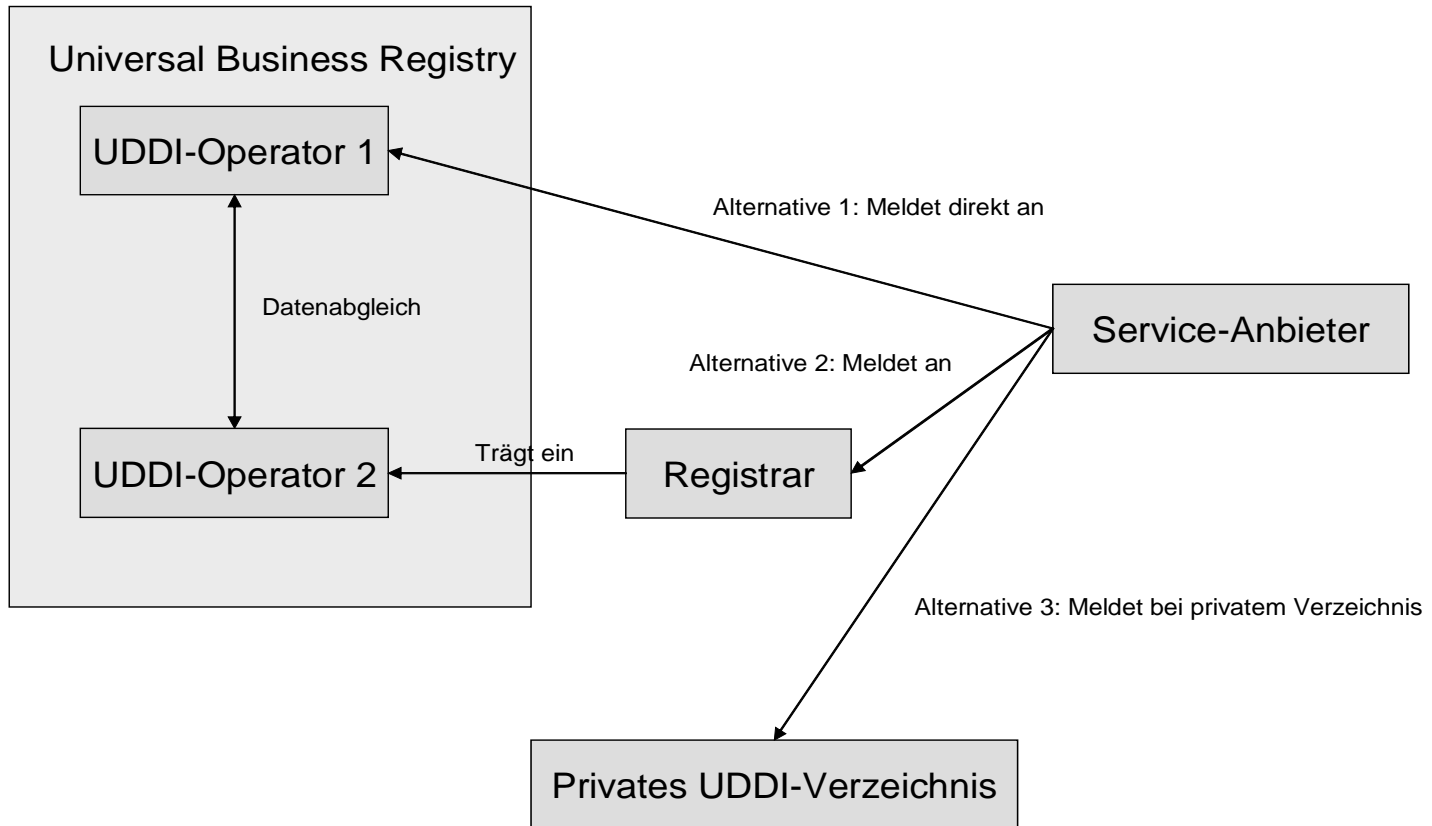
WSDL

- Web Service Description Language
- Auch WSDL wird vom W3C verwaltet und wurde dort (hauptsächlich) von Microsoft und IBM gefördert
- Die aktuelle Version ist 1.1 (<http://www.w3.org/TR/wsdl.html>), an 1.2 wird gearbeitet
- Basiert auf XML und beschreibt die Methoden und Datentypen von Web Services
- WSDL hilft damit der Implementierung, die SOAP-Nachricht korrekt zu interpretieren

UDDI

- Universal Description, Discovery and Integration (of Web Services)
- Ursprünglich UDDI-Projekt unter <http://www.uddi.org/>, wiederum von Microsoft und IBM
- Wird mittlerweile von OASIS verwaltet und weiterentwickelt, <http://www.uddi.org/> gibt es aber noch
- Standard für Web Service-Verzeichnisse und die Schnittstellen (APIs), die diese Verzeichnisse zur Verfügung stellen
- Aktuell ist Version 3, implementiert oft noch 2.
- Einige größere Software-Firmen bieten UDDI-Verzeichnisse (sie sind UDDI-Operatoren)
 - Microsoft: <http://uddi.microsoft.com/>
 - IBM: <https://uddi.ibm.com/ubr/registry.html>
 - SAP: <http://uddi.sap.com/>
 - HP: <http://uddi.hp.com/>
 - NTT: <http://www.ntt.com/uddi/>

UDDI – Aufbau





Inhalt

- Ausgangslage
- Service Oriented Architecture
- Web Services Modell und Stapel
- Basis-Standards
- Weitere Standards
- Bewertung

Weitere Standards für Web Services

- Sicherheit
 - Sicherheit als zentrale Voraussetzung für Web Services
 - Standardisierung noch nicht abgeschlossen, zeichnet sich aber ab
 - Wichtige Spezifikationen
 - XML Signature
 - XML Encryption
 - SAML (Secure Assertion Markup Language)
 - WS-Security
- Transaktionen und Prozesse
 - Kombination einer Menge von Web Services zu komplexeren Anwendungen
 - Vielzahl konkurrierender Ansätze, teilweise mit speziellem Einsatzbereich
 - Überschneidungen und Komplementaritäten erschweren Vergleichbarkeit
 - Wichtige Spezifikationen
 - BPEL(4WS) (Business Process Execution Language for Web Services)
 - BPSS (Business Process Specification Schema)
 - WS-CAF (Web Services Composite Application Framework)
 - WS-C+T (Web Services Coordination and Transaction)
 - WSCI (Web Services Choreography Interface)

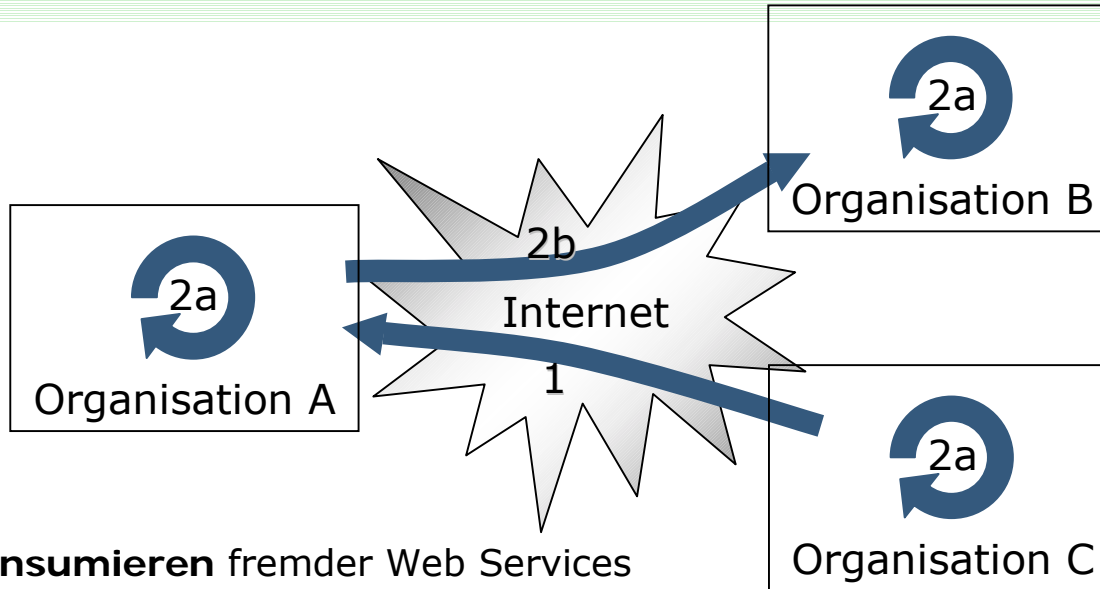


Inhalt

- Ausgangslage
- Service Oriented Architecture
- Web Services Modell und Stapel
- Basis-Standards
- Weitere Standards
- Bewertung



Prinzipielle Einsatzbereiche von Web Services



1. **Konsumieren** fremder Web Services

2a. **Interne** Nutzung eigener Web Services

2b. **Anbieten** eigener Web Services

- Unternehmen nutzen einerseits fremde Web Services, bieten andererseits selbst ihre Web Services an
- Die klare Trennung zwischen Konsumenten und Lieferanten von Web Services wird verwischt
- Wirtschaftliche Fragstellung: „Selbsterstellung oder Fremdbezug?“
- Vergleiche Outsourcing-Diskussionen



Fazit zu Web Services

Nachteile

- Web Services haben manchmal nicht die beste Performance
- Umdenken ist nötig; z.B. müssen Firewalls angepasst werden
- Der Bereich Sicherheit ist nicht besonders weit entwickelt
- Der Bereich Transaktionen und Prozesse ist noch weniger entwickelt
- Teilweise viele konkurrierende Standardisierungsvorhaben

Vorteile

- + Die wichtigsten Protokolle sind standardisiert
- + Sie basieren auf XML:
 - + Die Nachrichten sind von Menschen lesbar
 - + Viele weitere Standards stehen zur Verfügung
- + Mittlerweile gibt es viele Implementierungen
- + Die Software-Industrie setzt auf Web Services
- + Standardisierungsprozesse sehr offen, genaue Beobachtung und aktive Beteiligung möglich

→ Auch wenn einige Standardisierungsvorhaben und Implementierungen noch nicht vollständig sind, führt mittelfristig an Web Services kein Weg mehr vorbei!