



Definition

A „Web Service“ is an arbitrary piece of self-contained, descriptive logic

packaged in such a way that it can be **located** and **invoked programmatically** over the internet

using a set of low overhead, open standard network and application protocols

(i.e. UDDI, SOAP, XML, etc.)

Positionierung

- ☞ Web-Services sind eine Evolution der existierenden verteilten Modelle
- ☞ Sie beinhalten die Vorteile von Client/Server- und netzbasiertem Computing
- ☞ Durch die schwache Kopplung differenzieren sich Web-Services von herkömmlichen Modellen

3

Verschiedene Aspekte von Web Services

Web Services können aus verschiedenen Blickwinkeln betrachtet werden:

- Aus der Sicht des Internet Nutzers
- Aus der Sicht verteilter Systeme
Web Services im Vergleich zu
 - RMI
 - CORBA
 - DCOM
- Aus der .Net Sicht

4

Stark gekoppelte Integration

☞ Beispiele

- CORBA,
- RMI, DCOM,
- Distributed Smalltalk

☞ Grundlage

- Wechsel von 2- zu 3-tier Architekturen
- Austausch von Funktionsargumenten

☞ Problem

- Integrationstechniken haben sich als zu **unflexibel** für eine Interaktion über Firmengrenzen hinaus herausgestellt.

5

Ausweg: Schwach gekoppelte Integration

☞ Web-Services sind ein Beispiel für schwach gekoppelte Integration

☞ Programmiermodell:

- Service-orientiert
- Austausch von (XML-) Dokumenten

☞ Das grundlegende Protokoll ist definiert,

☞ Interfaces sind jedoch so flexibel, dass jede Seite Änderungen vornehmen kann, ohne die Interaktion zu zerstören

6

Unabhängigkeit

- Interfaces, werden durch den Austausch von XML-Dokumenten realisiert.
- Dadurch sind Web-Services
 - ✍ sprach-unabhängig
 - ✍ plattform-unabhängig
 - ✍ objektmodell-unabhängig.

7

Features

- ✍ Web Services versprechen einen *straightforward* und *Low Entry Cost* Mechanismus für System-zu-System Interaktion zwischen Firmen
- ✍ **Wiederverwendbarkeit:**
Bestehende Anwendungen und Komponenten können mit einem Web-Service *Wrapper* versehen werden und somit wiederverwendet werden.

8

SOAP

- ✍ Simple Objekt Access Protocol
- ✍ Früher: RPC over HTTP
- ✍ SOAP ist ein neues Protokoll für den Austausch von XML-Dokumenten über HTTP
- ✍ SOAP 1.1 definiert 2 XML-Schemata
 - **Envelope**
 - **SOAP Encoding**
- ✍ Messaging Style
 - RPC/encoded
 - Document/literal

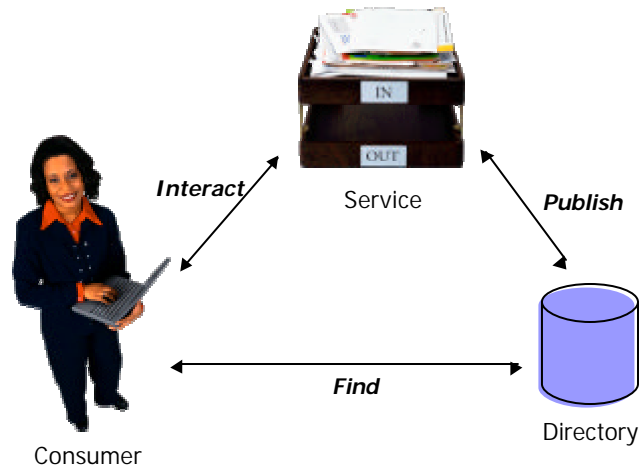
9

UDDI

- ✍ UDDI: Universal Description, Discovery, and Integration
- ✍ UDDI Server: Online Verzeichniss, das folgende Web Service Meta Informationen enthalten kann:
 - Beschreibung und Daten eines Business
 - Beschreibung des Service
 - Beschreibung des Interfaces
 - Endpoint (Physikalischer Ort des Service)
- ✍ UDDI ist derzeit ein Design-Time Directory
- ✍ Zugriff auf Server über Browser oder SOAP Client

10

Publish, Find, and Interact



11

Grundlegende Technologien

Publish & Find	UDDI, WS-Inspection
Interact	SOAP, SwA, ebXML Messaging Service
Define	XML, WSDL, XSLT
Connect	IP, DNS, URL, HTTP, HTTPS

12

WSDL

- ⌘ XML-basierte *Vertrags*-Sprache
- ⌘ De-facto Standards, wurde von IBM und Microsoft entwickelt
- ⌘ Definiert einen Standardmechanismus, um Nachrichten, die ein Web Service akzeptiert/generiert, zu dokumentieren
- ⌘ Inhalt einer WSDL Datei
 - Typen
 - Nachrichten
 - PortTypes
 - Operationen
 - Bindings
 - Ports
 - Services
 - ...

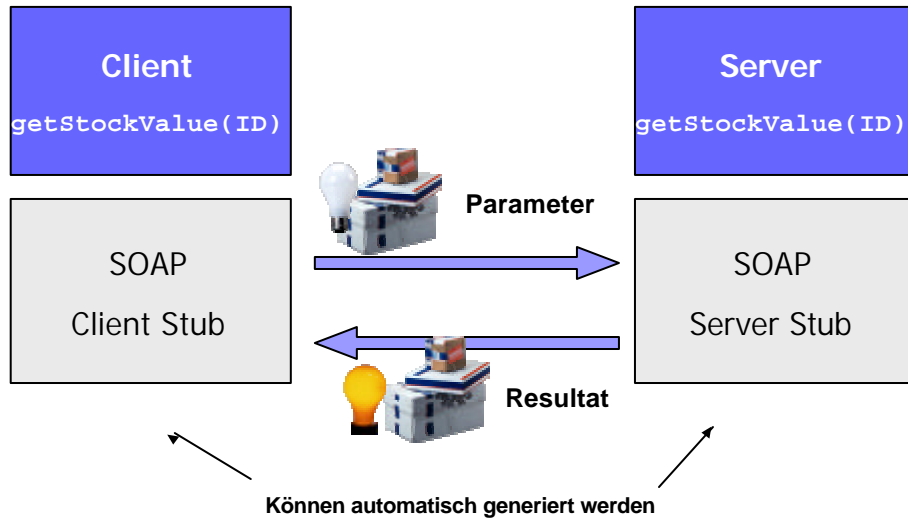
13

Inhalt einer WSDL Datei Mapping auf Java Artefakte

- <types>**
 - ⌘ Deklaration aller nicht-eingebauten Datentypen, die vom Web-Service benutzt werden, z.B. Arrays
- <message>**
 - ⌘ Nachricht, die zwischen Aufrufer und Service transportiert wird.
 - ⌘ Kann aus mehreren `parts` bestehen, welche Argumenten von Methoden entsprechen
- <portType>**
 - ⌘ Satz von `operations`, wobei eine `operation` einer Methode entspricht.
 - ⌘ Ein `portType` entspricht einem Interface oder einer Klasse
- <binding>**
 - ⌘ Ein `portType`, der auf Basis eines bestimmten Protokolls (z.B. SOAP, HTTP get/post) implementiert ist
 - ⌘ Für jedes Protokoll wird ein `binding` benötigt
- <service>**
 - ⌘ Deklaration Satz von Ports, wobei jeder Port einem `binding` an einem bestimmten Endpoint entspricht

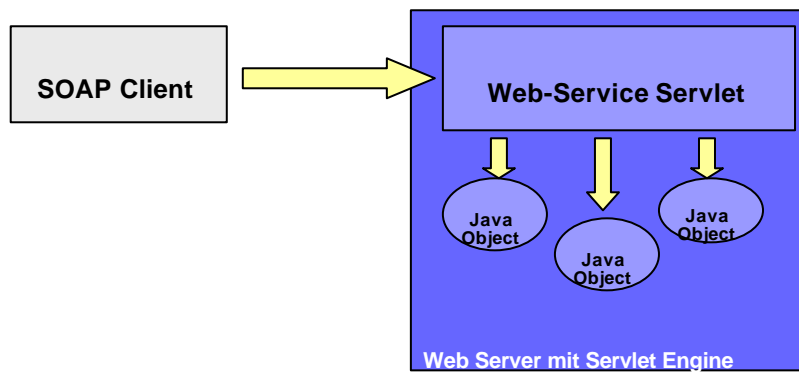
14

Web Services Architektur



15

Integration in Servlet Engines



16

Herausforderungen für Web Service Entwickler

- ✍ How do you seamlessly integrate when vendors typically deliver proprietary extensions and value-adds to their products/tools in order to differentiate their products and gain a competitive advantage?
 - o For example, there are differences in the manner that the Microsoft SOAP Toolkit 2.0 and the IBM Web Services Toolkit (WSTK) implement SOAP. This can cause challenges when a Microsoft SOAP client attempts to communicate with a WSTK server.
- ✍ How do you identify the type of device being used to access a service?

Quelle: Joseph M. Curran, Accenture

17

Herausforderungen für Service Provider

- ✍ How do you enable/enforce service level agreements and licenses existing between the consumer and the service provider?
- ✍ How do you charge for services and perform the appropriate metering and accounting?
- ✍ What framework do you utilize to deliver services in a secure, reliable, and available fashion?

Quelle: Joseph M. Curran, Accenture

18

Herausforderungen für Service Consumer

- ✍ How do ensure a web service is certified from a trustworthy source?
- ✍ How so you translate data to ensure it is the proper format for interfacing?

Quelle: Joseph M. Curran, Accenture

19

Herausforderungen für Web Service Assembler

- ✍ How do you assemble multiple discrete Web Services into a composite business service?
- ✍ How do you supply a personalized experience to your consumer when you are dependent on other multiple Web Services outside of your control?

20

Herausforderungen für Web Service Utilities

- ✎ How do you identify web service users (i.e. end user, a business, or another web service) and control the security policies, privacy policies, and business policies associated with that user?
- ✎ How will the network latency between disparate services be addressed in a transparent manner?
- ✎ How do you deliver a common data model for web services
- ✎ How do you share context (i.e. identity of consumer, location of consumer, etc.) across various Web Services?

Quelle: Joseph M. Curran, Accenture

21

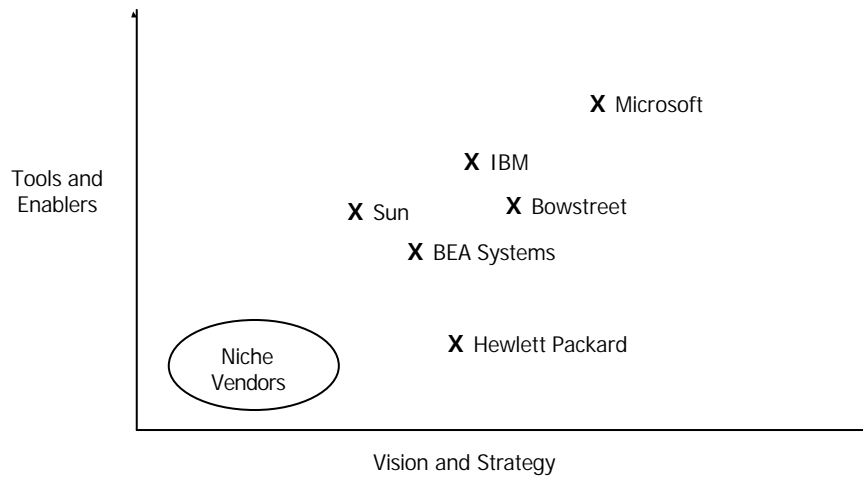
Web Services Markt

Marktteilnehmer befinden sich in verschiedenen Phasen

- Definition von Strategien und Ankündigungen von zukünftigen Produkten
- Angebot von Web Service Development und Deployment Werkzeugen
- Kollaboration mit Standardisierungsgremien und Unterstützung von Spezifikationen

22

Markteilnehmer



Quelle: Joseph M. Curran, Accenture

23

Weiterführende Informationen

- ☞ Glue: The Mind Electric
<http://www.themindelectric.com>
- ☞ XMethods : Öffentlich verfügbare Web Services
<http://www.xmethods.net>
- ☞ Java Web Service Tutorial/Web Services Developer Pack
<http://www.javasoft.com>
- ☞ BEA Systems
<http://www.beasys.com>
- ☞ Bowstreet
<http://www.bowstreet.com>
- ☞ Hewlett Packard Web Service Platform
<http://www.hp.com/go/webservices>
- ☞ IBM's Web Service Zone
<http://www.ibm.com/developerworks/webservices>
- ☞ Lutris Technologies
<http://www.lutris.com>
- ☞ Microsoft's .NET Homepage
<http://www.microsoft.com/net>
- ☞ Sun's Open Net Environment Homepage
<http://www.sun.com/sunone>

24

Standardisierungsgremien

- ✍ Business Process Management Initiative
<http://www.bpmi.org>
- ✍ ebXML Organization
<http://www.ebxml.org>
- ✍ OASIS Organization
<http://www.oasis-open.org/>
- ✍ UDDI Organization
<http://www.uddi.org/>
- ✍ World Wide Web Consortium (W3C)
<http://www.w3.org>

25

Vorstellung von GLUE

 THE MIND
ELECTRIC



26