

# Web Services – Eine Übersicht

Jörn Clausen

joern@TechFak.Uni-Bielefeld.DE

# Übersicht

- Was sind Web Services?
- XML-RPC und SOAP
- WSDL und UDDI
- Wo können wir Web Services einsetzen?

# Definition 1

A Web service is a software system identified by a URI, whose public interfaces and bindings are defined and described using XML. Its definition can be discovered by other software systems. These systems may then interact with the Web service in a manner prescribed by its definition, using XML based messages conveyed by internet protocols.

*Web Services Architecture*

Web Services Architecture Working Group

W3C

## Definition 2

At a minimum, [...] a Web service is any piece of software that makes itself available over the Internet and uses a standardized XML messaging system.

*Top Ten FAQs for Web Services*

Ethan Cerami

[webservices.xml.com](http://webservices.xml.com)

## Definition 3

Web services, in the general meaning of the term, are services offered via the Web.

[...]

Web services and consumers of Web services are typically businesses, making Web services predominantly business-to-business (B-to-B) transactions. An enterprise can be the provider of Web services and also the consumer of other Web services.

*The Java Web Services Tutorial*  
Sun Microsystems

# Definition 4

A Web service is a collection of functions that are packaged as a single entity and published to the network for use by other programs. Web services are building blocks for creating open distributed systems, and allow companies and individuals to quickly and cheaply make their digital assets available worldwide.

*The Web services (r)evolution*

Graham Glass

CEO The Mind Electric

## Definition 5

Web services let applications share data, and—more powerfully—invoke capabilities from other applications without regard to how those applications were built, what operating system or platform they run on, and what devices are used to access them. Although Web services remain independent of each other, they can loosely link themselves into a collaborating group that performs a particular task.

*What Are Web Services?*  
Microsoft Corporation

# Eigenschaften von Web Services

- Web Services ...
  - sind Software-Systeme
  - tauschen untereinander Daten aus
  - können gleichzeitig Anbieter und Konsument von Daten sein
  - können zu größeren Systemen kombiniert werden
  - können verschiedenen Sprachen/Betriebssysteme verwenden
  - verwenden XML zum Austausch von Daten
  - sind auffindbar
- ...haben wenig mit dem WWW zu tun



# verteilte Applikationen

- Sun RPC (Remote Procedure Calls)
  - XDR statt XML
  - `rpcgen`, C-lastig
  - Client und Server „aus einem Guß“
- RMI (Remote Method Invocation)
  - Java-spezifisch
  - mißbraucht HTTP als Transport-Protokoll
- CORBA (Common Object Request Broker Architecture)
  - groß, schwerfällig
  - hat sich nicht wirklich durchgesetzt

# XML-RPC

- entstanden 1998/99 im SOAP-Umfeld
- Spezifikation: 5 Seiten
- Transport-Protokoll: HTTP
- Implementierungen in vielen gängigen Sprachen
- relativ wenige Anwender, z.B. O'Reillys *Meerkat*

# Daten in XML-RPC

- atomare Datentypen:

```
<int>1234</int>
```

```
<double>-12.345678</double>
```

```
<string>Laurel & Hardy</string>
```

- arrays:

```
<array>
```

```
  <data>
```

```
    <value><string>Huey</string></value>
```

```
    <value><string>Dewey</string></value>
```

```
    <value><string>Louie</string></value>
```

```
    <value><int>3</int></value>
```

```
  </data>
```

```
</array>
```

# Daten in XML-RPC, cont.

- structs/hashes:

```
<struct>
  <member>
    <name>Scrooge</name>
    <value><string>duck</string></value>
  </member>
  <member>
    <name>Goofy</name>
    <value><string>dog</string></value>
  </member>
</struct>
```

- Kombinationen (Array von Structs)

# XML-RPC, Client-Seite

- Dictionary-Client in Perl

```
use RPC::XML::Client;

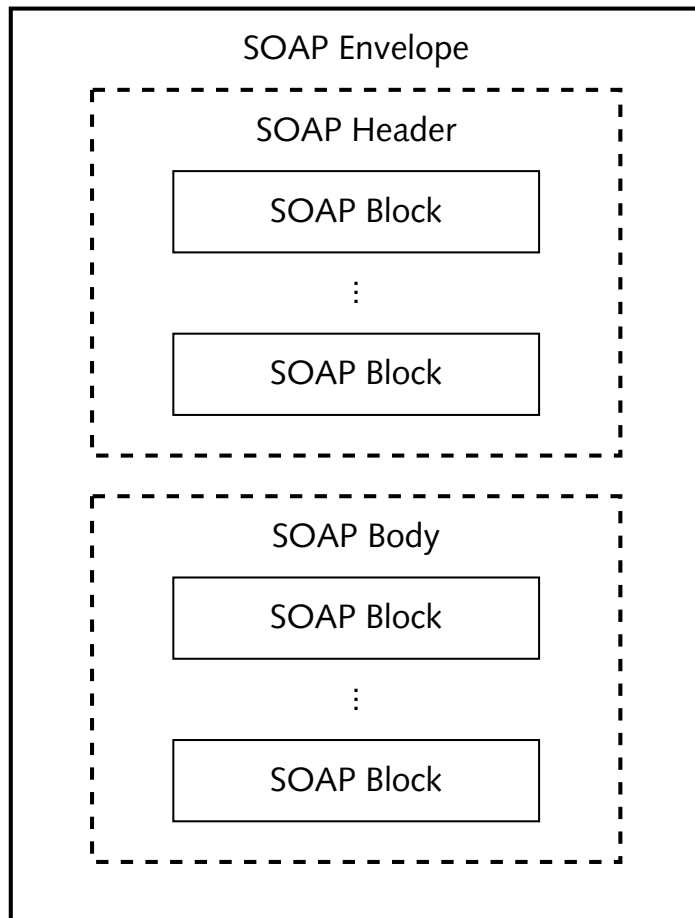
$client = RPC::XML::Client->
    new( 'http://babelfish.org/dictionary' );
$response = $client->
    send_request( 'translator.german2english' ,
                 'Schmusedecke' );
print $response->value, "\n";
```

- Wo ist XML geblieben?

# SOAP

- Version 1.2 seit 24.6.2003 „Recommendation“ des W3C
- früher „Simple Object Access Protocol“, heute kein Akronym mehr
- Messaging-Protokoll, RPC nur ein Spezialfall
- nicht an bestimmtes Transport-Protokoll gebunden
- Spezifikationen für HTTP, EMail, Jabber, BEEP
- bessere Typisierung durch W3C XML Schema

# SOAP Message



- zwei Teile: Header und Body
- unterteilt in Blöcke
- initial sender, intermediary, ultimate receiver
- Body wird vom Empfänger gelesen
- Header kann von intermediaries gelesen und verändert werden

# Services deklarieren

- WSDL – Web Services Description Language
- beschreibt einen Web Service
  - Datentypen
  - Methoden/Prozeduren
  - Bindung
- automatische Erzeugung von Code-Fragmenten
- ad-hoc-Methoden



# SOAP, Client-Seite

- Börsenticker in Perl

```
use SOAP::Lite;
```

```
$price = SOAP::Lite->
```

```
    service('http://www.xmethods.net/sd/StockQuoteService.wsdl')
```

```
    ->getQuote('HRL');
```

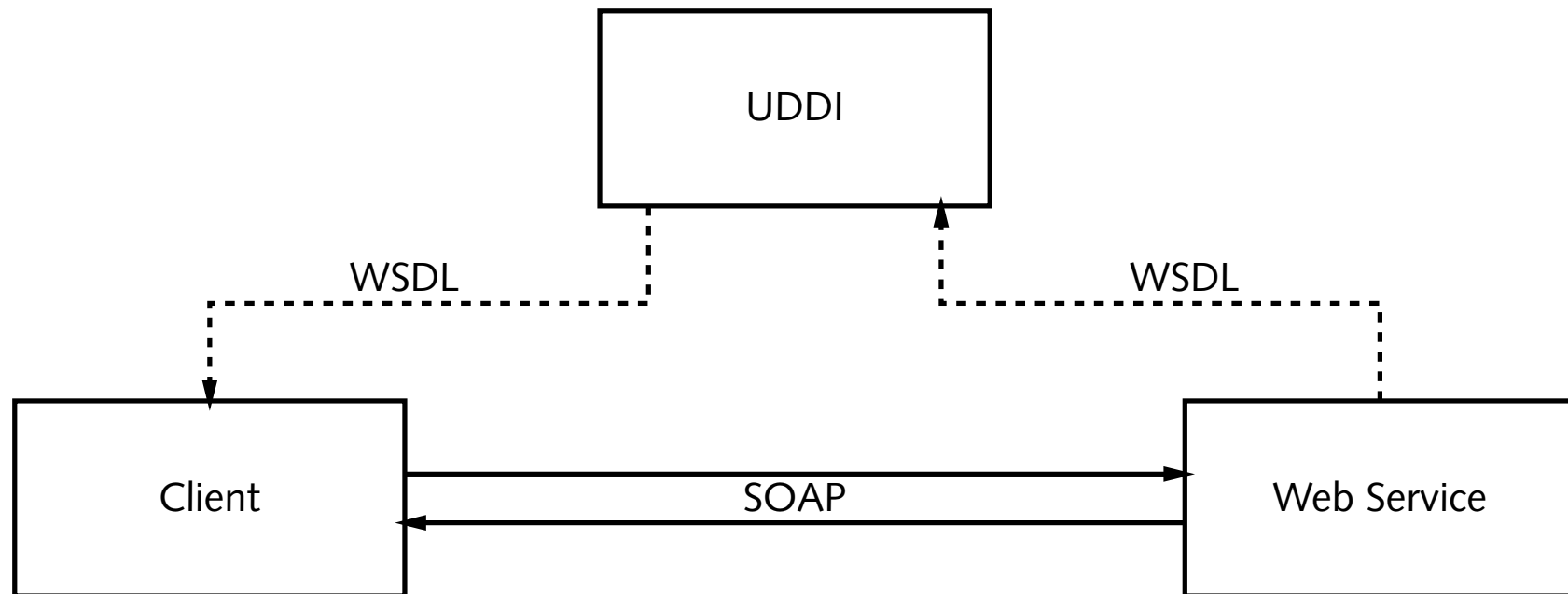
```
print "Hormel: $price\n";
```

- Methode `getQuote` wird on-the-fly definiert

# Services suchen und finden

- UDDI – Universal Description, Discovery and Integration
- Verzeichnisdienst(e) für Web Services:
  - white pages: Adressen, Ansprechpartner
  - yellow pages: Geschäftsbereiche, Dienstleistungen
  - green pages: technische Spezifikationen der Dienste
- Zukunft ungewiss, u.a. wegen Patent-„Problemen“

# Zusammenspiel



# zum Ausprobieren

- Frameworks:
  - ONE (Sun Microsystems)
  - .NET (Microsoft)
  - Mono (Ximian)
- Toolkits:
  - Java Web Services Developer Pack (Sun Microsystems)
  - Axis (Apache)
  - WebSphere SDK for Web Services (IBM)
  - ...

# Web Services in der Bioinformatik

- BioMOBY
  - Organismus-Datenbanken (AceDB, DAS, ...)
  - SOAP-basiert
  - eigenes Verfahren zur Registrierung
  - <http://biomoby.org/>
- XML Central of DDBJ
  - SOAP-Services und WSDL-Beschreibungen
  - Blast, ClustalW, Fasta, SRS, ...
  - <http://xml.nig.ac.jp/>

# Web Services auf BiBiServ

- BiBiServ idealer Kunde für Web Services

```
foreach $seqs_ref (@collection) {  
    $alig = SOAP::Lite->  
        service('http://bibiserv.TechFak.... /bibiserv.wsdl')  
        ->Dialign($seqs_ref, $thresh, $maxregions);  
    push(@results, $alig);  
}
```

- Voraussetzungen:
  - klare Definition der Schnittstellen
  - effektiver Scheduler
  - mehr Power...
- Was ist mit „asynchronen“ Jobs?