

Provide and Consume: PHP Web Services

Christian Wenz <chw@hauser-wenz.de>



ApacheCon
Europe 06

Some Quotes

- „Many Web Services Initiatives Risk Failure”
 - Gartner, July 2003
- „Gartner Says Web Services Ready for Prime Time”
 - eWeek, May 2004
- „Gartner predicts American business is going to squander \$1 billion on misguided Web services projects by 2007 “
 - CIO Magazine, October 2003



ApacheCon
Europe 06

1

And What About PHP?

- Focus on .NET and J2EE
- In comparisons, PHP gets very little attention
- However Googleficht knows:

web services comparison .NET J2EE (92 900 results) versus web services comparison PHP (1 940 000 results)

The winner is: web services comparison PHP



ApacheCon
Europe 06

2

Aim Of This Presentation

- What's this Web Services stuff all about?
- What's about these acronyms?
- How can I consume (use) Web Services with PHP?
- How can I create my own Web Services with PHP?
- PHP 4 vs. PHP 5

ApacheCon
Europe 06

3

Agenda

- A: Web Services - The Idea
- B: Web Services - The Standards
- C: Web Services in PHP
- (D: Wish List)

ApacheCon
Europe 06

4

Agenda

- A: Web Services - The Idea
- B: Web Services - The Standards
- C: Web Services in PHP
- (D: Wish List)

ApacheCon
Europe 06

5

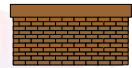
Web Services - The Idea

- Web Services are used for business-to-business communication
- This idea is rather old, however now there are specifications that describe how information is exchanged



Web Services - Data Transmission

- Often, data is transmitted using HTTP
 - Good: Goes through most firewalls
 - Bad: Firewall makers upgrade their systems



Old Idea, New Idea, Both?

- No, Web Services are no new concept
- No, Web Services were not invented (nor patented) by Microsoft
- However: Microsoft worked on many of the standards (!)
- However: Microsoft (and many others) promote Web Services with their huge budgets

Key Players

- Dave Winer - worked on XML-RPC
- Don Box - Microsoft's Web Services guru (but started on the standardization with his own company)
- IBM, HP, ...
- Reference implementations (Amazon, Google, Yahoo!, Dell)
- Standardization committees (W3C, OASIS)



Why Is Everybody Using This?

- Standards enable communication between totally different machines (and developers)
- Use of XML which is sexy, cool (and: human-readable, standardized)
- Machine-to-Machine communication means: No human interaction involved
- Gartner says it's important! ☺

Why Should You Be Using This?

- Currently, rather few visible implementations (Amazon, Google, Yahoo!, eBay et al.)
- However, Web Services offer many possibilities:
 - Communication between servers (or websites) is easy
 - Platform independence
 - New Markets, new business models
 - More and more support in the various languages and technologies
 - Currently, 18 PEAR packages

Why Not?

- It may be a hype
- Remember WAP?
- It is rather useless to use a Web Service on one page and call it from another page on the same web server
- KISS vs. early adopting



ApacheCon
Europe 06

12

Agenda

- A: Web Services - The Idea
- B: Web Services - The Standards
- C: Web Services in PHP
- (D: Wish List)

ApacheCon
Europe 06

13

What About These Acronyms?

- Enough inspiration for a whole round of buzzword bingo
 - XML-RPC
 - SOAP
 - WSDL
 - UDDI
 - DISCO
 - REST
- However, PHP takes care of most of this stuff for you



ApacheCon
Europe 06

14

XML-RPC

- „XML Remote Procedure Call(s)“
- Used for message exchange using HTTP and XML
- Quite easy to use
- Developed by Dave Winer (so we thought)
- However, done in then-secret cooperation with Microsoft

ApacheCon
Europe 06

15

Let's Have A Look

- A request

```
POST /webservice.php HTTP/1.1
Host: www.hauser-wenz.de
Content-Type: text/xml; charset=utf-8
Content-Length: 1234

<?xml version="1.0" encoding="utf-8"?>
<methodCall>
<methodName>hauserwenz.GetStockPrice</methodName>
  <params>
<param><value><string>PHP</string></value></param>
  </params>
</methodCall>
```

ApacheCon
Europe 06

16

Let's Have A Look (2)

- A response

```
HTTP/1.1 200 OK
Content-Type: text/xml; charset="utf-8"
Content-Length: 1234

<?xml version="1.0" encoding="utf-8"?>
<methodResponse>
  <params>
    <param><value><string>$41.00</string></value>
  </param>
  </params>
</methodResponse>
```

ApacheCon
Europe 06

17

So, What Do You Think?

- Good:
 - Easy to use
 - Not too much code overhead
 - Quite in use, e.g. PEAR project
 - Short learning curve (you could do this manually)
- Bad:
 - Not „standardized“ by a committee
 - Quite complex with simple data
 - No user-defined data types

SOAP



- Successor of XML-RPC
- Used to be an acronym for „Simple Object Access Protocol“
- Is, however, neither simple nor relevant for object access
- Therefore, now stands for ... SOAP
- XML-based protocol to exchange messages
- Can be embedded in HTTP (but also in SMTP, ...)

Let's Have A Look

• A request

```
POST /webservice.php HTTP/1.1
Host: www.hauser-wenz.de
Content-Type: text/xml; charset=utf-8
Content-Length: 1234
SOAPAction: "http://hauser-wenz.de/GetStockPrice"

<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <ns1:GetStockPrice xmlns:ns1="http://hauser-wenz.de/"
      soap:encodingStyle="http://schemas.xmlsoap.org/soap/encoding">
      <Stock xsi:type="xsd:string">PHP</Stock>
    </GetStockPrice>
  </soap:Body>
</soap:Envelope>
```

Let's Have A Look (2)

• A response

```
HTTP/1.1 200 OK
Content-Type: text/xml; charset=utf-8
Content-Length: 1234

<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <ns1:GetStockPriceResponse xmlns:ns1="http://hauser-wenz.de/"
      soap:encodingStyle="http://schemas.xmlsoap.org/soap/encoding">
      <Price xsi:type="xsd:string">$42.00</Price>
    </GetStockPriceResponse>
  </soap:Body>
</soap:Envelope>
```

So, What Do You Think?

- Good:
 - Very flexible
 - Very mighty (including user-defined data types)
 - Sanctioned by W3C
 - Most software vendors agree on SOAP
- Bad:
 - Not quite lightweight
 - Possibly bad data/information ratio

WSDL

- „Web Services Description Language“
- Describes the service
 - Methods
 - Parameters
 - Return values
 - Data types
 - Location of service

Let's Have A Look

- A WSDL description

```
<?xml version="1.0" encoding="utf-8"?>
<definitions name="WebService"
  xmlns="http://schemas.xmlsoap.org/wsdl/"
  xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap"
  xmlns:tns="http://hauser-wenz.de/WebService.wsdl"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <message name="GetStockPrice">
    <part name="Stock" type="xsd:string"/>
  </message>
  <message name="GetStockPriceResponse">
    <part name="Price" type="xsd:string"/>
  </message>
  <portType name="QuotePortType">
    <operation name="GetStockQuote">
      <documentation>Retrieve current stock quote of a
        company submitted as parameter</documentation>
      <input message="tns:GetStockPrice" />
      <output message="tns:GetStockPriceResponse" />
    </operation>
  </portType>
```

Lots of namespaces ...

Messages (input/output)

Port (with operations)

ApacheCon Europe 06

24

Let's Have A Look (2)

- A WSDL description (cont'd)

```
<binding name="WebServiceSoap"
  type="tns:QuotePortType">
  <soap:binding
    transport="http://schemas.xmlsoap.org/soap/http">
  </soap:binding>
  <operation name="GetStockPrice">
    <soap:operation soapAction="http://hauser-
      wenz.de/webservice.php"/>
    <input>
      <soap:body use="literal" />
    </input>
    <output>
      <soap:body use="literal" />
    </output>
  </operation>
</binding>
```

Bindings (rules for data transfer)

ApacheCon Europe 06

25

Let's Have A Look (3)

- A WSDL description (cont'd)

```
<service name="WebService">
  <port name="WebServiceSoap"
    binding="wsdl:WebServiceBinding">
    <soap:address
      location="http://www.hauser-
        wenz.de/webservice.php"/>
    </port>
  </service>
</definitions>
```

Which services, where?

- This looks rather messy!
 - Automatism do prove really helpful here

ApacheCon Europe 06

26

UDDI

- „Universal Description, Discovery, and Integration“
- Provide „Yellow Pages“ for web services (and more)
- Web services information is stored in UBRs - Universal Business Registries - that may be queried using a standardized API
- Various UBRs synchronize
 - Microsoft, IBM, SAP, HP, NTT, ...
- Unfortunately, most test registries have been recently shut down

ApacheCon Europe 06

27

DISCO

- Nope, no dancing, but short for „Discovery“
- A Microsoft invention
 - that failed to become successful
- Combines WSDL description and textual information about a web service

ApacheCon Europe 06

28

Let's Have A Look

- A DISCO description

```
<?xml version="1.0" encoding="utf-8" ?>
<discovery xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns="http://schemas.xmlsoap.org/disco/">
  <contractRef ref="http://hauser-
    wenz.de/webservice.php?wsdl">
    docRef="http://hauser-wenz.de/serviceinfo.php"
  </contractRef>
  <soap address="http://hauser-wenz.de/webservice.php"
    xmlns:q1="http://hauser-wenz.de/"
    binding="q1:GetStockPrice"
    xmlns="http://schemas.xmlsoap.org/disco/soap/" />
</discovery>
```

ApacheCon Europe 06

29

Agenda

- A: Web Services - The Idea
- B: Web Services - The Standards
- **C: Web Services in PHP**
- (D: Wish List)

Web Services With PHP

- Well, you can do it manually!
 - It's just XML and all standardized ☺
- However, there are modules that come in really handy!
 - On some web pages
 - In PEAR
 - In PHP itself

XML-RPC

- XML-RPC For PHP
 - Homepage: <http://phpxmlrpc.sourceforge.net/>
 - Widely used, after some stalling, development is up to speed again
 - You might be luckier in CVS
- PEAR::XML_RPC
 - Homepage: http://pear.php.net/package/XML_RPC
 - Stig Bakken's port of XML-RPC for PHP
 - Without the bugs I filed for XML-RPC ;-)
- XML-RPC Client/Server
 - Homepage: <http://keithdevens.com/software/xmlrpc>
 - Still in active development
 - Version 3.0 exclusively for PHP5

SOAP

- NuSOAP
 - Homepage: <http://dietrich.ganx4.com/nusoap/>
 - LGPL-rewrite of SOAPx4
 - Currently maintained by Scott Nichol
 - Quite easy to use (if you use PHP4)
 - Can create WSDL automatically
- PEAR::SOAP
 - Homepage: <http://pear.php.net/package/SOAP>
 - Also based on SOAPx4
 - Maintained by Shane Caraveo and Arnaud Limbourg
 - Under PHP4, the de-facto reference SOAP implementation

SOAP (2)

- ext/soap a.k.a. PHP5-SOAP
 - Homepage: <http://php.net/manual/en/ref.soap.php>
 - New SOAP extension for PHP5
 - Written in C, so much faster than PEAR::SOAP
 - However, still lacks a bit the stability of PEAR::SOAP
 - Still some problems
 - No built-in WSDL generation
 - But in the future the most promising extension

UDDI

- PEAR::UDDI
 - Homepage: <http://pear.php.net/package/UDDI>
 - Supports the inquiry API of UDDI 2.0
 - Publish API is in the works
 - LGPL-port of PHPUDDI to PEAR

Demo Time!

- Using XML-RPC
- Using NuSOAP
- Using PEAR::SOAP
- Using PHP5-SOAP
- Using PEAR::UDDI

Why So Complicated?



- Often, REST is "good enough"
- REpresentational State Transfer
 - PhD thesis of Roy Fielding, co-author of the HTTP specification
- Use HTTP for the request (parameters via GET)
- Retrieve XML via HTTP as the response
- Yep, that's it.

Why REST?

- Very simple
- PHP (5) offers great XML support
 - Especially SimpleXML comes in handy
- Less error-prone than anything else
- Good support from popular web services
 - Amazon, Yahoo!, Google, ...
- Every search engine uses REST, more or less
 - <http://www.google.com/search?q=REST>
 - <http://a9.com/REST>

Why not REST?

- You have to take care of object serialization
- Hard(er) to implement security
- There is no such thing as REST-WSDL
- More request/response approach than demanded by SOA activists

Yahoo! Web Services

- Demo time!
 - Calling Yahoo! Web Service using REST & SimpleXML
 - Add some AJAX to the mix ...

Agenda

- A: Web Services - The Idea
- B: Web Services - The Standards
- C: Web Services in PHP
- (D: Wish List)

Wish List

- More documentation ☺ (as always)
- stay tuned --> • WSDL generation for PHP5-SOAP
- Integration of web services functionality in relevant PHP IDEs
- Support of security, transaction and process standards in PHP
- More interesting, useful Web Services

ApacheCon
Europe 06

42

WSDL Generation With PHP5-SOAP

- There are options!
 - Services_Webservice (now in PEAR)
 - <http://thinkforge.org/projects/wsdngen/>
 - Ctrix_SOAP_AutoDiscover
 - <http://ctrx.org/index.php?page=CtrxSoapAutoDiscover>
 - Webservice Helper tool
 - http://jool.nl/new/index.php?file_id=1
 - WSDL_Gen
 - http://www.schlossnagle.org/~george/php/WSDL_Gen.tgz

ApacheCon
Europe 06

43

Thank You!

HAUSER & WENZ
PUBLICATION TRAINING CONSULTING

Questions?!

<http://www.hauser-wenz.de/blog/>

ApacheCon
Europe 06

44