ApacheCon

# PHP Web Services 2007

Christian Wenz
chw@hauser-wenz.de
http://www.hauser-wenz.de/blog/

Leading the Wave
of Open Source

ApacheCon

## 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?
- What's up with Open ID, WS-*, ...?

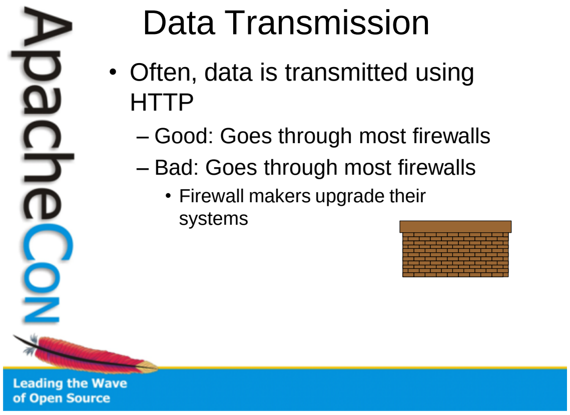
Leading the Wave
of Open Source

ApacheCon

## 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

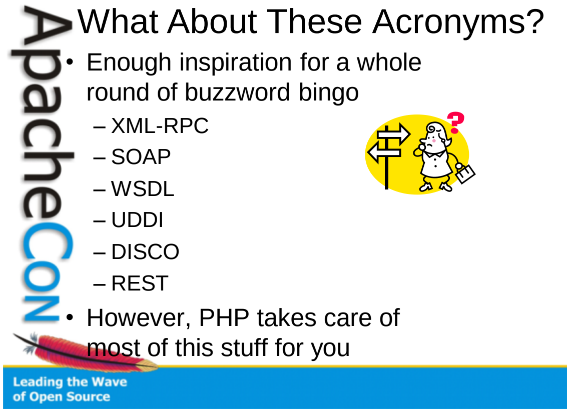
Leading the Wave
of Open Source



## Data Transmission

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

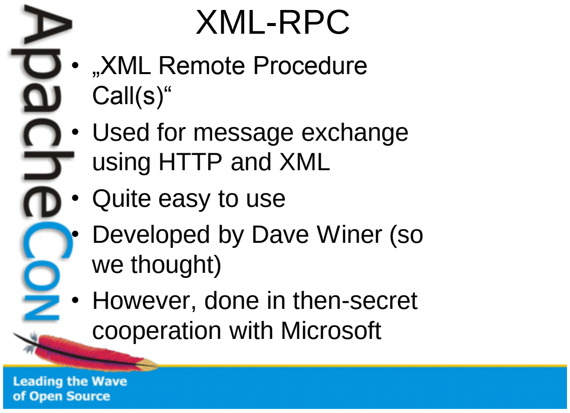
ApacheCon
Leading the Wave of Open Source



## 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

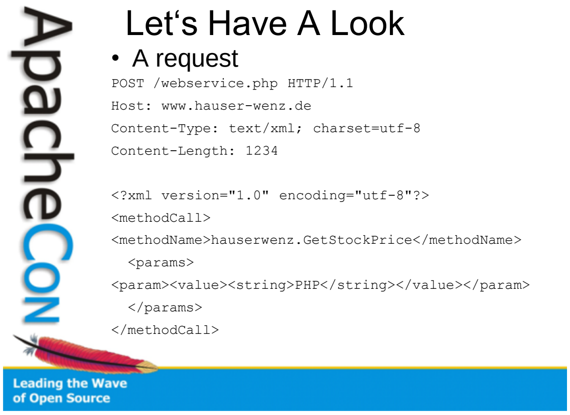
ApacheCon
Leading the Wave of Open Source



## 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
Leading the Wave of Open Source



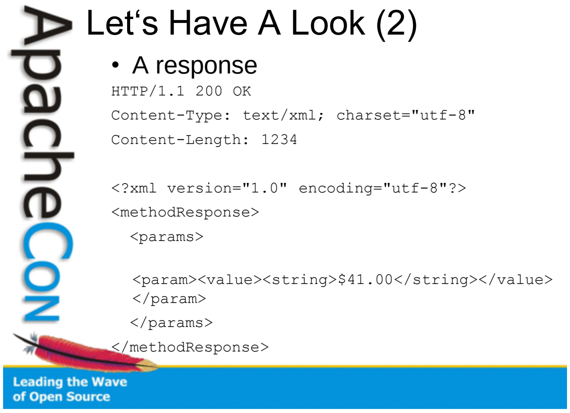
## Let‘s Have A Look

- 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
Leading the Wave of Open Source



## Let‘s Have A Look (2)

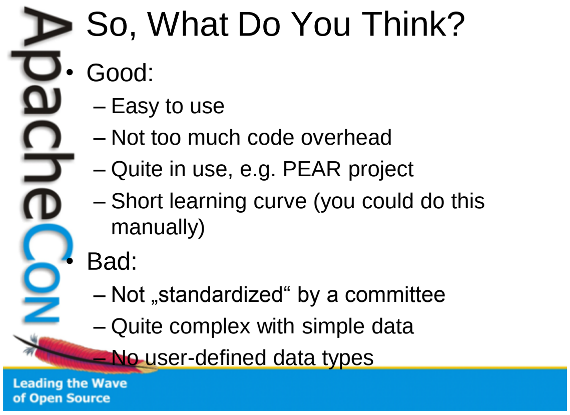
- A response

```
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>
```

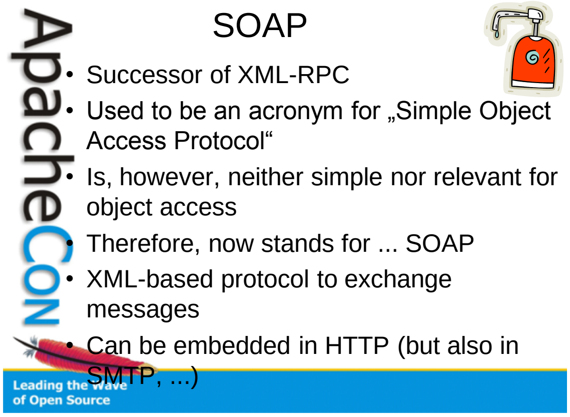
ApacheCon
Leading the Wave of Open Source



## So, What Do You Think?

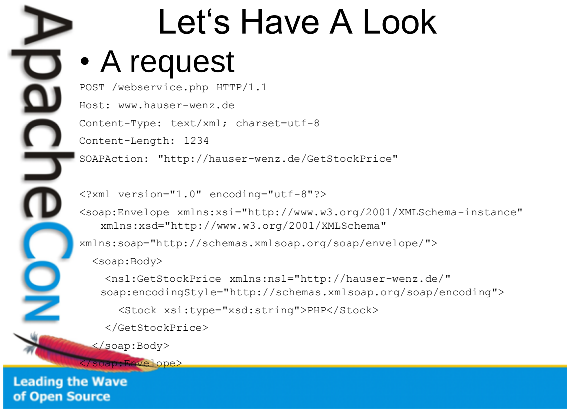
- Good:
  - Easy to use
  - Not too much code overhead
  - Quite in use, e.g. PEAR project
  - Short learning curve (you could do this manually)
- Bad:
  - Not „standardized“ by a committee
  - Quite complex with simple data
  - No user-defined data types

ApacheCon
Leading the Wave of Open Source



## 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, ...)

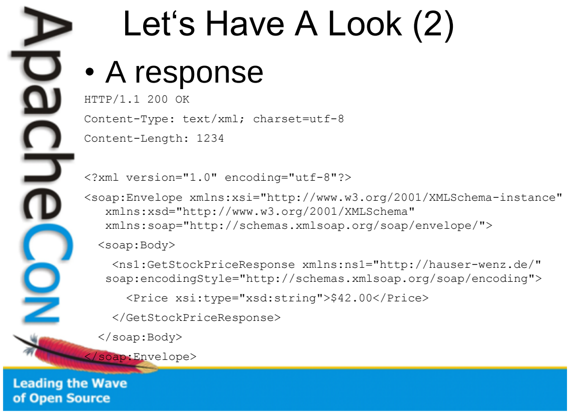


## Let‘s Have A Look

- 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>
```

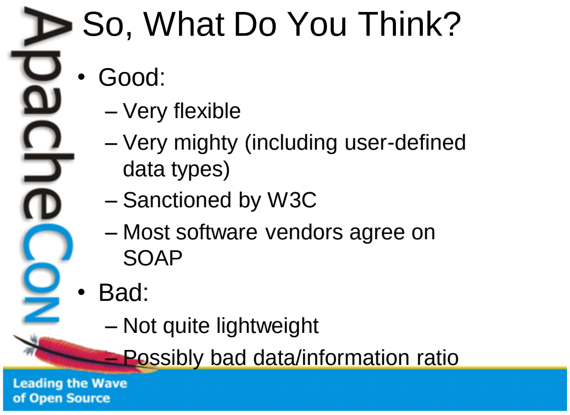


## Let‘s Have A Look (2)

- A response

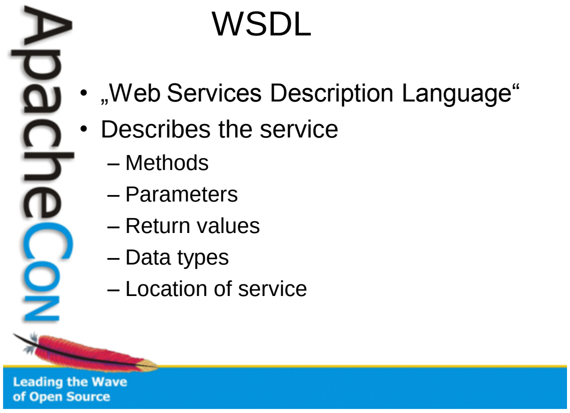
```
HTTP/1.1 200 OK
Content-Type: text/xml; charset=utf-8
Content-Length: 1234

<?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>
```



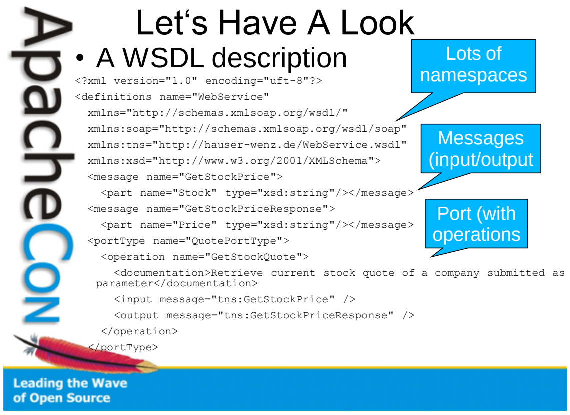
## So, What Do You Think?

- 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="uft-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

## 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"/>
  <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)

## Let‘s Have A Look (3)

- A WSDL description (cont‘d)

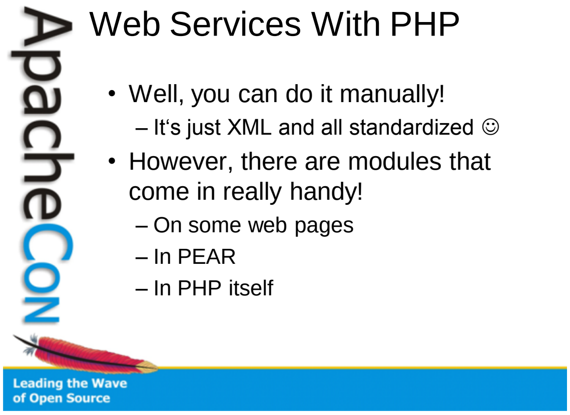
```
 <service name="WebService">
    <port name="WebServiceSoap"
  binding="wsdlns:WebServiceBinding">
      <soap:address
        location="http://www.hauser-
  wenz.de/webservice.php"/>
    </port>
  </service>
</definitions>
```

Which services,

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

## 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 URBs synchronize
  - Microsoft, IBM, SAP, HP, NTT, ...
- Unfortunately, most test registries have been recently shut down

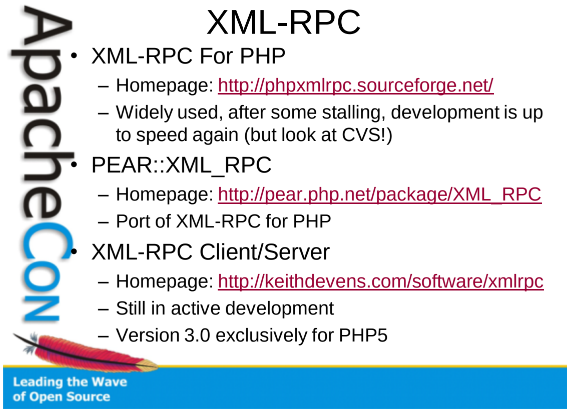


## Web Services With PHP

- 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

ApacheCon

Leading the Wave
of Open Source

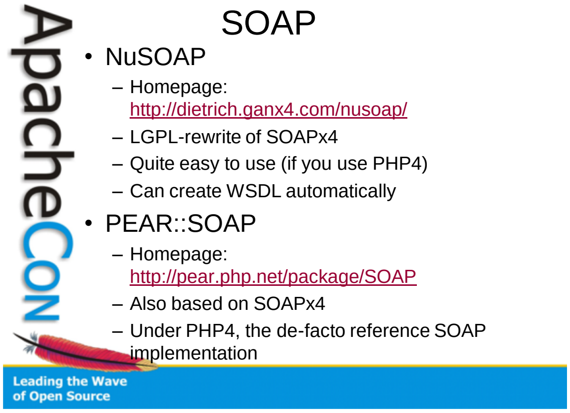


## XML-RPC

- XML-RPC For PHP
  - Homepage: http://phpxmlrpc.sourceforge.net/
  - Widely used, after some stalling, development is up to speed again (but look at CVS!)
- PEAR::XML_RPC
  - Homepage: http://pear.php.net/package/XML_RPC
  - Port of XML-RPC for PHP
- XML-RPC Client/Server
  - Homepage: http://keithdevens.com/software/xmlrpc
  - Still in active development
  - Version 3.0 exclusively for PHP5

ApacheCon

Leading the Wave
of Open Source

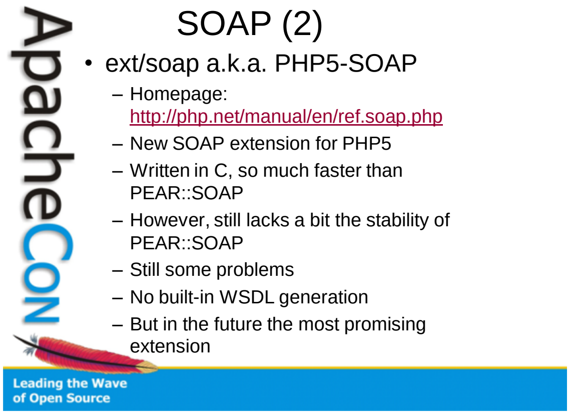


## SOAP

- NuSOAP
  - Homepage: http://dietrich.ganx4.com/nusoap/
  - LGPL-rewrite of SOAPx4
  - 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
  - Under PHP4, the de-facto reference SOAP implementation

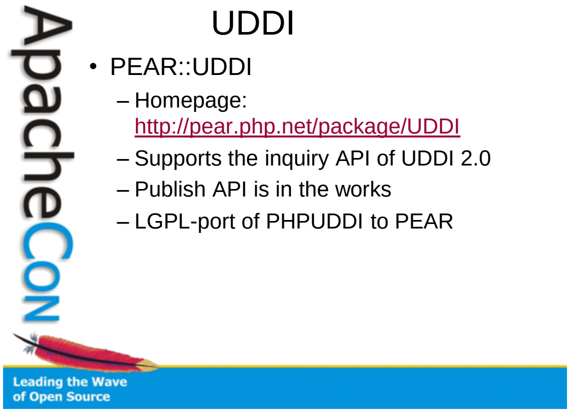
ApacheCon

Leading the Wave
of Open Source



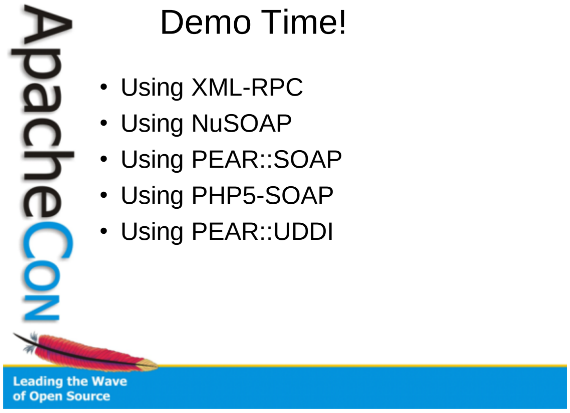
## SOAP (2)

- 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



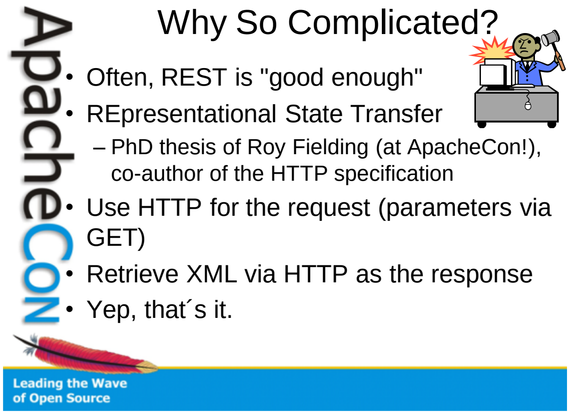
## UDDI

- 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?

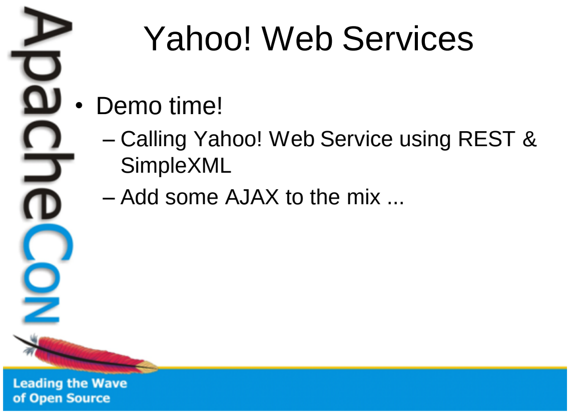
- Often, REST is "good enough"
- REpresentational State Transfer
  - PhD thesis of Roy Fielding (at ApacheCon!), co-author of the HTTP specification
- Use HTTP for the request (parameters via GET)
- Retrieve XML via HTTP as the response
- Yep, that´s it.

## 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?

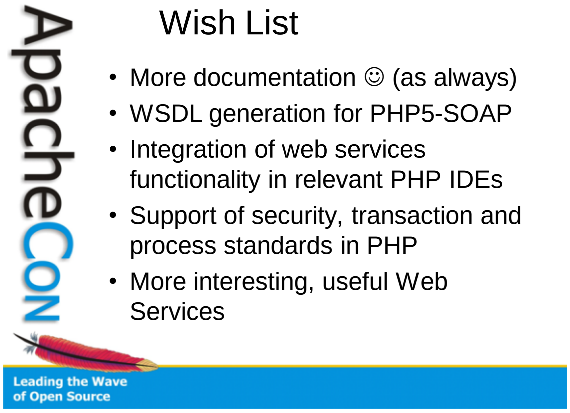
- You have to take care of object serialization
- Hard(er) to implement security
- There is no such thing as REST-WSDL
- More request/response approach than demanded by SOA activists



## Yahoo! Web Services

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

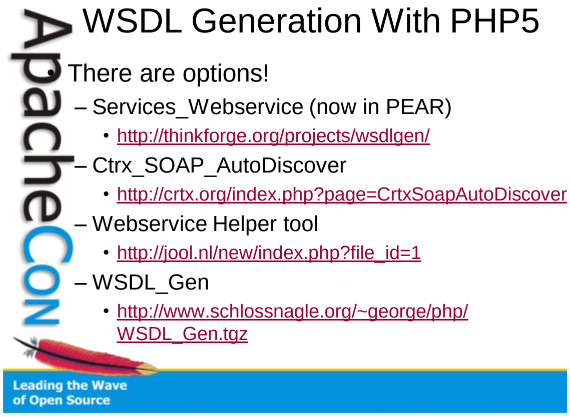
Leading the Wave
of Open Source



## Wish List

- More documentation ☺ (as always)
- 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

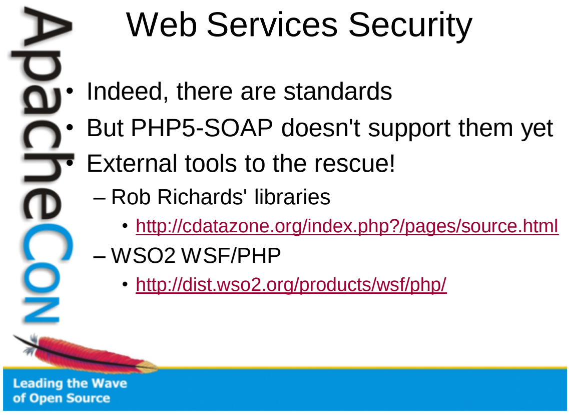
Leading the Wave
of Open Source



## WSDL Generation With PHP5

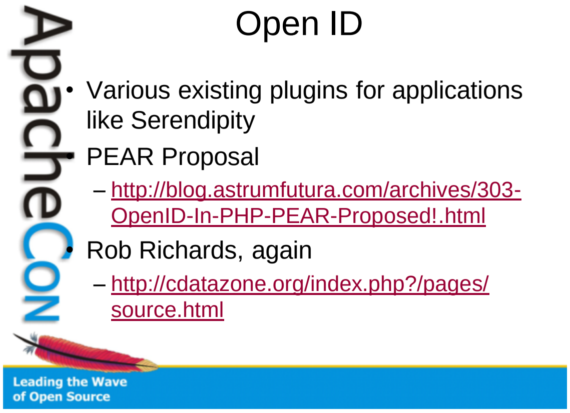
- There are options!
  - Services_Webservice (now in PEAR)
    - http://thinkforge.org/projects/wsdlgen/
  - Ctrx_SOAP_AutoDiscover
    - http://crtx.org/index.php?page=CrtxSoapAutoDiscover
  - Webservice Helper tool
    - http://jool.nl/new/index.php?file_id=1
  - WSDL_Gen
    - http://www.schlossnagle.org/~george/php/WSDL_Gen.tgz

Leading the Wave
of Open Source


ApacheCon
Leading the Wave
of Open Source

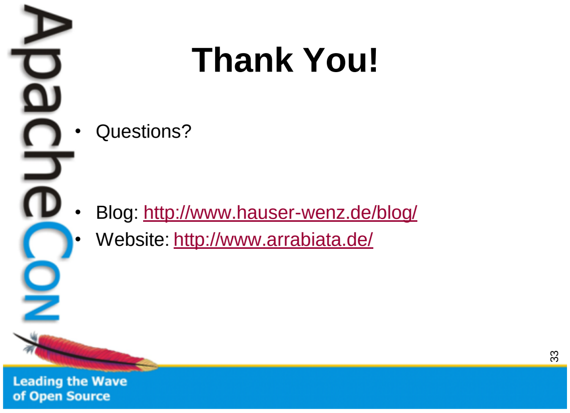
## Web Services Security

- Indeed, there are standards
- But PHP5-SOAP doesn't support them yet
- External tools to the rescue!
  - Rob Richards' libraries
    - http://cdatazone.org/index.php?/pages/source.html
  - WSO2 WSF/PHP
    - http://dist.wso2.org/products/wsf/php/


ApacheCon
Leading the Wave
of Open Source

## Open ID

- Various existing plugins for applications like Serendipity
- PEAR Proposal
  - http://blog.astrumfutura.com/archives/303-OpenID-In-PHP-PEAR-Proposed!.html
- Rob Richards, again
  - http://cdatazone.org/index.php?/pages/source.html


ApacheCon
Leading the Wave
of Open Source

## **Thank You!**

- Questions?

- Blog: http://www.hauser-wenz.de/blog/
- Website: http://www.arrabiata.de/

33