

# Web Services

Thierry Sans

How would you interface your web app with ...?

twitter



YAHOO! FINANCE  
YAHOO! FINANCE

Google™

amazon.com®

The  
Weather  
Channel



# How would you make your data available to others?

Other client platforms that could be

- a web app
- a phone app
- a desktop app

# Different solutions

<b>The “hacking” solution</b>	Data scraping (get the page and parse it)
<b>The “json” solution</b>	JSON data structure
<b>The “dedicated” solution</b>	Programming APIs (based on JSON under the hood)
<b>The “formal” solution</b>	Web Services (SOAP messages)



# Web Services

Implementation of Remote Procedure Calls (RPC) over HTTP (other protocols also supported such as SMTP)

- The remote procedure is called **a web service**
- Request/Response encoded in a SOAP envelope
- Data type and representation defined as a XML schema

➡ Mostly used between web servers (B2B)

✓ Service Oriented Architecture (SOA)

# Web Services Standards

Many W3C standards (source *xhtml.com*)



# The most important ones

**SOAP** (Simple Object Protocol)

- Provide a way to exchange message

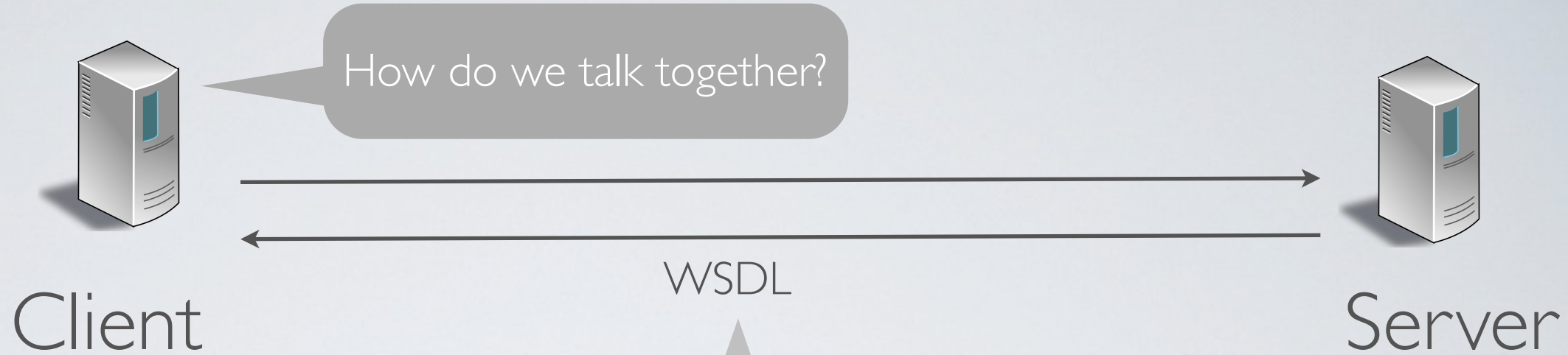
**WSDL** (Web Service Definition Language)

- Provide a way to describe your web service

**UDDI** (Universal Definition Language)

- Provide a way to advertise your web service

# WSDL



```
<message name="getPriceRequest">
  <part name="Item" type="xs:string"/>
</message>

<message name="getPriceResponse">
  <part name="Price" type="xs:double"/>
</message>

<portType name="glossaryPrice">
  <operation name="getPrice">
    <input message="getPriceRequest"/>
    <output message="getPriceResponse"/>
  </operation>
</portType>

<binding type="glossaryTerms" name="b1">
  <soap:binding style="document" transport="http://schemas.xmlsoap.org/soap/http" />
  <operation>
    <soap:operation soapAction="http://example.com/getPrice"/>
    <input><soap:body use="literal"/></input>
    <output><soap:body use="literal"/></output>
  </operation>
</binding>
```



# SOAP Request

```
<?xml version="1.0"?>
<soap:Envelope xmlns:soap="http://www.w3.org/2001/12/soap-
envelope"
soap:encodingStyle="http://www.w3.org/2001/12/soap-
encoding">

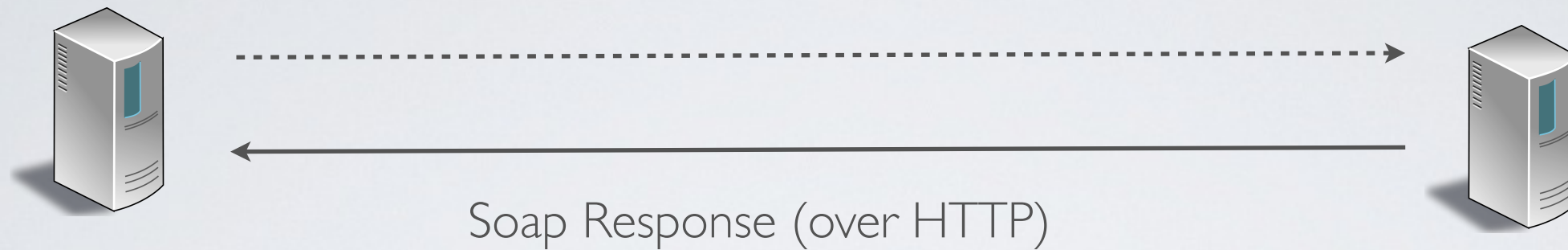
  <soap:Body>
    <m:GetPrice xmlns:m="http://www.w3schools.com/prices">
      <m:Item>Apples</m:Item>
    </m:GetPrice>
  </soap:Body>

</soap:Envelope>
```

example from [www.w3school.com](http://www.w3school.com)



# SOAP Response



Client

```
<?xml version="1.0"?>
<soap:Envelope xmlns:soap="http://www.w3.org/2001/12/soap-
envelope"
soap:encodingStyle="http://www.w3.org/2001/12/soap-
encoding">

  <soap:Body>
    <m:GetPriceResponse xmlns:m="http://www.w3schools.com/
prices">
      <m:Price>1.90</m:Price>
    </m:GetPriceResponse>
  </soap:Body>

</soap:Envelope>
```

example from [www.w3school.com](http://www.w3school.com)

# Conclusion

A good idea but have **not been widely adopted**

- Very flexible but very complex architecture
- Standards evolve faster than development frameworks
- Ad-hoc solutions adopted by the main actors of the web