Introduction to Web Services
Agenda

• Motivation
• History
• Web service model
• Web service components
• A walkthrough examples
Motivation

• The ability to program the Web.
• Example: Consider an Excel spreadsheet that summarizes your whole financial picture: stocks, bank accounts, loans, etc. If some of this information is available through XML Web services, Excel can update it and present the update information to the user.
Web Service definition

A simple definition:

“a Web Service is an application component accessible over open protocols”.
History

• Web services evolved from previous technologies that served the same purpose such as RPC, ORPC (DCOM, CORBA and JAVA RMI).

• Web Services were intended to solve three main problems:
  1. Interoperability
  2. Firewall traversal
  3. Complexity
Interoperability

- Earlier distributed systems suffered from interoperability issues because each vendor implemented its own on-wire format for distributed object messaging.
- Development of DCOM apps strictly bound to Windows Operating system.
- Development of RMI bound to Java programming language.
Firewall traversal

• Collaboration across corporations was an issue because distributed systems such as CORBA and DCOM used non-standard ports.

• Web Services use HTTP as a transport protocol and most of the firewalls allow access though port 80) HTTP, leading to easier and dynamic collaboration.
Complexity

- Web Services is a developer-friendly service system.
- Most of the above-mentioned technologies such as RMI, COM, and CORBA involve a whole learning curve.
- New technologies and languages have to be learnt to implement these services.
Web Service definition revisited

- A more precise definition:
  - an application component that:
    - Communicates via open protocols (HTTP, SMTP, etc.)
    - Processes XML messages framed using SOAP
    - Describes its messages using XML Schema
    - Provides an endpoint description using WSDL
    - Can be discovered using UDDI
Web Services Components

- **XML** — eXtensible Markup Language — A uniform data representation and exchange mechanism.
- **SOAP** — Simple Object Access Protocol — A standard way for communication.
- **UDDI** — Universal Description, Discovery and Integration specification — A mechanism to register and locate WS based application.
- **WSDL** — Web Services Description Language — A standard meta language to described the services offered.
Example – A simple Web Service

- A buyer (which might be a simple client) is ordering goods from a seller service.
- The buyer finds the seller service by searching the UDDI directory.
- The seller service is a Web Service whose interface is defined using Web Services Description Language (WSDL).
- The buyer is invoking the order method on the seller service using Simple Object Access Protocol (SOAP) and the WSDL definition for the seller service.
- The buyer knows what to expect in the SOAP reply message because this is defined in the WSDL definition for the seller service.
The Web Service Model

• The Web Services architecture is based upon the interactions between three roles:
  – Service provider
  – Service registry
  – Service requestor

• The interactions involve the:
  – Publish operations
  – Find operation
  – Bind operations.
The Web Service Model (cont)

The Web Services model follows the *publish*, *find*, and *bind* paradigm.

1. **publish**
2. **find**
3. **bind/invoke**
XML

• XML stands for EXtensible Markup Language.
• XML is a markup language much like HTML.
• XML was designed to describe data.
• XML tags are not predefined. You must define your own tags.
• The perfect choice for enabling cross-platform data communication in Web Services.
XML vs HTML

An HTML example:

```html
<html>
  <body>
    <h2>John Doe</h2>
    <p>2 Backroads Lane<br>
       New York<br>
       045935435<br>
       john.doe@gmail.com<br>
    </p>
  </body>
</html>
```
XML vs HTML

- This will be displayed as:

<table>
<thead>
<tr>
<th>John Doe</th>
</tr>
</thead>
</table>
| 2 Backroads Lane  
New York  
045935435  
John.doe@gmail.com |

- HTML specifies how the document is to be displayed, and not what information is contained in the document.
- Hard for machine to extract the embedded information. Relatively easy for human.
XML vs HTML

• Now look at the following:

```xml
<?xml version=1.0?>
<contact>
    <name>John Doe</name>
    <address>2 Backroads Lane</address>
    <country>New York</country>
    <phone>045935435</phone>
    <email>john.doe@gmail.com</email>
</contact>
```

• In this case:
  – The information contained is being marked, but not for displaying.
  – Readable by both human and machines.
SOAP

- SOAP originally stood for "Simple Object Access Protocol".
- Web Services expose useful functionality to Web users through a standard Web protocol called SOAP.
- Soap is an XML vocabulary standard to enable programs on separate computers to interact across any network. SOAP is a simple markup language for describing messages between applications.
- Soap uses mainly HTTP as a transport protocol. That is, HTTP message contains a SOAP message as its payload section.
SOAP Characteristics

• SOAP has three major characteristics:
  – Extensibility — security and WS-routing are among the extensions under development.
  – Neutrality - SOAP can be used over any transport protocol such as HTTP, SMTP or even TCP.
  – Independent - SOAP allows for any programming model.
SOAP Building Blocks

A SOAP message is an ordinary XML document containing the following elements:

– A required Envelope element that identifies the XML document as a SOAP message.
– An optional Header element that contains header information.
– A required Body element that contains call and response information.
– An optional Fault element that provides information about errors that occurred while processing the message.
SOAP Request

POST /InStock HTTP/1.1
Host :www.stock.org
Content-Type :application/soap+xml; charset=utf-8 Content-Length150 :

<?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 xmlns:m"=http://www.stock.org/stock">
    <m:GetStockPrice>
      <m:StockName>IBM</m:StockName>
    </m:GetStockPrice>
  </soap:Body>
</soap:Envelope>
SOAP Response

HTTP/1.1 200 OK
Content-Type :application/soap; charset=utf-8
Content-Length126 :

<?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 xmlns:m="http://www.stock.org/stock">
  <m:GetStockPriceResponse>
    <m:Price>34.5</m:Price>
  </m:GetStockPriceResponse>
</soap:Body>
</soap:Envelope>
SOAP Security

• SOAP uses HTTP as a transport protocol and hence can use HTTP security mainly HTTP over SSL.

• But, since SOAP can run over a number of application protocols (such as SMTP) security had to be considered.

• The **WS-Security specification** defines a complete encryption system.
WSDL

• WSDL stands for Web Services Description Language.
• WSDL is an XML vocabulary for describing Web services. It allows developers to describe Web Services and their capabilities, in a standard manner.
• WSDL specifies what a request message must contain and what the response message will look like in unambiguous notation. In other words, it is a contract between the XML Web service and the client who wishes to use this service.
• In addition to describing message contents, WSDL defines where the service is available and what communications protocol is used to talk to the service.
The WSDL Document Structure

• A WSDL document is just a simple XML document.
• It defines a web service using these major elements:
  – **port type** - The operations performed by the web service.
  – **message** - The messages used by the web service.
  – **types** - The data types used by the web service.
  – **binding** - The communication protocols used by the web service.
WSDL Document

<message name="GetStockPriceRequest">
  <part name="stock" type="xs:string"/>
</message>

<message name="GetStockPriceResponse">
  <part name="value" type="xs:string"/>
</message>

<portType name="StocksRates">
  <operation name="GetStockPrice">
    <input message="GetStockPriceRequest"/>
    <output message="GetStockPriceResponse"/>
  </operation>
</portType>
UDDI

- UDDI stands for Universal Description, Discovery and Integration.
- UDDI is a directory for storing information about web services, like yellow pages.
- UDDI is a directory of web service interfaces described by WSDL.
Resources

• http://msdn.microsoft.com/webservices/understanding/webservicebasics/default.aspx
• http://www.w3schools.com/
• http://uddi.microsoft.com/Default.aspx
• http://www.developer.com/services/article.php/2195981
• Many more on the web…