Service Oriented Web Applications
Course Overview

- Gaining knowledge and skills for designing, developing and maintaining service oriented web applications

- Creating
  - web services
  - n-tier web applications

- What’s covered?
  - XML, DTD, Schemas, CSS, XSLT, AJAX, DOM, SAX
  - PHP, C#, Java, JavaScript
  - MySQL, SQL Server
References

- There are plenty of resources on The Web
  - although some of it may be past its use by date
- You have all of my teaching material
- If you really do need...
    - both cover many but not all of the topics on the course
    - both good value classics from O'Reilly
Introduction to Web Services

Definition:

- A Web Service is a standards-based, language-agnostic software entity, that accepts specially formatted requests from other software entities on remote machines via vendor and transport neutral communication protocols, producing application specific responses.
The simplest Web service system has two participants:
A service producer (provider)
A service consumer (requester).
The provider presents the interface and implementation of the service, and the requester uses the Web service.
WWW Architecture

**Client**

http://w3.foo.org/bar/q.php

**Network**

HTTP over TCP/IP

**Server**

Platform: PC, Mac, *nix, Android, etc.
Web Browser: IE, Mozilla, Opera, etc.

Request:

Response:

<html>...

Platform: PC, Mac, *nix, etc.
Web Server: Apache, IIS, Nginx, etc.
Client/Server, Request/Response architecture

- Client requests a resource
  - HTTP request
  - e.g.
    - `http://w3.foo.org/bar/q.php?name=Leon&pie=apple`

- The web server responds with data
  - HTTP response
  - usually in the form of a web page
    - an HTML document
    - could be any file format

- Web pages are identified by a Uniform Resource Locator (URL)
  - protocol: e.g. http
  - web server: e.g. w3.foo.org
    - [machine name].[domain name]
  - resource: e.g. `bar/q.php`
  - can also provide get data: e.g. `?name=Leon&pie=apple`
  - globally unique
protocol

http://www.chicagosymphony.org/civicconcerts/index.htm

Directory

Server name

filename
A more sophisticated system:
A registry, acts as a broker for Web services.
A provider, can publish services to the registry
A consumer, can then discover services in the registry
What is a Web Service?

- A URL that returns XML.
- Web services are open standard (XML, SOAP, HTTP etc.) based Web applications that interact with other web applications for the purpose of exchanging data.
- Web Services can convert your existing applications into Web-applications.
- Web Services combine the best features of distributed computing and eliminates the worst:
  - Provides a mechanism for invoking methods remotely:
    - passing arguments and returning data
  - Uses existing Web standards:
    - i.e. HTTP and XML
    - considered scalable and safe
    - allowed through corporate firewall
To summarize, a complete web service can be described as any service that:

- Is available over the Internet or private (intranet) networks
- Uses a standardized XML messaging system
- Is not tied to any one operating system or programming language
- Is self-describing via a common XML grammar
- Is discoverable via a simple find mechanism
Web Services allow you (the software developer) to develop applications that make use of components anywhere on the Web as long as they are published as Web Services.

Because Web Services are based on open Internet standards (HTTP, XML, etc.) any software that complies to the Web Service protocol should be able to communicate with any other.
A Web Service

client object  ←  LAN or WAN  →  server object

the web (HTTP)  ←  web page  →  the web

plus distributed object systems - DCOM, RMI, CORBA

resulting in XML based web services
Service Oriented Example

- HTML HTTP
- Brokering application
- XML HTTP
- share price service
- XML HTTP
- currency conversion service
- XML HTTP
- authentication service
A Web Service exposes functionality to a consumer
  ◦ over the Internet or intranet
  ◦ a programmable URL
  ◦ functions you can call over the Internet
Based on Web standards
  ◦ HTTP, XML, SOAP, WSDL, UDDI and more
Can be implemented in any language on any platform
Black boxes
  ◦ component-like, reusable
Web Services allow you to interconnect:
  ◦ different companies
  ◦ many /any devices
  ◦ applications
  ◦ different clients
    ◦ not just browsers
Benefits of Web Services

Loosely Coupled
- Each service exists independently of the other services that make up the application. Individual pieces of the application to be modified without impacting unrelated areas.

Ease of Integration
- Data is isolated between applications creating ‘silos’. Web Services act as glue between these and enable easier communications within and across organizations.

Service Reuse
- Takes code reuse a step further. A specific function within the domain is only ever coded once and used over and over again by consuming applications.
Web Services
Evolution of the Web

Generation 1
Static HTML

Generation 2
Web Applications

Generation 3
Web Services
Components of Web Services

- The basic web services platform is XML + HTTP. All the standard web services work using the following components

- SOAP (Simple Object Access Protocol)
- UDDI (Universal Description, Discovery and Integration)
- WSDL (Web Services Description Language)
Web Service Protocol Stack

- **Service discovery**
  - (UDDI)

- **Service description**
  - (WSDL)

- **Messaging layer**
  - (XML, SOAP)

- **Transport layer**
  - (HTTP, SMTP, FTP)

Layers of Web Services Protocol Stack
- Service Transport
  This layer is responsible for transporting messages between applications. Currently, this layer includes Hyper Text Transport Protocol (HTTP), Simple Mail Transfer Protocol (SMTP), File Transfer Protocol (FTP), and newer protocols such as Blocks Extensible Exchange Protocol (BEEP).

- XML Messaging
  This layer is responsible for encoding messages in a common XML format so that messages can be understood at either end. Currently, this layer includes XML-RPC and SOAP.

- Service Description
  This layer is responsible for describing the public interface to a specific web service. Currently, service description is handled via the Web Service Description Language (WSDL).
Service Discovery
This layer is responsible for centralizing services into a common registry and providing easy publish/find functionality. Currently, service discovery is handled via Universal Description, Discovery, and Integration (UDDI).

As web services evolve, additional layers may be added and additional technologies may be added to each layer.
Hypertext Transport Protocol

HTTP

- Currently, HTTP is the most popular option for service transport. HTTP is simple, stable, and widely deployed.

- Most firewalls allow HTTP traffic. This allows XML-RPC or SOAP messages to masquerade as HTTP messages. This is good if you want to integrate remote applications, but it does raise a number of security concerns.
HTTP is the principle top-level TCP/IP protocol used to request and return data over the Web

- data may be a static file on the web server
  - HTML pages, GIFs, JPEGs, Microsoft Word documents, Adobe PDF documents, XML files, etc., etc.
- data may be the result of running some program on the web server
  - ASP, PHP, CGI, JSP, etc., etc.

Request-Response protocol

Methods: GET, POST, ...
HTTP Request

Method: GET
File (Resource): /default.php
HTTP version: HTTP/1.0

Headers:
- Accept: image/gif, image/x-bitmap, image/jpeg, */*
- Accept-Language: en
- User-Agent: Mozilla/1.22 (compatible; MSIE 2.0; Windows 95)
- Connection: Keep-Alive
- If-Modified-Since: Sunday, 17-Apr-96 04:32:58 GMT

POST Data – none for GET
## HTTP Response

<table>
<thead>
<tr>
<th>HTTP version</th>
<th>Status code</th>
<th>Reason phrase</th>
</tr>
</thead>
<tbody>
<tr>
<td>HTTP/1.0 200</td>
<td>OK</td>
<td></td>
</tr>
</tbody>
</table>

**Headers**

```
HTTP/1.0 200 OK
Date: Sun, 21 Apr 1996 02:20:42 GMT
Server: Microsoft-Internet-Information-Server/5.0
Connection: keep-alive
Content-Type: text/html
Last-Modified: Thu, 18 Apr 1996 17:39:05 GMT
Content-Length: 2543

<?xml version="1.0" encoding="UTF-8"?>
...
...```
Extensible Markup Language
XML

Definition:
- The eXtensible Markup Language (XML) is a W3C recommendation for creating special-purpose markup languages that enable the structuring, description and interchange of data.

NOTE: The World Wide Web Consortium (W3C) is the main international standards organization for the World Wide Web (abbreviated WWW or W3).
A simplified subset of SGML capable of describing many different kinds of data for any imaginable application domain.

It facilitates the sharing of structured text and information in databases and across the Internet.

Languages based on XML are themselves described in a formal way, allowing programs to modify and validate documents in these languages without prior knowledge of their form.

Separate syntax from semantics.

Inherently supports internationalization (Unicode) and platform independence.
A meta-language - extensible
  ◦ a language for defining other languages
    e.g. XHTML, RDF, RSS, MathML, SVG, WML, SMIL

Represents hierarchical data
  ◦ tree structure
  ◦ easy for humans and machines to work with

Useful for data exchange and transformation

W3C standard

Separates of content from presentation

Enabling technology for web services and the semantic web

NOTE: The Semantic Web is an extension of the Web through standards by the World Wide Web Consortium.
XML Basics

- XML is designed to represent and transfer structured data
  - In HTML:
    `<p>Jan 15, 2000</p>`
  - In XML:
    `<OrderDate>Jan 15, 2000</OrderDate>`

- XML does not display or transform data
  - XML separates data from formatting and transforming
  - HTML and XML are both derived from SGML
    - in different ways
    - inherited useful legacy

NOTE: SGML (Standard Generalized Markup Language) is a standard for how to specify a document markup language or tag set.
XML Syntax

- XML is composed of tags and attributes
  - tags can be nested
    - representing entities, entity properties, and entity hierarchy

```xml
<root>
<Orders OrderID="10643" CustomerID="ALFKI"
EmployeeID="6" OrderDate="1997-08-25T00:00:00"
RequiredDate="1997-09-22T00:00:00"
ShippedDate="1997-09-02T00:00:00" />
</root>
```
1. Document has a single root element
2. Tags must be properly nested
   - no overlapping tag pairs
3. All tags must have a closing tag
   - or be self closing
4. Tag names are case sensitive
5. Tag attributes are in the opening tag
   - unique attribute name
   - attribute value must be quoted
XML Schemas

- XML resources, documents or languages can be defined using:
  - Document Type Definitions (DTDs)
    - SGML technology
  - XML Schemas
    - XML technology

- Both define:
  - vocabulary
    - the tags and attributes
  - grammar
    - constraints on the tags, attributes and values
    - XML schemas describe the structure of an XML document

- Both enable automated validation of XML
- XML schemas and DTDs are mutually exclusive
An XML Schema enables the following:

- Identification of the elements that can be in a document
- Identification of the order and relation between elements
- Identification of the attributes of every element and whether they’re optional or required or have some other special properties
- Identification of the datatype of attribute content

Think of it as an elaborate UML Class diagram where classes only have field and no methods.
Simple Object Access Protocol (SOAP)

- A mechanism for defining the unit of communication.
- A mechanism for error handling.
- An extensibility mechanism
- Lives above the transport layer of OSI

Simply put its a mechanism that allows the transmission of XML documents, regardless of transport layer protocol.
SOAP is an XML-based protocol for exchanging information between computers.

- It is a communication protocol.
- SOAP is for communication between applications.
- SOAP is a format for sending messages.
- SOAP is designed to communicate via Internet.
- SOAP is platform independent.
- SOAP is language independent.
- SOAP is simple and extensible.
- SOAP allows you to get around firewalls.
- SOAP will be developed as a W3C standard.
Originally developed by Microsoft as a Simple Object Access Protocol circa 1998

- Adopted by W3C who maintain the standards
- A lightweight protocol for exchanging information in a distributed, heterogeneous environment
  - enables cross-platform interoperability
- Interoperable
  - OS, object model & programming language neutral
  - hardware independent
  - protocol independent
- Operates over existing Internet infrastructure
Builds on key Internet standards
  - SOAP ≈ HTTP + XML

The SOAP specification defines:
  - the SOAP message format
  - how to send messages
  - how to receive responses
  - data encoding
SOAP requests are HTTP POST requests

POST /WebCalculator/Calculator.asmx HTTP/1.1
Content-Type: text/xml
SOAPAction: "http://tempuri.org/Add"
Content-Length: 386

<?xml version="1.0"?>
<soap:Envelope ...>
   ...
</soap:Envelope>
The complete SOAP message
Protocol binding headers
<Envelope> encloses payload
<Header> encloses headers
Individual headers
<Body> contains SOAP message name
XML-encoded SOAP message name & data
<?xml version="1.0"?>
<soap:Envelope ...
  <soap:Header ...
    ...
  </soap:Header>
<soap:Body>
  <Add xmlns="http://tempuri.org/"
    <n1>12</n1>
    <n2>10</n2>
  </Add>
</soap:Body>
</soap:Envelope>
HTTP/1.1 200 OK
...
Content-Type: text/xml
Content-Length: 391

<?xml version="1.0"?>
<soap:Envelope ...>
  <soap:Body>
    <AddResult xmlns="http://tempuri.org/">
      <result>28.6</result>
    </AddResult>
  </soap:Body>
</soap:Envelope>
<soap:Envelope ...>
    <soap:Body>
        <GetStockDataResult xmlns="http://tempuri.org/">
            <result>
                <Description>Plastic Novelties Ltd</Description>
                <Price>129</Price>
                <Ticker>PLAS</Ticker>
            </result>
        </GetStockDataResult>
    </soap:Body>
</soap:Envelope>
WSDL

- Web Services Description Language (WSDL) is an XML format for describing all the information needed to invoke and communicate with a Web Service.
- It gives the answers to the questions Who? What? Where? Why? How?
XML schema for describing Web Services

1. Service interface definition
   abstract semantics for the Web Service

2. Service implementation definition
   concrete end points and network addresses where
   the Web Service can be invoked

Clear delineation between abstract and concrete messages
- WSDL was developed jointly by Microsoft and IBM.
- WSDL is an XML based protocol for information exchange in decentralized and distributed environments.
- WSDL is the standard format for describing a web service.
- WSDL definition describes how to access a web service and what operations it will perform.
- WSDL is a language for describing how to interface with XML-based services.
- WSDL is an integral part of UDDI, an XML-based worldwide business registry.
- WSDL is the language that UDDI uses.
UDDI

- Universal Description, Discovery, and Integration
- Open industry initiative to address discovery
  - a registration database for Web Services
- An OASIS standard
  - Organization for the Advancement of Structured Information Standards
  - schema for service providers and descriptions
  - API for publishing and searching
  - developed on industry standards (XML, HTTP, TCP/IP, SOAP)
  - applies to both XML and non-XML services
- UDDI is an XML-based standard for describing, publishing, and finding web services.
- UDDI is a specification for a distributed registry of web services.
- UDDI is platform independent, open framework.
- UDDI can communicate via SOAP, CORBA, and Java RMI Protocol.
- UDDI uses WSDL to describe interfaces to web services.
- UDDI is seen with SOAP and WSDL as one of the three foundation standards of web services.
- UDDI is an open industry initiative enabling businesses to discover each other and define how they interact over the Internet.
REST
- Representational State Transfer
- Originally referred to a collection of architectural principles:
  - A stateless client/server protocol
    - HTTP
  - A set of well-defined operations
    - GET, POST, DELETE
  - A universal syntax for resource-identification
    - URL
  - The use of hypermedia
    - HTML, XML
- Simply a URL that returns XML
- Plain Old XML (POX)