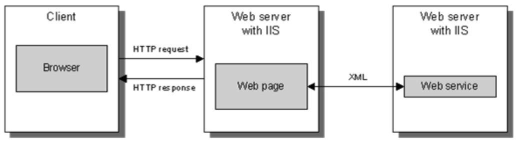
CSIS 4135

Web Services

What is it?

- A *web service* is program code that's accessible over the network for use by other programs
- Exposes part of an application (methods) to other applications
- Independent of language and platform:
  - Could be Java program running on a Linux server calling a service written in VB on a Windows 2008 server

A distributed ASP.NET web application

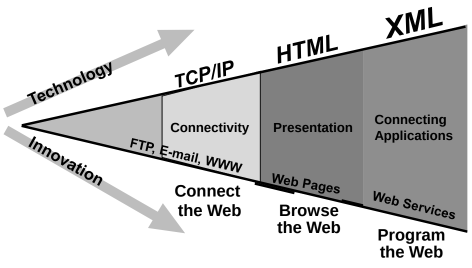


Potential Applications

- Credit purchase approval: Pass credit card number, expiration date, and purchase amount to an authorization service
- Purchase shipping service: Pass origin and destination to service that returns a list of shipping options and prices
- Bookstore pricing service: Give title or ISBN to service that returns prices at online bookstores
  - Aggregation of content from multiple web sites

Underlying Technologies

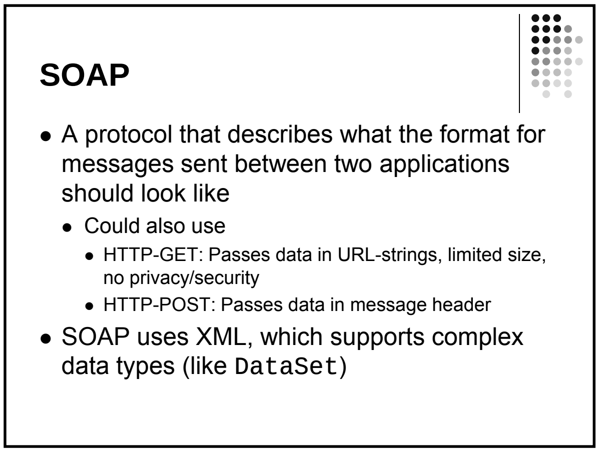
XML Is the Glue



Speaking a Common Language

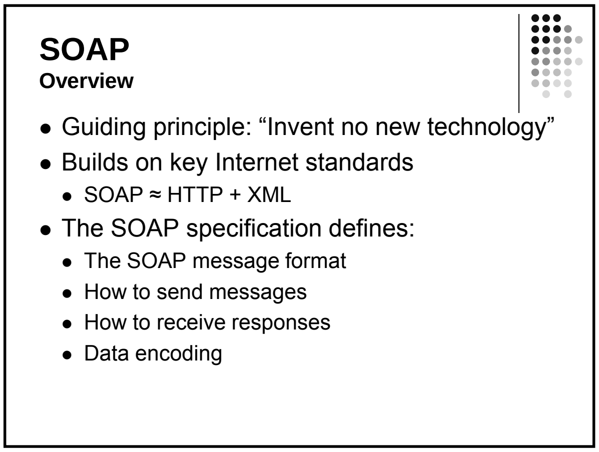
Everything is based on XML

- SOAP – Simple Object Access Protocol
- WSDL – Web Service Description Language
- UDDI – Universal Description, Discovery and Integration (going away?)



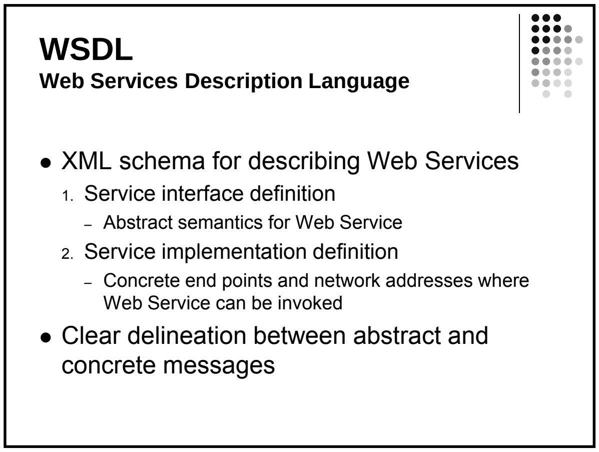
SOAP

- A protocol that describes what the format for messages sent between two applications should look like
  - Could also use
    - HTTP-GET: Passes data in URL-strings, limited size, no privacy/security
    - HTTP-POST: Passes data in message header
- SOAP uses XML, which supports complex data types (like `DataSet`)



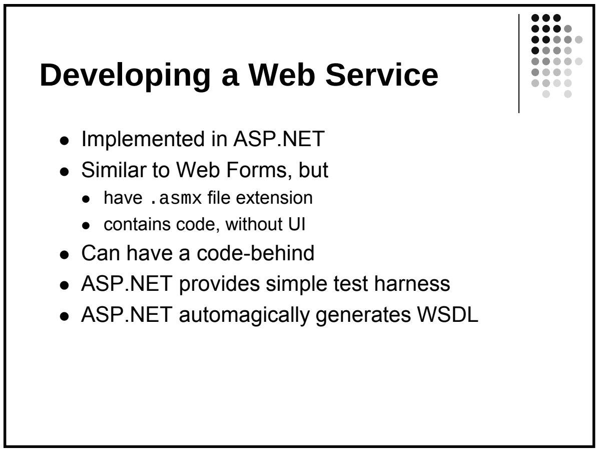
SOAP
Overview

- Guiding principle: “Invent no new technology”
- 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



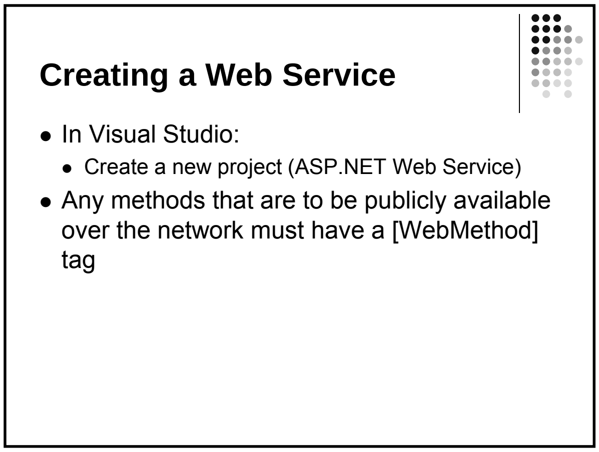
WSDL
Web Services Description Language

- XML schema for describing Web Services
  1. Service interface definition
     - Abstract semantics for Web Service
  2. Service implementation definition
     - Concrete end points and network addresses where Web Service can be invoked
- Clear delineation between abstract and concrete messages



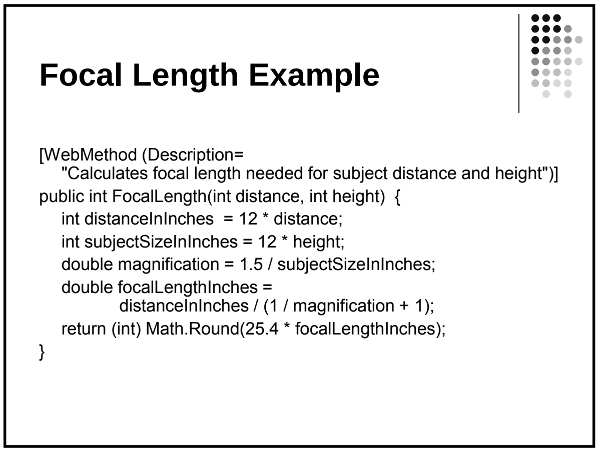
Developing a Web Service

- Implemented in ASP.NET
- Similar to Web Forms, but
  - have `.asmx` file extension
  - contains code, without UI
- Can have a code-behind
- ASP.NET provides simple test harness
- ASP.NET automagically generates WSDL



Creating a Web Service

- In Visual Studio:
  - Create a new project (ASP.NET Web Service)
- Any methods that are to be publicly available over the network must have a [WebMethod] tag



Focal Length Example

```
[WebMethod (Description=
    "Calculates focal length needed for subject distance and height")]
public int FocalLength(int distance, int height)  {
    int distanceInInches  = 12 * distance;
    int subjectSizeInInches = 12 * height;
    double magnification = 1.5 / subjectSizeInInches;
    double focalLengthInches =
            distanceInInches / (1 / magnification + 1);
    return (int) Math.Round(25.4 * focalLengthInches);
}
```

Focal Length Example

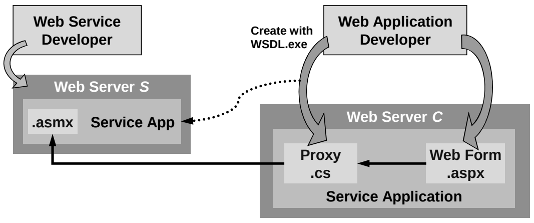
- View in browser to see available methods
- *Service Description* link shows WSDL definition for the service
- *FocalLength* link loads a testing page that shows the SOAP request/response schema and form for testing FocalLength method
- Submit this form to see actual SOAP response

Consuming Web Services

- Locate the desired Web Service
  - UDDI, DISCO
- Get detailed description of Web Service
  - WSDL
- Create a proxy that represents the Web Service
  - Proxy has the same methods/arguments/return values as the Web Service
- Application instantiates and uses the proxy as if it were a local object

Consuming Web Services

Overview



Web Service Developer

Web Server *S*

.asmx Service App

Create with WSDL.exe

Web Application Developer

Web Server *C*

Proxy .cs

Web Form .aspx

Service Application

Consuming Web Services

Overview

- Web Services are URL addressable
  - HTTP request/response
- Can request WSDL via URL
- Can invoke via:
  - HTTP-POST
  - HTTP-SOAP1.1
  - HTTP-SOAP1.2

And Now - REST

Representational State Transfer

(Simpler than SOAP)

- Uses all of the HTTP transfer methods
  - GET (Retrieve)
  - POST (Create)
  - PUT (Update)
  - DELETE (Delete)

Consuming Web Services

Trying It Out

- Request without method name or parameters
  - ASP.NET returns a page listing all methods

`http://localhost/FocalLengthService/FocalLength.asmx`

- Click one of the methods to test it

`http://localhost/FocalLengthService/FocalLength.asmx?op=FocalLength`

  - Specify parameters and Invoke
    - Only for primitive data types
  - Sample requests/responses

Consuming Web Services
Trying It Out

- Request with parameter “WSDL”
  - Formal WSDL description of Web Service
  - XML-based grammar
  - Can be used as input for `wsdl.exe`

http://localhost/FocalLengthService/FocalLength.asmx?WSDL

Consuming Web Services
Using Visual Studio.NET

- Use Add Web Reference to search UDDI or to discover Web Services given a URL
- This builds a proxy, and you can start using the Web Service immediately

Consuming a Web Service

- Make a new ASP.NET Web Application
- In the Solution Explorer, add a *Web Reference*
  - Search on local machine or (UDDI server if online) for the desired service
  - This service can be used by the web application

Example Consumer

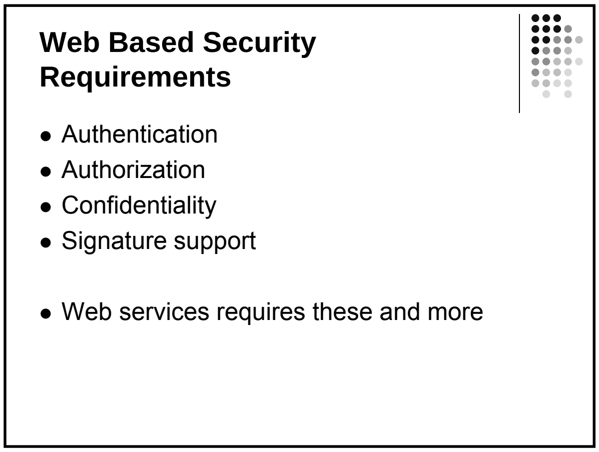
```
private void Button1_Click(object sender,
   System.EventArgs e) {
  LensCalc.LensCalcService myService =
     new LensCalc.LensCalcService();
   int result = myService.FocalLength(
                              Int32.Parse(TextBox1.Text),
                              Int32.Parse(TextBox2.Text));
   Label1.Text = "A " + result + " mm lens should do it";
}
```

Web Services for Developers

- Amazon.com eCommerce services
- Google web services
- Lot’s of publicly available web services
  - Zip Code: http://www.webservicex.net/uszip.asmx
  - Weather: http://www.weather.gov/xml/
  - Commercial UDDI registries: Microsoft, IBM, HP, Sun, …
    (Most have been discontinued?)

Web Service Security

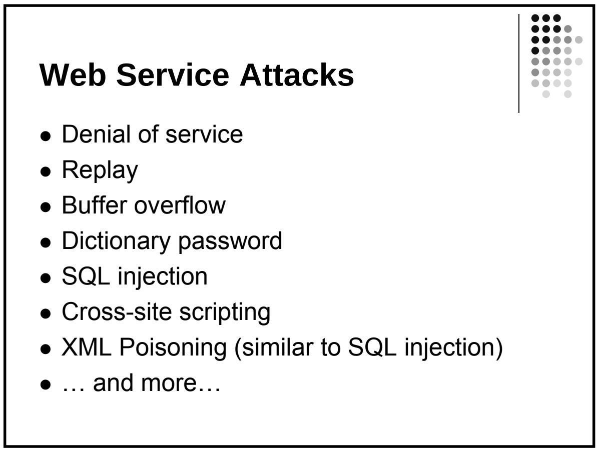
- Web services expose significant new security risks
- They are designed to use the standard http port 80, which easily passes through network firewalls



Web Based Security Requirements

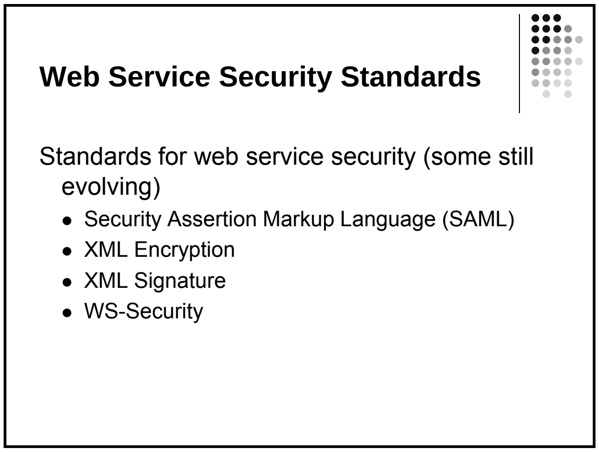
- Authentication
- Authorization
- Confidentiality
- Signature support

- Web services requires these and more



Web Service Attacks

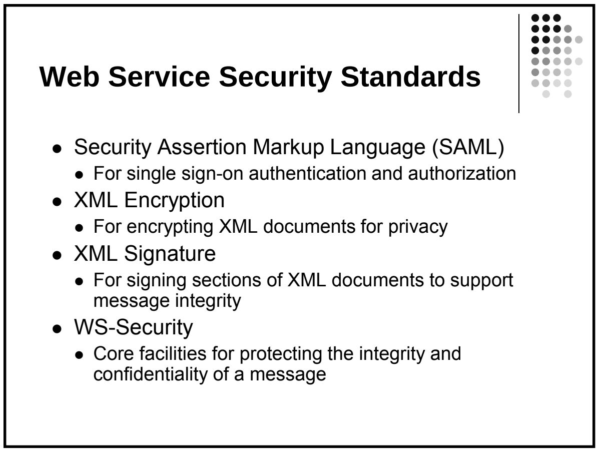
- Denial of service
- Replay
- Buffer overflow
- Dictionary password
- SQL injection
- Cross-site scripting
- XML Poisoning (similar to SQL injection)
- … and more…



Web Service Security Standards

Standards for web service security (some still evolving)

- Security Assertion Markup Language (SAML)
- XML Encryption
- XML Signature
- WS-Security



Web Service Security Standards

- Security Assertion Markup Language (SAML)
  - For single sign-on authentication and authorization
- XML Encryption
  - For encrypting XML documents for privacy
- XML Signature
  - For signing sections of XML documents to support message integrity
- WS-Security
  - Core facilities for protecting the integrity and confidentiality of a message