Welcome to Amsterdam

Web Services Boot Camp

Doug Tidwell

Monday, March 26
8:30 a.m. to 12:00 p.m.

www.xmlworld.org

Web Services Boot Camp: Building Web Services

with stuff you probably have around the office

Doug Tidwell

o developer**Works**

dtidwell@us.ibm.com

ibm.com/developerWorks

Agenda

- Web services overview
- XML underpinnings
- SOAP
- WSDL
- UDDI
- Other considerations
- Free tools

ibm.com/developerWorks

Web services overview

What's this all about?

ibm.com/developerWorks

Web services

- **“We believe that applications will be based on compositions of services discovered and marshaled dynamically at runtime.”**

  — George Washington, in his 1796 Farewell Address to the U.S. Congress

But first, some background

Some more history...

Industry trends

- Open, cross-platform standards are the basis of the Web.
  - TCP/IP (universal networking)
  - HTML (universal rendering)
  - Java (universal code: once written, runs everywhere with great performance and no defects)
  - XML (universal data)

Industry trends

- Content is becoming more dynamic
  - Personalized, up-to-the-minute news, etc.
- Bandwidth is getting cheaper
- Storage is getting cheaper
- Pervasive computing is more important
  - PCs are the most common platform on the Internet, but not for much longer...

Where Web services fit in

- Content is becoming more dynamic
  - Web services will have to gather, interpret, deliver, and/or render information from lots of sources.
- Bandwidth is getting cheaper
  - Web services can deliver XML data **or** streaming audio, video, dynamic VRML, etc.

Where Web services fit in

- Storage is getting cheaper
  - Web services must be able to navigate and filter massive amounts of data. They must also use caching, load balancing, and intelligent techniques to scale.
- Pervasive computing is more important
  - Web services must be able to deliver content to lots of different devices.

What are Web services?

- Functions that are published to, and accessible from, a network
- Transaction-oriented
- Described with XML (WSDL)
- Accessible through service brokers (UDDI)

Behind the technology

- Everything we'll discuss here is based on XML at some point:
  - SOAP
  - WSDL
  - UDDI
- In addition, many Web services will be delivering XML-tagged content; that content will most likely be transformed with XSLT.

Web services architecture

Tying things together

ibm.com/developerWorks

Web services components

- Service provider
  - Delivers services across the network
  - Publishes their services to a broker
- Service requestor
  - Asks the broker for a service
  - Binds to the provider once its found
- Service broker
  - Matchmaker between providers and requestors

Web services operations

- Publish/unpublish
  - Service providers advertise (or not) their services with a service broker
- Find
  - Service requestors ask the broker for a service that meets certain criteria
- Bind
  - Service requestors bind to the service providers, and Web service transactions ensue!

The big idea

- If you've got some code that does something useful (say, a JavaBean, maybe), you can publish that service to a registry.
- If you need some useful function, you can describe what you're looking for and see what the broker finds.
- **All of this can be automated.**

Base technologies

(which we'll talk about in a minute)

- SOAP – Simple Object Access Protocol
- WSDL – Web Services Description Language
- UDDI – Universal Description, Deliver, and Integration

Rub-a-Dub-Dub-Dubya

SOAP and the Web

ibm.com/developerWorks

SOAP

- The Simple Object Access Protocol
- Originally developed by Microsoft, UserLand, developMentor, etc.
- 1.0 perceived as too Windows-centric
- IBM joined the fray to help make SOAP vendor- and platform-neutral
- IBM's second-generation implementation is at xml.apache.org

XML-RPC

- The original efforts that became SOAP were to implement remote procedure calls over XML.
- Not everyone (Bruce Schneier, for example) thinks letting people invoke code through port 80 is a good idea....

Quotable

- **Those pesky firewalls** prevent applications from sending commands to each other, so SOAP lets vendors hide those commands as HTTP so the firewall won't notice.... **Firewalls have good reasons for blocking protocols** like DCOM coming from untrusted sources. **Protocols that sneak them through are not what's wanted**.
  - Bruce Schneier, www.counterpane.com

SOAP design

- Simplicity
- Vendor-neutral
- Language-neutral
- Object-model-neutral
- Transport-neutral

SOAP message structure

- There are request and response messages
  - A request invokes a method on a remote object
  - A response returns the result of running the method

Envelopes

- A SOAP envelope contains the message itself.
- The message is in an application-specific vocabulary; namespaces are used to distinguish the parts.

A SOAP request

```
<SOAP-ENV:Envelope
   xmlns:SOAP-ENV="..."
   SOAP-ENV:encodingStyle="..">
   <SOAP-ENV:Body>
      <m:GetLastTradePrice xmlns:m="my-ns">
         <symbol>IBM</symbol>
      </m:GetLastTradePrice>
   </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

ibm.com/developerWorks

A SOAP response

```
<SOAP-ENV:Envelope
   xmlns:SOAP-ENV="..."
   SOAP-ENV:encodingStyle="..">
   <SOAP-ENV:Body>
      <m:GetLastTradePriceResponse
        xmlns:m="my-ns">
         <price>95.25</price>
      </m:GetLastTradePriceResponse>
   </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

ibm.com/developerWorks

Application view

- If your application is using SOAP to invoke a service, you need to build the request and parse the response.
- You can use XSLT, SAX, DOM, or JDOM to convert the XML data into anything else.
- In some cases, you may not have to process the XML at all…

The Apache SOAP toolkit

- Originally based on IBM's SOAP V1.1 implementation
- Available (as are many great tools) at xml.apache.org.

ibm.com/developerWorks

Deploying a SOAP service

- The toolkit makes this easy…
- Later, we'll talk about more tools from IBM that make deploying a SOAP service even easier.

ibm.com/developerWorks

`Call` and `Response`

- The toolkit makes it easier to deal with SOAP requests and responses:
  - To call a method (request), create a **`Call`** object
  - **`Call.invoke()`** returns a **`Response`** object

Passing parameters

- Here's how to pass parameters to a SOAP method:

```
Vector params = new Vector();
 params.add(new Parameter("database",
  String.class,
  "jdbc:db2:sample", null));
```

- Notice that we pass the name of the parameter, its type, and its value.

Demo time

Presenting the Magic 8-Ball

ibm.com/developerWorks

The Magic 8-Ball…

- …is a cheesy American toy that displays one of 20 answers in response to your heartfelt questions:
  - Reply hazy, ask again later.
  - My sources say no.
  - Without a doubt.
  - As I see it, yes.

The Magic 8-Ball

- “50,000 years of technological advancement has culminated in a system to bring you mysticism on demand.”
- Feel free to shout your questions from the audience....

Magic 8-Ball resources

- **`http://ofb.net/8ball/`**
  - A developer site, really, full of 8-ball arcana, the official phrases, how to take an 8-ball apart, etc.
- **`http://8ball.federated.com/`**
  - My personal favorite: a Linux box, a Magic 8-ball, a Web cam, and a Lego MindStorms robot.

The point of this demo

- We took a piece of code and deployed it through our Web server.
- SOAP makes this possible; anyone that knows our address and the ID of our service can access it.
- That's all well and good if you know our address and service ID to begin with; what if you don't?
  - **That's why we need WSDL and UDDI.**

WSDL

Describing your Web service

ibm.com/developerWorks

Web Services Description Language

- If we're going to find services automatically, we have to have a way to describe them.
  - I describe my service when I publish it
  - You describe what you're looking for when you search
  - We describe our interfaces to each other when we meet.

WSDL sample

```
<binding name="StockQuoteBinding"
  type="StockQuoteType">
  <soap:binding style="rpc"
  transport="schemas.xmlsoap.org..."/>
  <operation name="getQuote">
    <input>
      <soap:body
  type="InMessageRequest"…/>
    </input>
```

WSDL sample

```
    <output>
  <soap:body
  type="OutMessageResponse"...>
    </output>
  </operation>
</binding>
```

ibm.com/developerWorks

Making WSDL disappear

- As tools improve, you won't have to worry much about WSDL.
- WSDL will still be there, it'll just be the underlying standard for representing metadata about Web services.

Demo

- We'll generate a WSDL file from a Java class.
- Notice that this happens automatically; we don't see any angle brackets or other XML syntax.

ibm.com/developerWorks

UDDI

Publish, find, and bind

ibm.com/developerWorks

UDDI

- UDDI is Universal Description, Discovery, and Integration
- Defines standards for a distributed registry of Web Services:
  - White pages (general information)
  - Yellow pages (categories of services)
  - Green pages (business rules)

ibm.com/developerWorks

White pages

- Information about a service provider:
  - Business name
  - Text description (multiple languages)
  - Contact information (phone number, address, etc.)
  - Other names (stock ticker symbols, etc.)

Yellow pages

- Business categories
  - Three taxonomies in UDDI V1:
    - NAICS (U.S. govt industry codes)
    - UN/SPSC (ECMA product/services codes)
    - Geographical information
  - These are implemented as name/value pairs, so they're very extensible.

Green pages

- Information that describes how to work with someone:
  - Business processes
  - Service descriptions
  - Binding information
- Language-, platform-, and implementation-agnostic

ibm.com/developerWorks

Registries in the industry

- To encourage Web services development, IBM, Microsoft, and Ariba have free UDDI registries:
  - www.ibm.com/services/uddi
  - uddi.ariba.com
  - uddi.microsoft.com

DISCO is dead

- ...and so are WDS, NASSL, etc.
- As standards coalesce, lots of early proposals are falling by the wayside.
- IBM, Microsoft, Ariba, and others have all ditched their own work to be part of the standards effort.
- Sun, HP, Oracle, etc. are joining the marketplace; stay tuned to see what/who else bites the dust.

Publish, find, and bind

Some examples

ibm.com/developerWorks

UDDI4J

- An open-source implementation of the client side of UDDI.
- Java classes that make it easy (easier, anyway) to interact with a UDDI registry.
- We'll review the publish, find, bind, and unpublish operations here.

ibm.com/developerWorks

Publishing a service

```
ServiceRegistryProxy srp = new
  ServiceRegistryProxy();
ServiceProvider svcProv = new
  ServiceProvider(providerName,
  providerName, category);
srp.publish(svcProv);
```

- The UDDI classes make this relatively simple.

ibm.com/developerWorks

Finding a service

```
ServiceProvider[]
 serviceProviderList
 =srp.findAllServiceProviders();
```

- Assuming the returned array contains providers, we can access each service provider and its properties individually.

Binding to a service

```
Object[] oArgs;
ServiceProxy sp =
  ServiceProxyFactory.
  getServiceProxy(…)
Response r = sp.invoke(svcName,
  oArgs);
```

ibm.com/developerWorks

Unpublishing a service

```
ServiceProvider svcProv =
  srp.findOwnedServiceProvider(name);
srp.unpublish(svrProv)
```

ibm.com/developerWorks

The New Tools Revue

Comin’ right at you

ibm.com/developerWorks

Tools

- Web Services Toolkit
- XML and Web Services Development Environment
  - Both of these are available at IBM's alphaWorks site.

ibm.com/developerWorks

Other considerations

Points to ponder…

ibm.com/developerWorks

Transformations

- As we move data inside SOAP envelopes, we'll often need to transform that data in a variety of ways.
- We'll take a look at a number of SOAP services that transform XML into a variety of useful things.

XML and security

- We can create an XML document that represents a customer order:

```
<customer_order>
  <item quantity="200">Pencils</item>
  <credit_payment type="Visa">
    <bank>Deutsche Bank</bank>
    <card_num ex_month="12"ex_year="2002">
      3829 2839 2838 9183
    </card_num>
  </credit_payment>
</customer_order>
```

- How do we secure this document?

XML signatures

- With IBM's XML Security Suite, you can digitally sign this document:

```
<Signature>
  <SignedInfo>
  <SignatureMethod Algorithm="…">
  …
  <DigestValue>nsODB51lYrjUIfo25…
```

- Digital signatures are a standard defined by the W3C and the IETF.
- You can digitally sign any resource…

Partial encryption

- To further secure this document, you can encrypt portions of it:

```
<customer_order>
  <item quantity="200">Pencils</item>
  <encrypted_element algorithm="DES5…">
  qktQhEHO5+vLOLAFgIioDIRQGHHmHng3CLd…
  </encrypted_element>
</customer_order>
```

ibm.com/developerWorks

Applications

- The Web Services Toolkit contains a SOAP service that can encrypt and decrypt messages.
- We also have an XSLT stylesheet that uses an extension to encrypt and decrypt XML documents.

The transport layer

- All of our demos here have been built on HTTP. SOAP doesn't require HTTP, however; the latest Web Services Toolkit includes an MQ transport for SOAP.

COM one, COM all

- Just because we're such a friendly, open company, the latest Web Services Toolkit includes COM support.
- You can take a COM object and deploy it as a SOAP service, just as you can deploy a Java Bean…

Resources

Free tools and information
to help you get started

ibm.com/developerWorks

developerWorks zones

- dW features technology zones, including:
  - Java
  - XML
  - Security
  - Linux
  - Open Source
  - Components
  - Web Services

ibm.com/developerWorks

dW's Web services zone

- developerWorks has just launched **`ibm.com/developer/webservices`**, a free portal for Web services resources.
- Free tools, tutorials, links to other sites, sample code, etc.
- Let us know what you'd like to see added to the zone...

ibm.com/developerWorks

Web services tutorial

- There is an “Intro to Web services” tutorial at the developerWorks Web services zone.
- More tutorials are coming soon:
  - SOAP
  - Building Web Services
  - Etc.

ibm.com/developerWorks

www.alphaworks.ibm.com

- Web Services Toolkit
  - Supports UDDI, WSDL, EJBs, and SOAP with digital signatures; Embedded WebSphere AS 3.5 included.
- XML Security Suite
  - Combines XML and world-class security technology

ibm.com/developerWorks

www.alphaworks.ibm.com

- XML and Web Services Development Environment
  - A huge piece of code (87 MB!), but a complete development environment. Contains tools for XML development, database integration, etc., etc.
  - Will be a product at some point, probably

ibm.com/developerWorks

Other vendors

- IBM donated its SOAP implementation to the Apache XML Project (xml.apache.org).
  - As you'd expect, source is included
- Microsoft's SOAP site is msdn.microsoft.com/soap/default.asp
  - As you'd expect, source is **not** included

ibm.com/developerWorks

The Apache XML Project

- **`xml.apache.org`**
- This site contains free parsers, stylesheet engines, server-side XML publishing tools, PDF generators, the SOAP toolkit, etc.
- Everything is free, everything is open source, and quite a bit of it was written by IBM.

UDDI

- **uddi.org**
- IBM's registry:
  - **www.ibm.com/services/uddi/**
- Microsoft's registry:
  - **uddi.microsoft.com/**
- Ariba's registry:
  - **uddi.ariba.com/**

SVG

- If you're interested in Scalable Vector Graphics, see:
  - The spec at **`www.w3.org/TR/SVG/`**
  - You can get Adobe's SVG plug-in at **`www.adobe.com/svg/`**.

Other resources

- ZDNet's review of UDDI:
  - www.zdnet.com/products/stories/reviews/0,4161,2649000,00.html
- DevelopMentor's SOAP site:
  - www.develop.com/soap
- James Snell's SOAP site:
  - www.soap-wrc.com/webservices/default.asp

Let’s get started!

You can change the world…

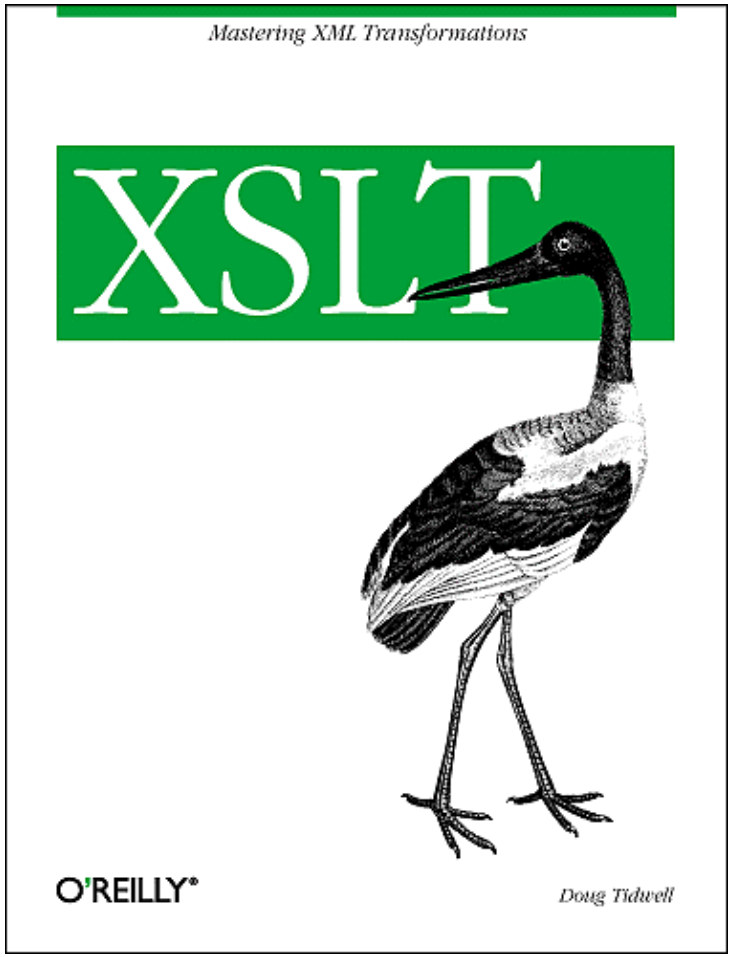
ibm.com/developerWorks

You're all pioneers

- If you've heard of these technologies, you're ahead of most of the industry.
  - Do we know how this will change the Web?
  - Do we have all the business models worked out?
  - Do we know how the software and services industries will change?
- No, no, and no. The rules have changed; go out there and change the world!

ibm.com/developerWorks

Shameless self-promotion…



ibm.com/developerWorks

Thanks for coming!

Doug Tidwell

IBM developer**Works**

dtidwell@us.ibm.com

ibm.com/developerWorks