

Web Services Dev. with Eclipse Web Tools Platform Project

Lawrence Mandel
Jeffrey Liu

IBM Rational Software

Agenda

- Web Services Overview
- Web Service Description Language (WSDL)
- Web Service Communication
- Interoperability
- Java Web Services
- WTP Overview
- WTP Web Service Tools

Web Services Overview

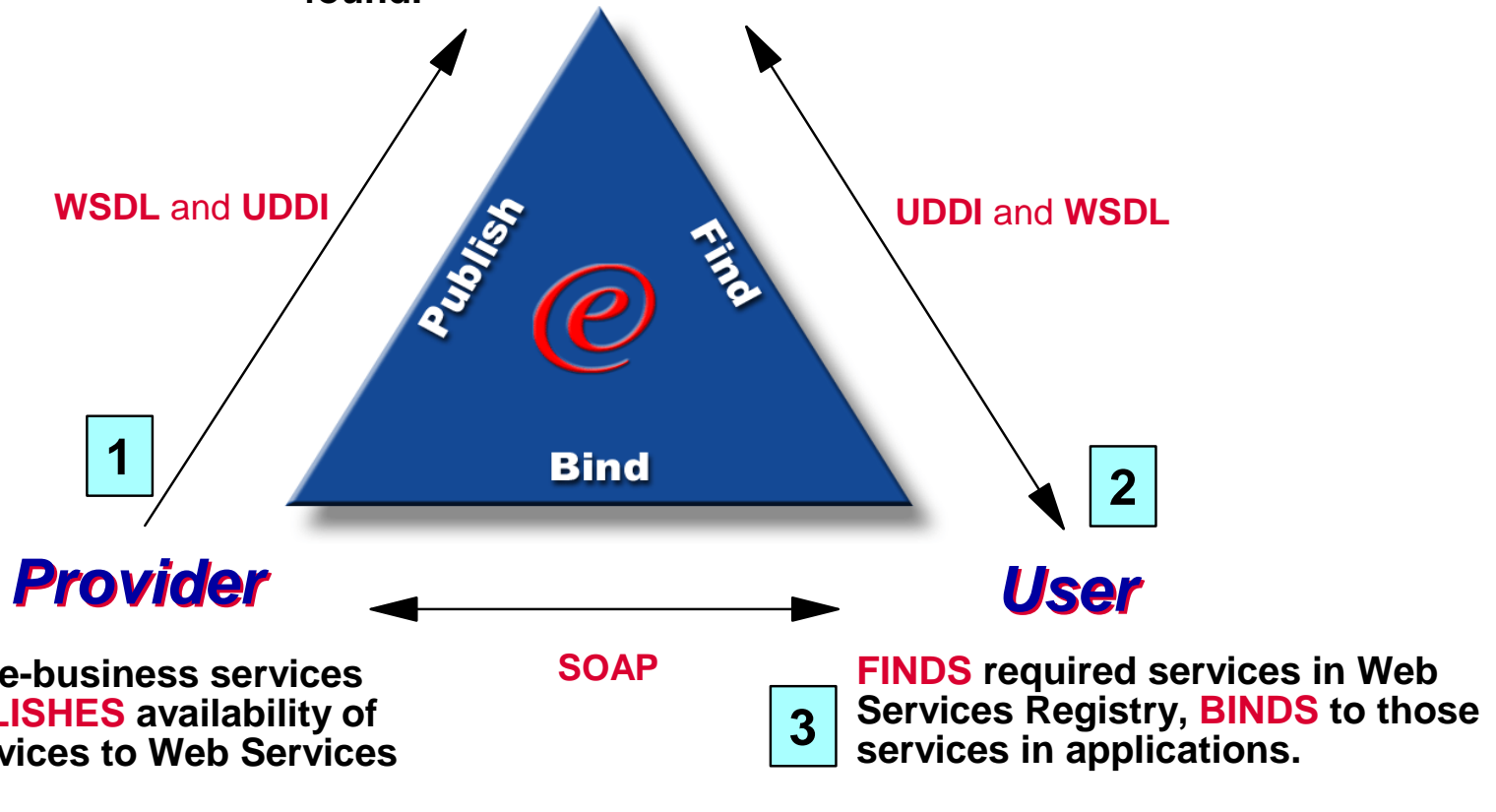
- The invention of the Internet and the World Wide Web has dramatically changed the way we do business.
- Began with information sharing through static documents such as HTML.
- Followed by distributed programs that do transactions over the Web.
- Evolving to Web service-based systems that enable integrations of businesses and processes that run on heterogeneous platforms and environments.



Web Services Overview

Broker

Maintains Web Services Registry where services are published and found.



Web Service Description Language (WSDL)

- From the World Wide Web consortium (W3C)
 - Currently WSDL 1.1 – W3C note
 - WSDL 2.0 – W3C working draft

- Defines the contract between the Web service producer and consumer

- A WSDL document describes
 - Where a Web service is located
 - How to connect to the service
 - What operations are supported by the service
 - What information to send and what information will be returned

Web Service Description Language (WSDL)

- definitions
 - the top level element of the WSDL document
- message
 - describes what will be sent in a message to/from the Web service
- portType
 - defines an abstract interface for a Web service method
- Binding
 - the concrete details about how a portType interface is implemented
- Service
 - specifies where the service is located

Web Service Description Language (WSDL)

- Example

```

<definitions xmlns="http://schemas.xmlsoap.org/wsdl/" name="MyExample"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/"
  targetNamespace="http://tempuri.org/MyExample"
  xmlns:tns="http://tempuri.org/MyExample">
  <message name="MyMessage">
    <part name="MyPart" type="xsd:String"/>
  </message>
  <portType name="MyPortType">
    <operation name="MyOperation">
      <input message="tns:MyMessage"/>
      <output message="tns:MyMessage"/>
    </operation>
  </portType>
  <binding name="MyBinding" type="tns:MyPortType">
    <soap:binding style="rpc" transport="http://schemas.xmlsoap.org/soap/http"/>
    <operation name="MyOperation">
      <input/>
      <output/>
    </operation>
  </binding>
  <service name="MyService">
    <port name="MyPort" binding="tns:MyBinding">
      <soap:address location="http://www.example.com/services/MyService"/>
    </port>
  </service>
</definitions>
  
```

Web Service Communication

- Web services can communicate using a variety of different protocols
 - SOAP, HTTP get and post, JMS, etc.
- SOAP over HTTP is recommended for interoperability

```

<?xml version="1.0" encoding="UTF-8"?>
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <soapenv:Body>
    <logEventRequestElement
      xmlns="http://www.wsi.org/SampleApplications/SupplyChainManagement/2002-
      08/LoggingFacility.xsd">
      <DemoUserID>RetailerClientGUI</DemoUserID>
      <ServiceID>Retailer.submitOrder</ServiceID>
      <EventID>UC1-5</EventID>
      <EventDescription>Order placed by COHMQTCOFI for 605002, 605005,
      605006</EventDescription>
    </logEventRequestElement>
  </soapenv:Body>
</soapenv:Envelope>
  
```

Web Service Communication

```
<?xml version="1.0" encoding="UTF-8"?> WSDL Document
<wsdl:definitions name="Echo"
  xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/"
  xmlns:tns="http://tempuri.org/Echo/"
  xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  targetNamespace="http://tempuri.org/Echo/"
  >
  <wsdl:types>
    <xsd:schema targetNamespace="http://tempuri.org/Echo/"
      xmlns:xsd="http://www.w3.org/2001/XMLSchema"
      >
      <xsd:element name="Request" type="xsd:string" />
      <xsd:element name="Response" type="xsd:string" />
    </xsd:schema>
  </wsdl:types>
  <wsdl:message name="Response">
    <wsdl:part element="tns:Response" name="Response" />
  </wsdl:message>
  <wsdl:message name="Request">
    <wsdl:part element="tns:Request" name="Request" />
  </wsdl:message>
  <wsdl:portType name="Echo">
    <wsdl:operation name="EchoOperation">
      <wsdl:input message="tns:Request" />
      <wsdl:output message="tns:Response" />
    </wsdl:operation>
  </wsdl:portType>
  <wsdl:binding name="EchoSOAP" type="tns:Echo">
    <soap:binding style="document"
      transport="http://schemas.xmlsoap.org/soap/http" />
    <wsdl:operation name="EchoOperation">
      <soap:operation
        soapAction="http://tempuri.org/Echo/NewOperation" />
      <wsdl:input>
        <soap:body use="literal" />
      </wsdl:input>
      <wsdl:output>
        <soap:body use="literal" />
      </wsdl:output>
    </wsdl:operation>
  </wsdl:binding>
  <wsdl:service name="Echo">
    <wsdl:port binding="tns:EchoSOAP" name="EchoSOAP">
      <soap:address location="http://tempuri.org" />
    </wsdl:port>
  </wsdl:service>
</wsdl:definitions>
```

SOAP Request

```
<?xml version='1.0' encoding='UTF-8'?>
<SOAP-ENV:Envelope xmlns:q0="http://tempuri.org/Echo/"
  xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <SOAP-ENV:Body>
    <q0:Request>Hello World!</q0:Request>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

SOAP Response

```
<?xml version="1.0" encoding="UTF-8"?>
<soapenv:Envelope
  xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <soapenv:Body>
    <Response xmlns="http://tempuri.org/Echo/">
      Hello World!
    </Response>
  </soapenv:Body>
</soapenv:Envelope>
```

Interoperability

The Goal of Web services

Interoperability

“Interoperability via evolving open standards is the cornerstone for Web services”

- Rod Smith, Vice President of Emerging Technology, IBM

“Without interoperability, the promise of Web services to provide standards-based business integration will be unrealized.”

- Greg O'Connor, President, Sonic Software

“Interoperable Web services are the foundation of all future enterprise software.”

- Junji Maeyama, Fujitsu

Interoperability

- Services can run on a number of platforms
 - Java (WebSphere, Apache Axis, Apache SOAP, Glue)
 - .Net
- Clients can be written using any platform
 - Java (WebSphere, Apache Axis, Apache SOAP, Glue)
 - .Net

Interoperability means:

Connect **ANY** Client to **ANY** Service

Interoperability

- No longer tied to one platform
 - Business partners will use different platforms
 - May be different platforms within one company
 - Don't want to be locked in to one vendor

- Interoperability facilitates integration
 - Many systems acting together as one

Interoperability

- Myth: A Web service with a conformant WSDL document and conformant SOAP traffic will be interoperable
 - The WSDL and SOAP specifications allow for variability
- There is no way to guarantee 100% interoperability
- Your Web service should be as interoperable as possible

How can I ensure my Web service is as interoperable as possible?

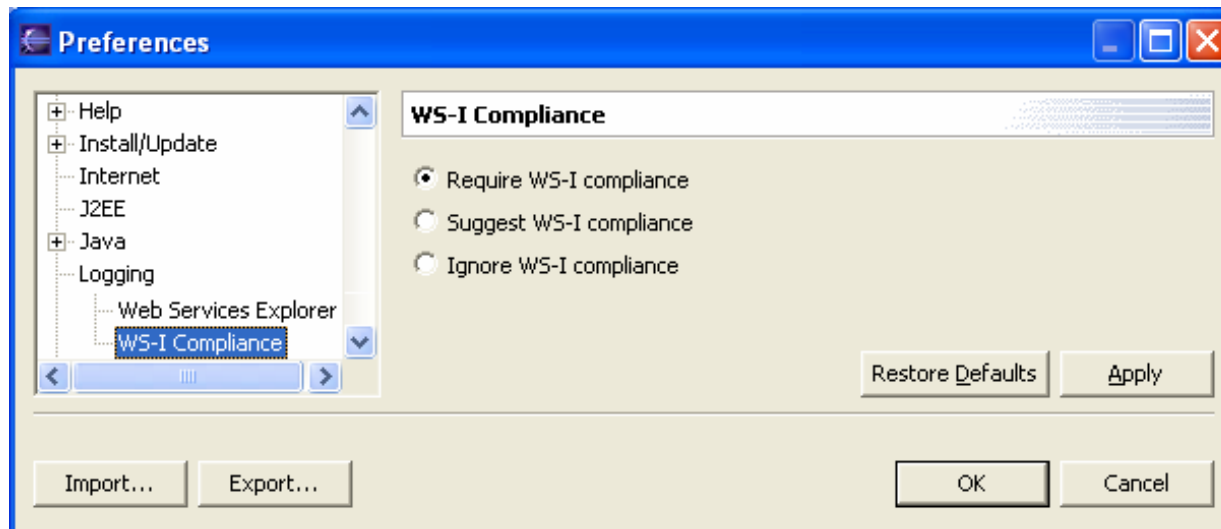
Interoperability

The Web Services Interoperability (WS-I) Organization

- An open, industry organization chartered to promote Web services interoperability.
- Founded February 2002
- 150+ members
 - Software vendors of all sizes
IBM, Microsoft, Oracle, BEA, HP, Sun, SAP, Parasoft, ...
 - Enterprise customers
AT&T, United, Fidelity, Daimler-Chrysler, ...
- WS-I Deliverables
 - **Profiles** – interoperability guidance for developers
 - **Tools** – support the testing of products for Profile conformance
 - Test your WSDL document and Web service SOAP traffic

Interoperability

- The WS-I Test Tools are integrated in the WTP project
- Preference for setting the WS-I conformance level



- Setting the level to Suggest or Require automatically enables WS-I Basic Profile 1.0 Conformance validation in the WSDL validator.

Java Web Services

- A collection of Java technologies
- Describe how Java applications can communicate with other systems using Web services.
- Technologies offer a comprehensive architecture for developing and deploying Web service systems using Java as your programming language.

Java Web Services

- Defines correlations between Web services standards and Java.
- To reinforce the interoperability story.
- To achieve maximum interoperability and portability between Java Web service implementations.
- Ease of Java Web services development.

Java Web Services

Specifications for Java Web Services

- Java API for XML-Based RPC (JAX-RPC, JSR-224, JSR-101)
- Java Architecture for XML Binding (JAXB, JSR-31)
- Implementing Enterprise Web Services (JSR-921, JSR-109)
- SOAP with Attachments API for Java (SAAJ, JSR-67)
- Web Services Metadata for the Java Platform (JSR-181)

WTP Overview

- Two goals
 - 1. Provide Web tools for end users
 - 2. Provide an extensible framework on which additional Web tools can be built

- The WTP project is comprised of two subprojects
 - Web Standard Tools (WST)
 - Industry standard, not programming language specific
 - HTML, SQL, Web services, XML
 - Common tools and frameworks
 - Source editing, validation framework, Web browser and monitor
 - Java Standard Tools (JST)
 - EJB, J2EE including Java Web services

WTP Overview

- Web Standard Tools (WST)
 - Editors and validators for DTD, HTML, WSDL, XML, XML Schema
 - Web browser
 - TCP/IP monitor
 - Database connectivity
 - Server control

- Java Standard Tools (JST)
 - EJB creation
 - JSP editor and validator
 - JBoss and Tomcat server control and deploy
 - Java Web services Wizard and generation tools

WTP Overview

- IBM and ObjectWeb provided the initial contribution of tools
- Project leads, committers and contributors from

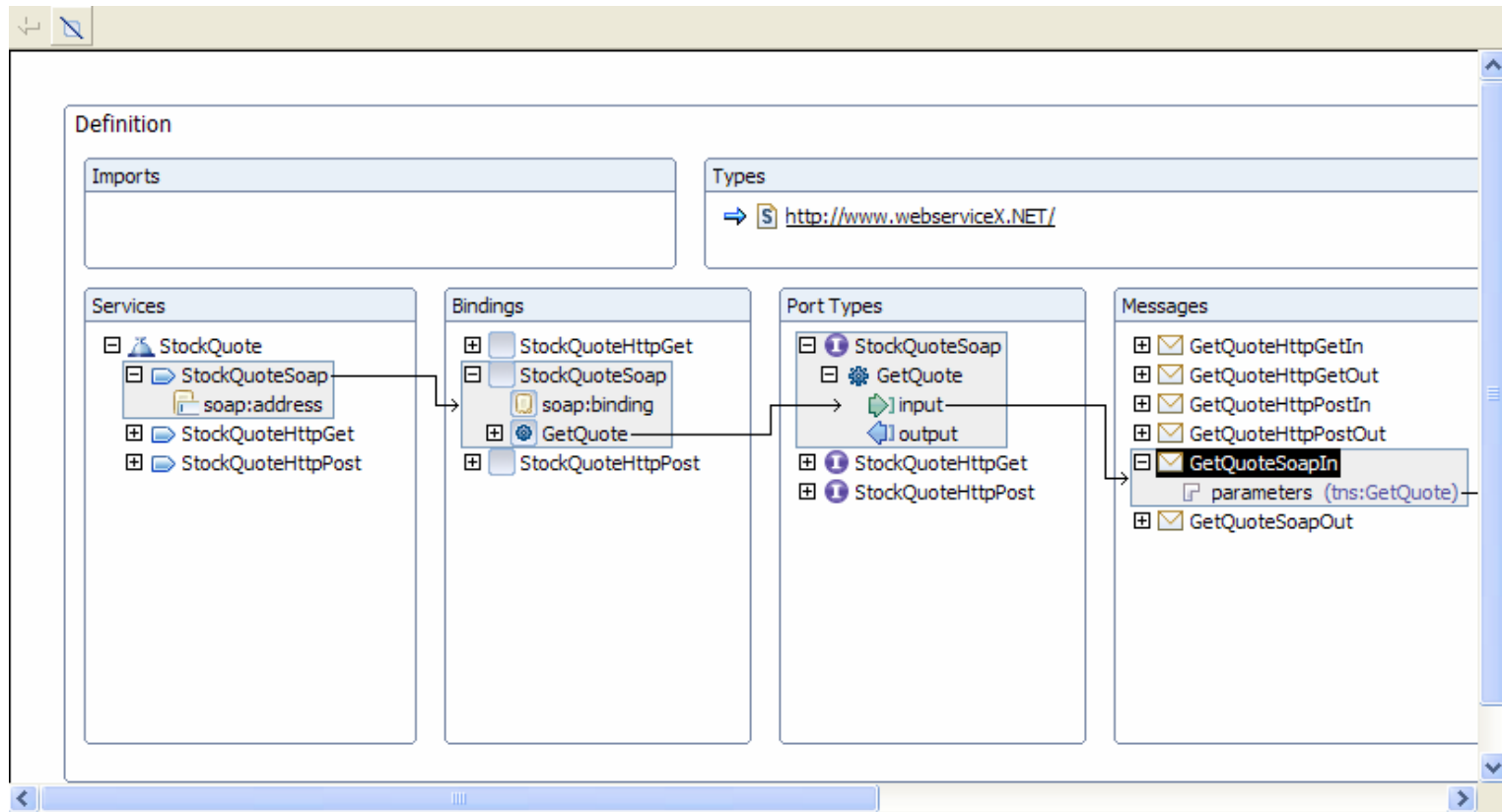


WTP Web Services Tools

- Graphical WSDL/XML Schema Editor (authoring)
- XML Schema, WSDL and WS-I validators (authoring)
- Axis 1.1 Tooling (deployment)
- Web Services Explorer (testing)

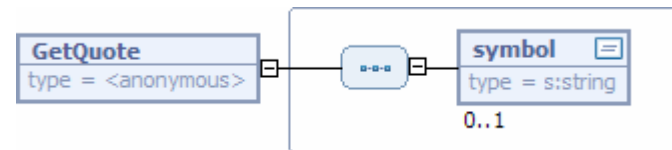
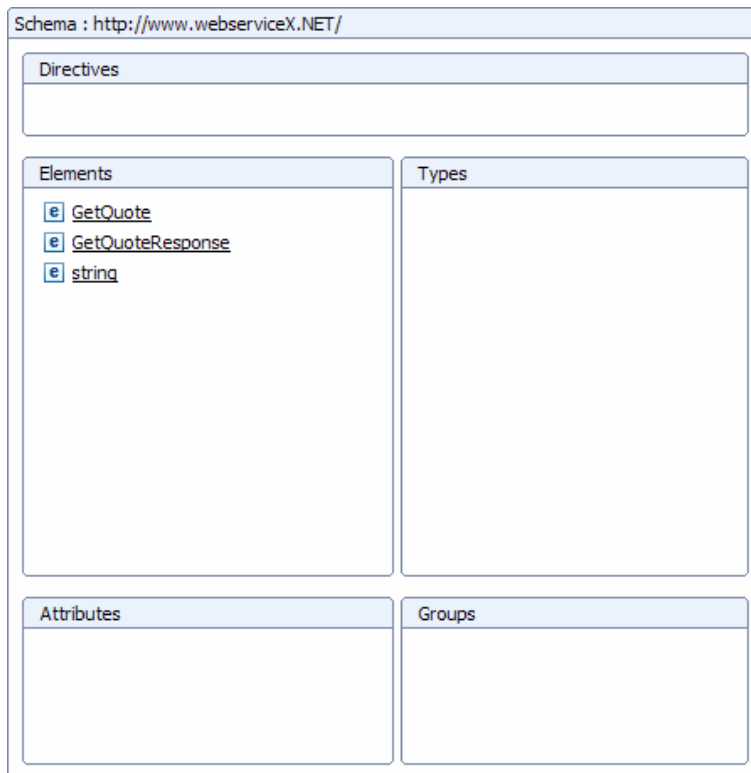
Graphical WSDL/XSD Editor

- Edit your WSDL file without wrestling with the syntax

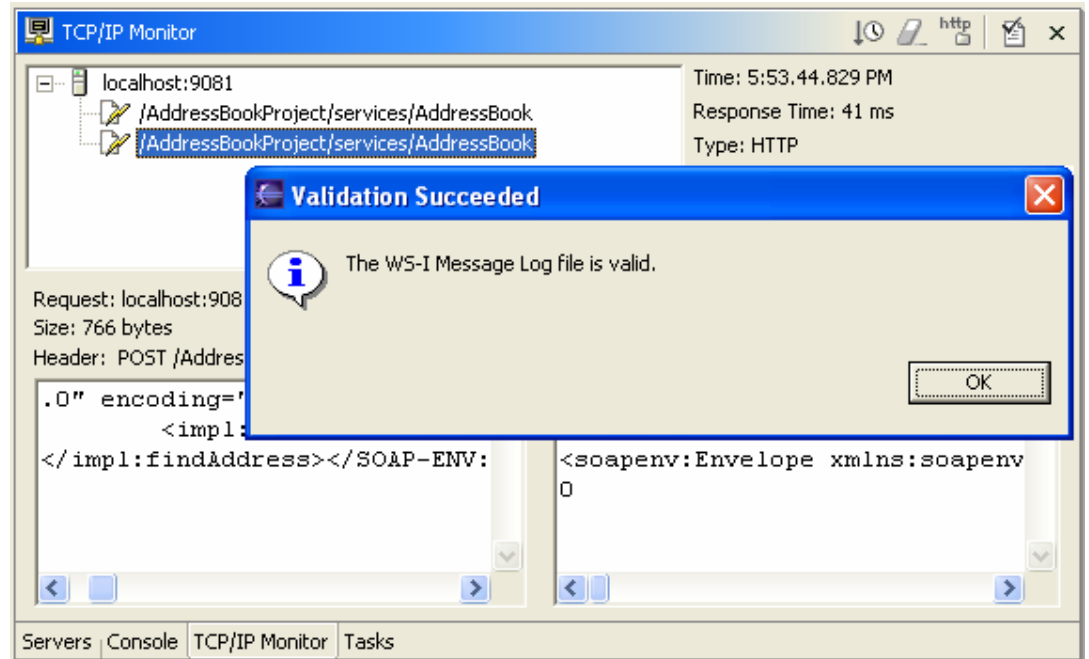
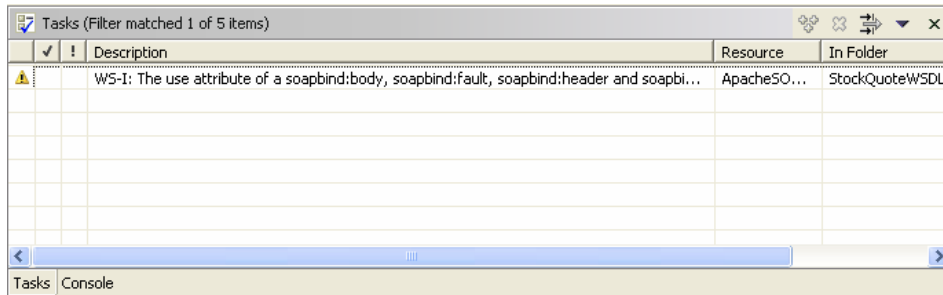


Graphical WSDL/XSD Editor

- Edit your XML schema without wrestling with the syntax

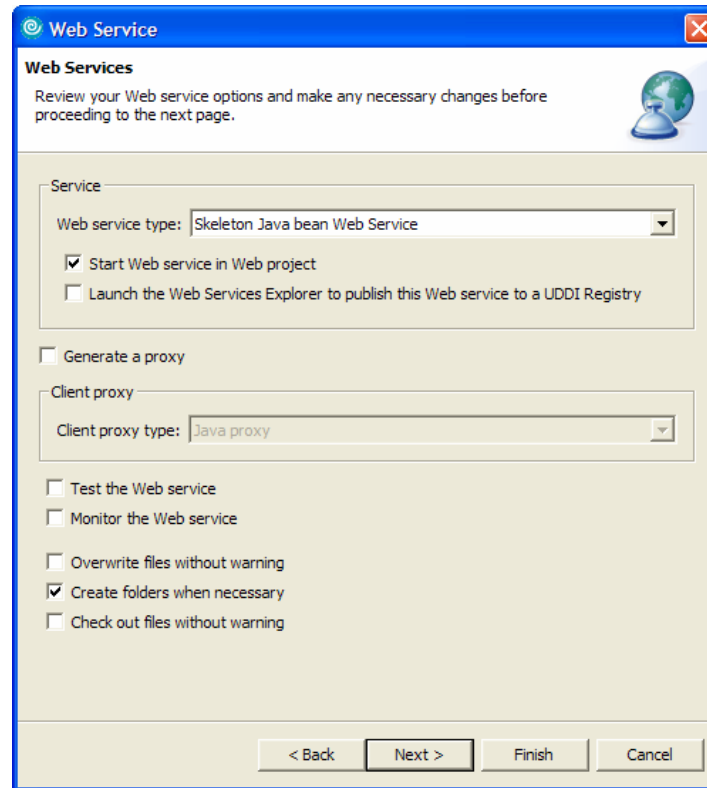


XML Schema, WSDL, and WS-I validators



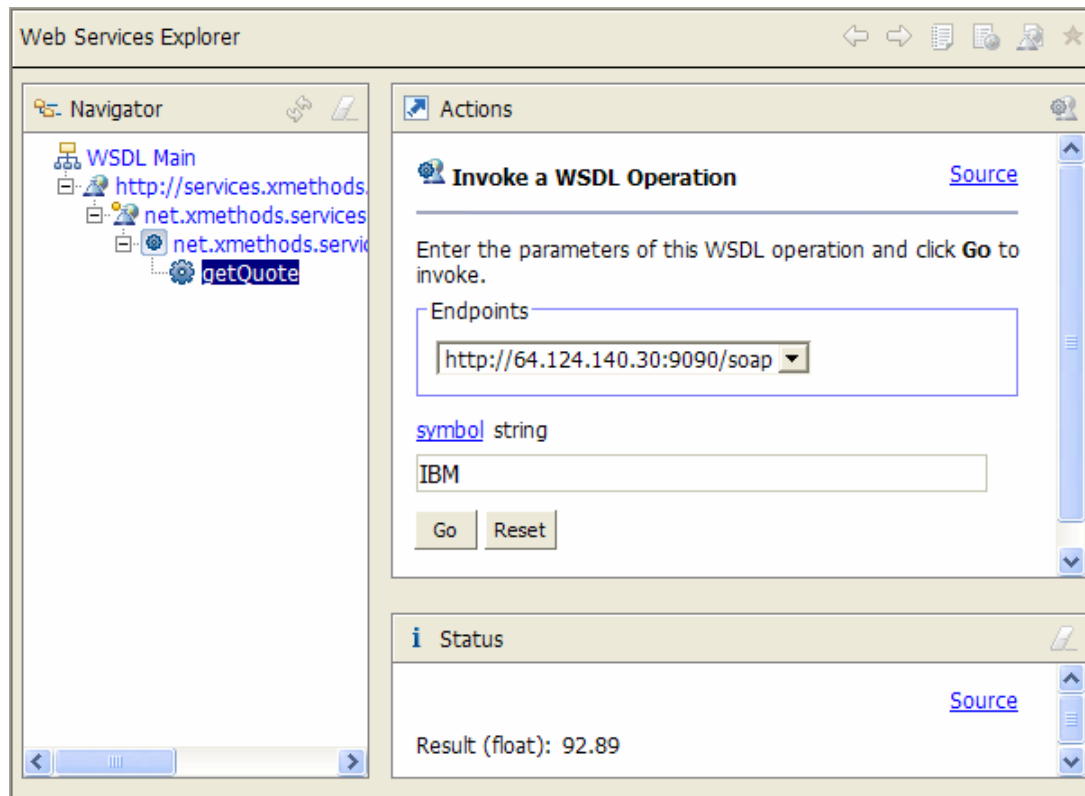
Axis 1.1 Tooling

- Wizard to create Web service top-down (from WSDL) and bottom-up (from Java).
- Wizard to create a Java stub that binds to a Web service.



Web Services Explorer

- Operates purely in Web services domain. Deals with core Web services standards.
- Publish/Discover Web services.
- Invoke Web services dynamically. No code generation required.



Questions?

- Eclipse Web Tools Platform project website:

<http://www.eclipse.org/webtools>

- Lawrence Mandel – lmandel@ca.ibm.com
- Jeffrey Liu – jeffliu@ca.ibm.com