Web Services Middleware

Ivan Hernández

University of Toronto

ECE1770, Spring 2007
A Web service is a software system designed to support interoperable machine-to-machine interaction over a network.

- Interoperate using XML-based standards
- Self-contained and self-describing
- XML + HTTP
- Platform, object model, and programming language independent
- Basic standards SOAP, WSDL, and UDDI
SOAP

- SOAP 1.2 is a W3C standard for exchanging XML-based messages over a computer network.
- It defines a message structure
- Message exchange using a variety of underlying protocols
- It defines the rules for processing a SOAP message
- SOAP defines message exchange patterns
- Why SOAP anyway? Simple Object Access Protocol (SOAP) is a standard for exchanging XML-based messages over a computer network, defined by the World Wide Web Consortium (W3C). It allows for the exchange of messages using a variety of underlying protocols and defines the rules for processing these messages. SOAP provides a framework for message exchange patterns, which can be used in different scenarios.
Skeleton SOAP Message

```xml
<?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:Header>
  ...
</soap:Header>

<soap:Body>
  ...
  <soap:Fault>
    ...
  </soap:Fault>
</soap:Body>

</soap:Envelope>
```
SOAP Examples

- **SOAP Request**
  ```xml
  <?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"
  xmlns:m="http://www.foo.org/prices">
    <soap:Body>
      <m:GetPrice xmlns:m="http://www.foo.org/prices">
        <m:Item>Apples</m:Item>
      </m:GetPrice>
    </soap:Body>
  </soap:Envelope>
  ```

- **SOAP Reply**
  ```xml
  <soap:Body>
    <m:GetPriceResponse xmlns:m="http://www.foo.com/prices">
      <m:Price>2</m:Price>
    </m:GetPriceResponse>
  </soap:Body>
  ```
WSDL

- Web Service Description Language using an XML document
- It specifies the location of the service
- It specifies the methods that the service exposes
- WSDL 2.0 about to become a W3C recommendation
WSDL basic structure

<definitions>

<types>
   Definition of types used, XMLSchema
</types>

<message>
   Definition of parts of each message and the data elements
</message>

<portType>
   Definition the operations performed and involved messages
</portType>

<binding>
   Communication protocol used
</binding>

</definitions>
WSDL Example

<message name="getTermRequest">
  <part name="term" type="xs:string"/>
</message>

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

<portType name="glossaryTerms">
  <operation name="getTerm">
    <input message="getTermRequest"/>
    <output message="getTermResponse"/>
  </operation>
</portType>
<binding type="glossaryTerms" name="b1">
<soap:binding style="document"
transport="http://schemas.xmlsoap.org/soap/http" />
<operation>
  <soap:operation
    soapAction="http://example.com/getTerm"/>
<input>
  <soap:body use="literal"/>
</input>
<output>
  <soap:body use="literal"/>
</output>
</operation>
</binding>
Universal Discovery, Description, and Integration

Service providers use UDDI to advertise the services they offer.

Service Requesters use UDDI to discover services that suit their requirements
Putting It All Together

- **UDDI** advertise the services by providing WSDLs

- **WSDL** specifies location and methods of the service

- **SOAP** defines the messages format and processing

Extensible standards to define, publish and use Web Services.
How are WS related to middleware?

Definition

Middleware systems are comprised of abstractions and services to facilitate the design, development, integration and deployment of distributed applications in heterogeneous networking environments.

- This is exactly what Web Service are about!
- XML + HTTP provide a uniform and widely accessible interface!
Axis2

- Apache’s core engine for Web services
- Fast, low memory footprint, flexibility, stability, ..., and extensible
- Core engine
This is nice, but . . .

- What about reliability, transactions, security, policies ... ?
- SOAP does not define any of these :( 
- But it is extensible! :)

Ivan Hernández (UofT)
Web Services
January 22, 2007 | 19 / 33
### And more Web Service Platform!

<table>
<thead>
<tr>
<th>Web Service Feature</th>
<th>Description</th>
<th>Tool</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>WS-Notification</strong></td>
<td>Standardise the way WS interact using Notifications or Events. Event Driven Architectures using WS</td>
<td>Subscribe</td>
</tr>
<tr>
<td><strong>WS-Coordination</strong></td>
<td>Coordinate the actions of distributed apps. It enables transaction processing, workflow, and other systems for coordination</td>
<td>Kandula</td>
</tr>
<tr>
<td><strong>WS-BPEL</strong></td>
<td>Business Process Execution Language. Compose WS functionality in the right order</td>
<td>Twister</td>
</tr>
<tr>
<td><strong>WS-Security</strong></td>
<td>Message integrity, confidentiality, and single message authentication</td>
<td>WSS4J</td>
</tr>
<tr>
<td><strong>WS-Policy</strong></td>
<td>To express a set of requirements that have to be met in order to consume a web service</td>
<td>WS-Commons</td>
</tr>
</tbody>
</table>

Using all these together we can do anything that an Enterprise Middleware does!
WS-ReliableMessaging

- Many errors may interrupt a conversation.
- Messages may be lost, duplicated or reordered and lost of volatile state.
- WS-RM is a protocol to deliver messages reliably in the presence of software component, system, or network failures.
- The protocol allows to identify, track, and manage the reliable delivery of messages.
- Transport-independent and extensible
- Delivery assurances: AtMostOnce, AtLeastOnce, ExactlyOnce, and InOrder.
WS-RM: Model and example

Initial Sender

Application Source

Send

RM Source

Transmit

Ultimate Receiver

Application Destination

Deliver

RM Destination

Receive

Transmit

Acknowledge
WS-RM: Model and example
WS-RM: Sandesha2

- Sandesha2 is an implementation for the server and client side
- To use on the server side, just add the module to the Axis2 handler stack
- In the client side ...
WS-RM: Sandesha2, client side

ServiceClient client = new ServiceClient(configContext, null);
// set client options

client.engageModule(new QName("sandesha2"));

Callback cb1 = new TestCallback("Callback 1");
client.sendReceiveNonBlocking(
    getEchoOMBlock("echo1","sequence1"), cb1);

Callback cb2 = new TestCallback("Callback 2");
client.sendReceiveNonBlocking(
    getEchoOMBlock("echo2","sequence1"), cb2);

clientOptions.setProperty(
    SandeshaClientConstants.LAST_MESSAGE, "true");

Callback cb3 = new TestCallback("Callback 3");
client.sendReceiveNonBlocking(
    getEchoOMBlock("echo3","sequence1"), cb3);
Questions?
Discussion

- Additional text-processing and bandwidth introduced by XML
  - CPU to serialize/de-serialize, message and transport encryption, XML-tags for all elements, …
  - Problem solved by technology?
  - Are there XML alternatives?
  - Web Services Invocation Framework. Takes advantage of WSDL’s capability to offer multiple bindings for the same service
  - Trade between performance and highly flexible protocols
Middleware’s success and proliferation has recreated, at a higher level, the very problem it was designed to address.

- Web services provide middleware for middleware

- WS-Security (message level security), WS-Reliable Messaging, WS-Addressing only works for SOAP, not at abstract WSDL level

- Web based services vs Web services, which is better?
  - Each one to solve different problems :)

▶ Each one to solve different problems :)

Discussion (2)
Discussion (3)

- With RMI or CORBA, you get the functionality of a remote class just as you would a local class, but with WS . . .
  - You cannot enforce business logic on the client
  - You cannot access read-only properties objects
  - You cannot serialize datatypes such as HashTable
  - You cannot enforce logic in property getters or setters on objects
- All these differences are because . . . It is all about XML messages rather than objects and methods!
Open Source implementation (Apache) vs proprietary solution (IBM)
References

- Web Services Introduction http://www.w3schools.com/webservices/ws_intro.asp
- SOAP Tutorial http://www.w3schools.com/soap/
- WSDL, http://www.w3.org/TR/wSDL
- UDDI.org http://www.uddi.org