Lecture 24

March 23, 2005

SOAP

- Simple Object Access Protocol
- Same general idea as XML-RPC, but more features:
  - enumerations
  - polymorphism (type determined at run time)
  - user defined data types

SOAP

- SOAP is a lightweight protocol for exchange of information in a decentralized, distributed environment
- Microsoft, UserLand Software, DevelopMentor, IBM, Lotus
- SOAP Version 1.2
  - recommendation by W3C

Why SOAP?

- 4th Wave: Application to Application (B2B).
- Heterogeneous systems must be able to communicate.
- Binary protocols don’t always work:
  - CORBA, DCOM, etc. don’t work well through firewalls
  - nobody can agree on a standard binary format (usually due to platform-related issues)
  - we’re dealing with many heterogeneous environments (MVS, Unix, Windows NT, Linux, PalmOS, etc.)
  - security models differ (Kerberos, NTLM, OSF-DCE)

What Do We Have?

- XML - An extensible framework that is easy-to-use and has a low-cost of entry
- HTTP - Industry accepted transport protocol that is already supported by Enterprise servers and is friendly with firewalls

SOAP

- SOAP is a specification for defining...
  - an encoding style that uses XML to represent information graphs
  - a standard way to move XML with HTTP
  - rules for passing messages
  - error (fault) definition
  - a medium for performing Remote Procedure Calls (RPC)
  - one layer in a multi-layer architecture
Components of SOAP 1.1
- SOAP Envelope
- SOAP Encoding Rules
- SOAP RPC Representation
- HTTP Binding

Simple Example
```
<Envelope>
  <Header>
    <transId>1234</transId>
  </Header>
  <Body>
    <Add>
      <a>3</a>
      <b>4</b>
    </Add>
  </Body>
</Envelope>
```

System Flow
Client → Server

```
<Envelope>
  <Header>
    <transId>1234</transId>
  </Header>
  <Body>
    <Add>
      <a>3</a>
      <b>4</b>
    </Add>
  </Body>
</Envelope>
```

```
<Envelope>
  <Header>
    <transId>1234</transId>
  </Header>
  <Body>
    <AddResponse>
      <c>7</c>
    </AddResponse>
  </Body>
</Envelope>
```

c = Add(a, b)

What’s Missing?
- Avoiding ambiguity in messages
- XML Namespaces
- Describing data types
- XML Schemas
- Binding to a transport
- HTTP header fields

XML Namespaces
- Default namespace
```xml
<Book xmlns="Some-URI">
  <Author>Tom Clancy</Author>
</Book>
```
- Qualifying names with prefixes
```xml
<b:Book xmlns:b="Some-URI">
  <Author>Tom Clancy</Author>
</b:Book>
```

XML Schemas
- Type defines…
  - Hierarchies
  - Structure
  - Constraints
- Simple types
  - Integers, strings, floats, time, etc.
- Compound types
  - Arrays, structures

URI: A string that can contains a string that is assumed unique.
URLs are one form of URI since domain names must be unique!
http://www.somecompany.com/xml
XML Schema Example

```xml
<?xml version="1.0"?>
<schema
    targetNamespace="http://example.com/stockquote.xsd"
    xmlns="http://www.w3.org/1999/XMLSchema">
    <element
        name="TradePriceRequest">
        <complexType>
            <all>
                <element
                    name="tickerSymbol"
                    type="string"/>
            </all>
        </complexType>
    </element>
    <element
        name="TradePrice">
        <complexType>
            <all>
                <element
                    name="price"
                    type="float"/>
            </all>
        </complexType>
    </element>
</schema>
```

Actual SOAP Request

```xml
<SOAP-ENV:Envelope
    xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
    SOAP-ENV:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/">  
    <SOAP-ENV:Header>  
        <t:transId xmlns:t="http://a.com/transaction" t:transId="1234"/>
    </SOAP-ENV:Header>
    <SOAP-ENV:Body>
        <m:Add xmlns:m="http://a.com/Calculator">
            <a xsi:type="integer">3</a>
            <b xsi:type="integer">4</b>
        </m:Add>
    </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

Actual SOAP Response

```xml
<SOAP-ENV:Envelope
    xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
    SOAP-ENV:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/">  
    <SOAP-ENV:Header>  
        <t:transId xmlns:t="http://a.com/transaction" t:transId="1234"/>
    </SOAP-ENV:Header>
    <SOAP-ENV:Body>
        <m:AddResponse xmlns:m="http://a.com/Calculator">
            <c xsi:type="integer">7</c>
        </m:AddResponse>
    </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

Closer Look (Request)

Scopes the message to the SOAP namespace
Establishes the type of encoding that is used within the message
Requests As RPC

- Method element immediately follows <Body>
- Parameters are serialized in left-to-right order
- Normally, xsi:type is used to provide type information (XML Schemas)
- Method element NS qualifies entire message

Closer Look (Request)

- Qualifies transId
- Establishes the interface/method
- Describes parameter types

Requests As RPC

- Method element immediately follows <Body>
- Parameters are serialized in left-to-right order
- Normally, xsi:type is used to provide type information (XML Schemas)
- Method element NS qualifies entire message

Closer Look (Response)

- Metadata element with “Response” appended

Binding to HTTP (Request)

HTTP/1.0 200 OK
Content-type: text/xml
Content-length: nnnn

<SOAP-ENV:Envelope
xmlns:SOAP-ENV=http://schemas.xmlsoap.org/soap/envelope/
SOAP-ENV:encodingStyle=http://schemas.xmlsoap.org/soap/encoding/>

<SOAP-ENV:Header
  xmlns:SOAP-ENV=http://schemas.xmlsoap.org/soap/envelope/>
<SOAP-ENV:Body
  xmlns:SOAP-ENV=http://schemas.xmlsoap.org/soap/envelope/>
<Add xmlns:SOAP-ENV=http://schemas.xmlsoap.org/soap/envelope/>
</SOAP-ENV:Envelope>

Binding to HTTP (Response)

HTTP/1.0 200 OK
Content-type: text/xml
Content-length: nnnn

<SOAP-ENV:Envelope
xmlns:SOAP-ENV=http://schemas.xmlsoap.org/soap/envelope/
SOAP-ENV:encodingStyle=http://schemas.xmlsoap.org/soap/encoding/>

<SOAP-ENV:Header
  xmlns:SOAP-ENV=http://schemas.xmlsoap.org/soap/envelope/>
<SOAP-ENV:Body
  xmlns:SOAP-ENV=http://schemas.xmlsoap.org/soap/envelope/>
</SOAP-ENV:Envelope>
SOAP Faults

HTTP/1.1 200 OK
Content-type: text/xml
Content-length: nnnn

<SOAP-ENV:Envelope
xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
SOAP-ENV:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/">
    <SOAP-ENV:Body>
        <SOAP-ENV:Fault>
            <faultcode>SOAP-ENV:Server</faultcode>
            <faultstring>Internal Application Error</faultstring>
            <detail xmlns:f="http://www.a.com/CalculatorFault">
                <f:errorCode>794634</f:errorCode>
                <f:errorMsg>Divide by zero</f:errorMsg>
            </detail>
        </SOAP-ENV:Fault>
    </SOAP-ENV:Body>
</SOAP-ENV:Envelope>

Security

- SOAP spec says nothing
- Delegated to other levels
  - Transport (HTTPS/SSL)
  - Needs standardization

SOAP + Attachments

- Binding for a SOAP 1.1 message to carried in MIME multipart/related message
  - http://www.w3.org/TR/SOAP-attachments

- Useful for:
  - Long XML payloads
  - Non-XML documents

SOAP + Attachments Example

- Start with a SOAP message...
  <soap:Envelope">
    <soap:Body>
      <Person>
        <Image href="http://example.com/myPict.jpg"/>
      </Person>
    </soap:Body>
  </soap:Envelope>

SOAP + Attachments Example

- Add MIME packaging...
  <soap:Envelope>
    <soap:Body>
      <Person>
        <Image href="http://example.com/myPict.jpg"/>
      </Person>
    </soap:Body>
  </soap:Envelope>
SOAP + Attachments Example

- Add MIME part and link to it...
  - MIME-Version: 1.0
  - Content-Type: Multipart/Related; boundary=MIME_boundary; type=text/xml;
  - start="&lt;soapmsg.xml@example.com&gt;"
  - --MIME_boundary
  - Content-Type: text/xml
  - Content-ID: soapmsg.xml@example.com
  - &lt;soap:Envelope xmlns:soap="&gt;
  - &lt;soap:Body&gt;
  - &lt;Person&gt;
  - &lt;Picture href="cid:myPict.jpg@example.com"/&gt;
  - &lt;/Person&gt;
  - &lt;/soap:Body&gt;
  - --MIME_boundary
  - Content-Type: image/jpg
  - Content-Transfer-Encoding: binary
  - Content-ID: &lt;myPict.jpg@example.com&gt;
  - &lt;binary image&gt;
  - --MIME_boundary--

SOAP in a Nutshell

1. XML for encoding data
   - LANGUAGE OF COMMUNICATION
2. HTTP as transport
   - MEDIUM OF COMMUNICATION
   - "Simplicity over Complexity"

Final Example

- www.weather.com
- float CurrentTemp(zip_code)
- The process

REQUEST Example

POST /Temperature HTTP/1.1
Host: www.weather.com
Content-Type: text/xml
Content-Length: &lt;whatever&gt;
SOAPMethod=CurrentTemp

&lt;SOAP:Envelope xmlns:SOAP="urn:schemas-xmlsoap-org:soap.v1">
  &lt;SOAP:Header&gt;
    &lt;t:Transaction xmlns:t="some-URI"/&gt;
  &lt;/SOAP:Header&gt;
  &lt;SOAP:Body&gt;
    &lt;m:CurrentTemp xmlns:m="some-URI"&gt;
      &lt;zip_code&gt;37919&lt;/zip_code&gt;
    &lt;/m:CurrentTemp&gt;
  &lt;/SOAP:Body&gt;
&lt;/SOAP:Envelope&gt;

RESPONSE Example

HTTP/1.1 200 OK
Content-Type: text/xml
Content-Length: &lt;whatever&gt;

&lt;SOAP:Envelope xmlns:SOAP="urn:schemas-xmlsoap-org:soap.v1">
  &lt;SOAP:Header&gt;
    &lt;t:Transaction xmlns:t="some-URI"/&gt;
  &lt;/SOAP:Header&gt;
  &lt;SOAP:Body&gt;
    &lt;m:CurrentTempResponse xmlns:m="some-URI"&gt;
      &lt;return&gt;42&lt;/return&gt;
    &lt;/m:CurrentTempResponse&gt;
  &lt;/SOAP:Body&gt;
&lt;/SOAP:Envelope&gt;

What Is It Not?

- Object activation
  - who invokes CurrentTemp function?
- Bi-directional communications
- Garbage collection
- Language bindings

wizard.cse.nd.edu www.weather.com


### SOAP: Summary

- It's not something path-breaking
- "Right mix of technology at the right time"
- Structure more important than content
- XML - ASCII of the future
- Step in the right direction for B2B applications