

CmpE 494: Service-Oriented Architectures and Web Services

Pinar Yolum

`pinar.yolum@boun.edu.tr`

Department of Computer Engineering
Boğaziçi University

Part V: WSDL

Web Service Description Language

- XML-Based language for describing services
 - Abstract message formats (like SOAP)
 - Bound to a network protocol such as HTTP
 - Extendible
 - Current standard: Version 1.1
- Usage
 - Publish the WSDL in UDDI Registry
 - Consumer access the WSDL description of the service provider
 - Automatically read the WSDL and bind to the service

WSDL Terminology

- Types: A container for data type definitions
- Message: Definition of contents; typed XML
- Operation: Definition of an action supported by the service
- Port Type: Set of operations supported by an endpoint
- Binding: Concrete protocol and data format specification
- Port: An endpoint with a network address and binding
- Service: A collection of endpoints

WSDL Types

- Data type definitions for exchanged messages
- WSDL doesn't have its own type system
- Default: XSD, but other type systems can be used

```
<types>
  <schema targetNamespace="http://example.com/test.xsd"
    xmlns:xsd="http://www.w3.org/2001/XMLSchema"
    <!-- Define types and possibly elements here -->
  </schema>
</types>
```

WSDL Types

```
<definitions .... >
  <types>
    <schema .... >
      <complexType name="POType">
        <all>
          <element name="id" type="string"/>
          <element name="name" type="string"/>
          <element name="items">
            <complexType>
              <all>
                <element name="item" type="tns:Item" minOccurs="1" maxOccurs="1"/>
              </all>
            </complexType>
          </element>
        </all>
      </complexType>
    </schema>
  </types>
</definitions>
```

WSDL Types

```
<complexType name="Item">
  <all>
    <element name="quantity" type="int"/>
    <element name="product" type="string"/>
  </all>
</complexType>
<complexType name="InvoiceType">
  <all>
    <element name="id" type="string"/>
  </all>
</complexType>

<complexType name="Composite">
  <choice>
    <element name="PO" minOccurs="1" maxOccurs="1" type="t...>
    <element name="Invoice" minOccurs="0" maxOccurs="unbou...>
  </choice>
</complexType>
</schema>
</types>
```

WSDL Definitions

- Contains typed logical parts
- Unique message name within the WSDL document
- Unique part name within each message
- Parts define message content

```
<message name="PO">  
  <part name="composite" type="tns:Composite"/>  
</message>  
</definitions>
```

WSDL Port Types

- Set of abstract operations and abstract messages
- Unique port name inside WSDL document
- Four transmission primitives (operations)
 1. One-way: Endpoint receives a message
 2. Request-response: Endpoint sends a response after receiving the message
 3. Solicit-response: Endpoint sends a message and receives a response
 4. Notification: Endpoint sends a message
- Primitives 2 and 3 can be represented with Primitive 1

Operation Examples

● One-way Operation

```
<wsdl:definitions .... > <wsdl:portType .... > *  
  <wsdl:operation name="nmtoken">  
    <wsdl:input name="nmtoken"? message="qname" />  
  </wsdl:operation>  
</wsdl:portType >  
</wsdl:definitions>
```

● Request-Response Operation

```
<wsdl:definitions .... >  
  <wsdl:portType .... > *  
    <wsdl:operation name="nmtoken" parameterOrder="nmtokens">  
      <wsdl:input name="nmtoken"? message="qname" />  
      <wsdl:output name="nmtoken"? message="qname" />  
      <wsdl:fault name="nmtoken" message="qname" />*  
    </wsdl:operation>  
  </wsdl:portType >  
</wsdl:definitions>
```

WSDL Binding

- Binding of operations and messages to actual protocols
- Defines how a WSDL document will be communicated in another protocol
- Must specify exactly one protocol (e.g., SOAP)
- Should not specify any address information

WSDL Binding Template

```
<wsdl:definitions .... >
  <wsdl:binding name="nmtoken" type="qname"> *
    <!-- extensibility element (1) --> *
    <wsdl:operation name="nmtoken"> *
      <!-- extensibility element (2) --> *
      <wsdl:input name="nmtoken"? > ?
        <!-- extensibility element (3) -->
      </wsdl:input>
      <wsdl:output name="nmtoken"? > ?
        <!-- extensibility element (4) --> *
      </wsdl:output>
      <wsdl:fault name="nmtoken"> *
        <!-- extensibility element (5) --> *
      </wsdl:fault>
    </wsdl:operation>
  </wsdl:binding>
</wsdl:definitions>
```

SOAP Binding (1)

- Put soap:binding element
 - Set SOAP transport protocol (HTTP, SMTP, etc.)
 - Set message style as document or rpc
- Set soap:operation
 - Set soapAction for http
 - Set message style as document or rpc
- Put parts into SOAP body
 - Set use as literal or encoded (encodingStyle, too)
 - Set message style as document or rpc
- Specify soap:faults
 - Corresponds to soap:Details
 - Set attributes similar to soap:Body
- Specify soap:header

WSDL Ports

- Defines an endpoint by a single address for binding (e.g., by soap:address)
- Unique name among all ports in a WSDL document
- Contains a defined *binding* attribute
- Must not specify more than one address

WSDL Services

- Groups one or more ports together
- Unique name among all services in a WSDL document
- Ports do not communicate with each other
- Ports with same port types but different bindings (or addresses) are considered alternatives

WSDL Document

```
<?xml version="1.0"?>
<definitions name="StockQuote"
  targetNamespace="http://example.com/stockquote.wsdl"
  xmlns:tns="http://example.com/stockquote.wsdl"
  xmlns:xsd="http://example.com/stockquote.xsd"
  xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/"
  xmlns="http://schemas.xmlsoap.org/wsdl/">

  <types>
    <schema targetNamespace="http://example.com/stockquote.xsd"
      xmlns="http://www.w3.org/2000/10/XMLSchema">
      <element name="TradePriceRequest">
        <complexType>
          <all>
            <element name="tickerSymbol" type="string"/>
          </all>
        </complexType>
      </element>
    </types>
  </definitions>
```

WSDL Document

```
<element name="TradePrice">
  <complexType>
    <all>
      <element name="price" type="float"/>
    </all>
  </complexType>
</element>
</schema>
</types>
<message name="GetLastTradePriceInput">
  <part name="body" element="xsd1:TradePriceRequest"/>
</message>

<message name="GetLastTradePriceOutput">
  <part name="body" element="xsd1:TradePrice"/>
</message>
```

WSDL Document

```
<portType name="StockQuotePortType">
  <operation name="GetLastTradePrice">
    <input message="tns:GetLastTradePriceInput"/>
    <output message="tns:GetLastTradePriceOutput"/>
  </operation>
</portType>
```

```
<binding name="StockQuoteSoapBinding" type="tns:StockQuotePortType">
  <soap:binding style="document" transport="http://schemas.xmlsoap.org/soap/http"/>
  <operation name="GetLastTradePrice">
    <soap:operation soapAction="http://example.com/GetLastTradePrice"/>
    <input>
      <soap:body use="literal"/>
    </input>
    <output>
      <soap:body use="literal"/>
    </output>
  </operation>
</binding>
```

WSDL Document

```
<service name="StockQuoteService">
  <documentation>My first service</documentation>
  <port name="StockQuotePort" binding="tns:StockQuoteBinding">
    <soap:address location="http://example.com/stockquote"/>
  </port>
</service>

</definitions>
```