AG Heterogene Informationssysteme
Prof. Dr.-Ing. Stefan Deßloch
Fachbereich Informatik
Technische Universität Kaiserslautern

Middleware for Heterogeneous and Distributed Information Systems – Exercise Sheet 6

Wednesday, December 3, 2008 – 10:00 to 11:30 – Room 48-379

Web Service Description Language (WSDL)

The Web Services Description Language (WSDL) provides a model and an XML format for describing web service interfaces. Table 1 shows a (shortened) WSDL document of a publicly available web service that provides current weather information for U.S. cities[1].

1. What information is generally required to call web services? How is this information organized within the WSDL document? What parts are distinguished?
2. What data is provided by the weather web service shown in Table 1? What input parameters are required?
3. What XML related standard besides WSDL is used in the sample WSDL document?

Java API for XML Web Services (JAX-WS)

The Java API for XML Web Services (JAX-WS) provides the means for building web services and web service clients. With JAX-WS it is possible to develop a java client application that invokes the weather web service introduced above.

Describe the required steps to develop the weather service client using JAX-WS! For each step, name the required utility program and the input and output files!

SOAP

The client communicates with the weather service using SOAP over HTTP.

1. Are there alternative techniques that could have been used instead?
2. How are SOAP messages structured in general?
3. What SOAP messages are exchanged when the client application invokes the `GetCityWeatherByZIP` operation and how do they look like?

[1]The service is listed on www.xmethods.net. This web site lists publicly available web services offered by various vendors.

```xml
<?xml version="1.0" encoding="utf-8"?>
<wsdl:definitions xmlns:xs="http://www.w3.org/2001/XMLSchema"
xmlns:tns="http://ws.cdyne.com/WeatherWS/" xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/"
targetNamespace="http://ws.cdyne.com/WeatherWS/" xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">

  <wsdl:types>
    <xs:schema elementFormDefault="qualified" targetNamespace="http://ws.cdyne.com/WeatherWS/">

      <xs:element name="GetCityWeatherByZIP">
        <xs:complexType>
          <xs:sequence>
            <xs:element minOccurs="0" maxOccurs="1" name="ZIP" type="xs:string" />
          </xs:sequence>
        </xs:complexType>
      </xs:element>

      <xs:element name="GetCityWeatherByZIPResponse">
        <xs:complexType>
          <xs:sequence>
            <xs:element minOccurs="1" maxOccurs="1" name="GetCityWeatherByZIPResult"
                type="tns:WeatherReturn" />
          </xs:sequence>
        </xs:complexType>
      </xs:element>

      <xs:complexType name="WeatherReturn">
        <xs:sequence>
          <xs:element minOccurs="1" maxOccurs="1" name="Success" type="xs:boolean" />
          <xs:element minOccurs="0" maxOccurs="1" name="ResponseText" type="xs:string" />
          <xs:element minOccurs="0" maxOccurs="1" name="State" type="xs:string" />
          <xs:element minOccurs="0" maxOccurs="1" name="City" type="xs:string" />
          <xs:element minOccurs="0" maxOccurs="1" name="WeatherStationCity" type="xs:string" />
          <xs:element minOccurs="1" maxOccurs="1" name="WeatherID" type="xs:short" />
          <xs:element minOccurs="0" maxOccurs="1" name="Description" type="xs:string" />
          <xs:element minOccurs="0" maxOccurs="1" name="Temperature" type="xs:string" />
          <xs:element minOccurs="0" maxOccurs="1" name="RelativeHumidity" type="xs:string" />
          <xs:element minOccurs="0" maxOccurs="1" name="Wind" type="xs:string" />
          <xs:element minOccurs="0" maxOccurs="1" name="Pressure" type="xs:string" />
          <xs:element minOccurs="0" maxOccurs="1" name="Visibility" type="xs:string" />
          <xs:element minOccurs="0" maxOccurs="1" name="WindChill" type="xs:string" />
          <xs:element minOccurs="0" maxOccurs="1" name="Remarks" type="xs:string" />
        </xs:sequence>
      </xs:complexType>

    </xs:schema>
  </wsdl:types>

  <wsdl:message name="GetCityWeatherByZIPSoapIn">
    <wsdl:part name="parameters" element="tns:GetCityWeatherByZIP" />
  </wsdl:message>
  <wsdl:message name="GetCityWeatherByZIPSoapOut">
    <wsdl:part name="parameters" element="tns:GetCityWeatherByZIPResponse" />
  </wsdl:message>

  <wsdl:portType name="WeatherPortType">
    <wsdl:operation name="GetCityWeatherByZIP">
      <wsdl:documentation xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">
        Allows you to get your City's Weather, which is updated hourly. U.S. Only
      </wsdl:documentation>
      <wsdl:input message="tns:GetCityWeatherByZIPSoapIn" />
      <wsdl:output message="tns:GetCityWeatherByZIPSoapOut" />
    </wsdl:operation>
  </wsdl:portType>

  <wsdl:binding name="WeatherBinding" type="tns:WeatherPortType">
    <soap:binding transport="http://schemas.xmlsoap.org/soap/http" />
    <wsdl:operation name="GetCityWeatherByZIP">
      <soap:operation soapAction="http://ws.cdyne.com/WeatherWS/GetCityWeatherByZIP"
         style="document" />
      <wsdl:input>
        <soap:body use="literal" />
      </wsdl:input>
      <wsdl:output>
        <soap:body use="literal" />
      </wsdl:output>
    </wsdl:operation>
  </wsdl:binding>

  <wsdl:service name="WeatherService">
    <wsdl:port name="WeatherPortType" binding="tns:WeatherBinding">
      <soap:address location="http://ws.cdyne.com/WeatherWS/Weather.asmx" />
    </wsdl:port>
  </wsdl:service>

</wsdl:definitions>
```

Table 1: Sample Weather Service WSDL