



WSDL

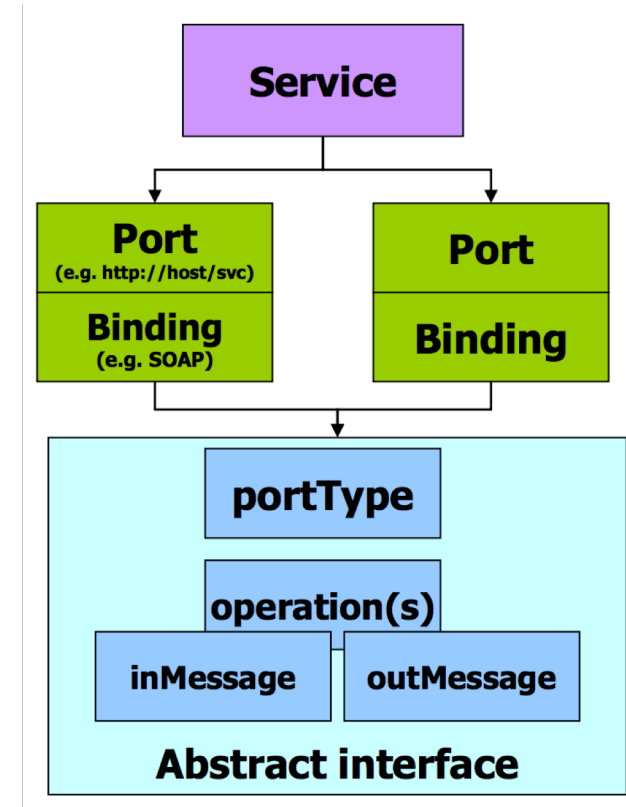
Web Service Definition Language

WSDL Example

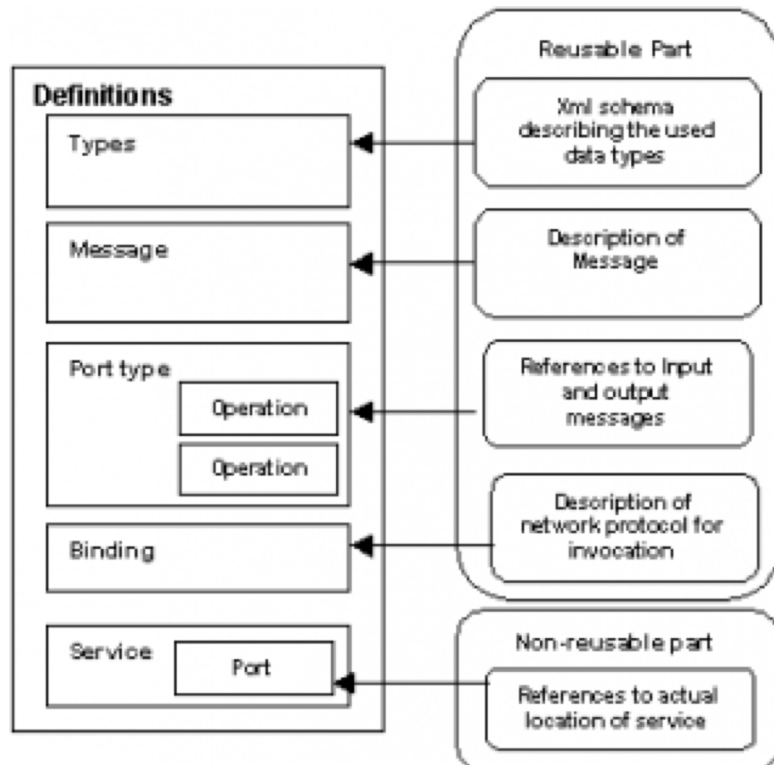
```
<?xml version='1.0' encoding='UTF-8'?><!-- Published by JAX-WS RI (http://jax-ws.java.net). RI's version is Meti
<types>
  <xsd:schema>
    <xsd:import namespace="http://wservice/" schemaLocation="http://localhost:8080/ServerWS/Hello?xsd=1" />
  </xsd:schema>
</types>
<message name="hello">
  <part name="parameters" element="tns:hello" />
</message>
<message name="helloResponse">
  <part name="parameters" element="tns:helloResponse" />
</message>
<portType name="Hello">
  <operation name="hello">
    <input wsam:Action="http://wservice/Hello/helloRequest" message="tns:hello" />
    <output wsam:Action="http://wservice/Hello/helloResponse" message="tns:helloResponse" />
  </operation>
</portType>
<binding name="HelloPortBinding" type="tns:Hello">
  <soap:binding transport="http://schemas.xmlsoap.org/soap/http" style="document" />
  <operation name="hello">
    <soap:operation soapAction="" />
    <input>
      <soap:body use="literal" />
    </input>
    <output>
      <soap:body use="literal" />
    </output>
  </operation>
</binding>
<service name="Hello">
  <port name="HelloPort" binding="tns:HelloPortBinding">
    <soap:address location="http://localhost:8080/ServerWS/Hello" />
  </port>
</service>
</definitions>
```

WSDL

- Describes an **interface** for consuming a **web service**
- Analogue to typical Java interface, but written in XML
- Contains description of
 - **interface as a whole**
 - **methods** in it
 - **input** parameters and **output** results for each method
 - all input parameters of one method are packed in one message
 - the result of a method is packed in another message
 - **types** of the input and output parameters
 - primitive types – int, double, ...
 - composite types
 - way of **binding** a client to the service
 - transport protocol
 - required input and output



WSDL Format



- **<types>** – a tag container for data types definition using some standard, such as **XSD**
- **<message>** – for definition of the data being communicated including its type
- **<operation>** – an abstract description of an action supported by the service
- **<portType>** – an abstract **set of operations** supported by one or more endpoints
- **<binding>** – a concrete **protocol and data format** specification for a particular port type
- **<port>** – a single **endpoint** defined as a combination of a **binding and a network address**
- **<service>** – a collection of related **endpoints**

XML Schema

- Describes the structure of the WSDL document
- Provides a data type for each SOAP message that travels in either direction, between the service and the client
- **Messages** are described as of a `complexType`, opposed to a simple XML types such as `xsd:date`, `xsd:string`, `xsd:integer`
- Each **operation** is described by **two messages**: **request** and **response**
- Special error messages, known as **SOAP faults**, may also be defined

WSDL Schema Definition Example

```
<?xml version="1.0"?>
<definitions name="StockQuote" targetNamespace="http://example.com/stockquote.wsdl"
    xmlns:tns="http://example.com/stockquote.wsdl"
    xmlns:xsd1="http://example.com/stockquote.xsd"
    xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/"
    xmlns=http://schemas.xmlsoap.org/wsdl/>
    <types> ... </types>
    <message> ... </message>
    <portType>
        <operation> ... </operation>
    </portType>
    <binding>
        <operation> ... </operation>
    </binding>
    <service>
        <port> ... </port>
    </service>
</definitions>
```

Web Service Correspondence to OOP

WSDL	Application
<code><portType></code>	Class
<code><operation></code>	Method
<code><message></code>	Method parameter or return value
<code><types></code>	Data types (e.g., int, double, String, etc.)

WSDL portType Example

Port Type - an abstract **set of operations** supported by one or more endpoints

- XML representation of an application class, which provides the web service

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


WSDL *types* Example

Type - a tag container for data types definition using some standard, such as XSD

- XSD => XML Schema Description
- XML representation data types, used in the web service description in the application, providing the web service

<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>
  <element name="TradePrice">
    <complexType>
      <all>
        <element name="price" type="float"/>
      </all>
    </complexType>
  </element>
</schema>
</types>
```

WSDL *message* Example

Message - for definition of the input and output data being communicated including its type

- one message for the request and one for the response
- sum of parts (name + type)
- XML representation of method invocation in the application, which provides the web service

```
<message name="GetLastTradePriceInput">
  <part name="body" element="xsd1:TradePriceRequest"/>
</message>

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

WSDL **binding** Example

Defines the concrete communication **protocol and data format** specification for each port type

```
<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 **port** and **service** Example

Port - a single **endpoint** defined as a combination of a **binding** and a **network address**

- **address for binding**
- **Service** - a collection of related **endpoints**
 - sum of ports
 - XML representation of the whole - all web services, provided by the application

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

Java vs WSDL

Java Interface

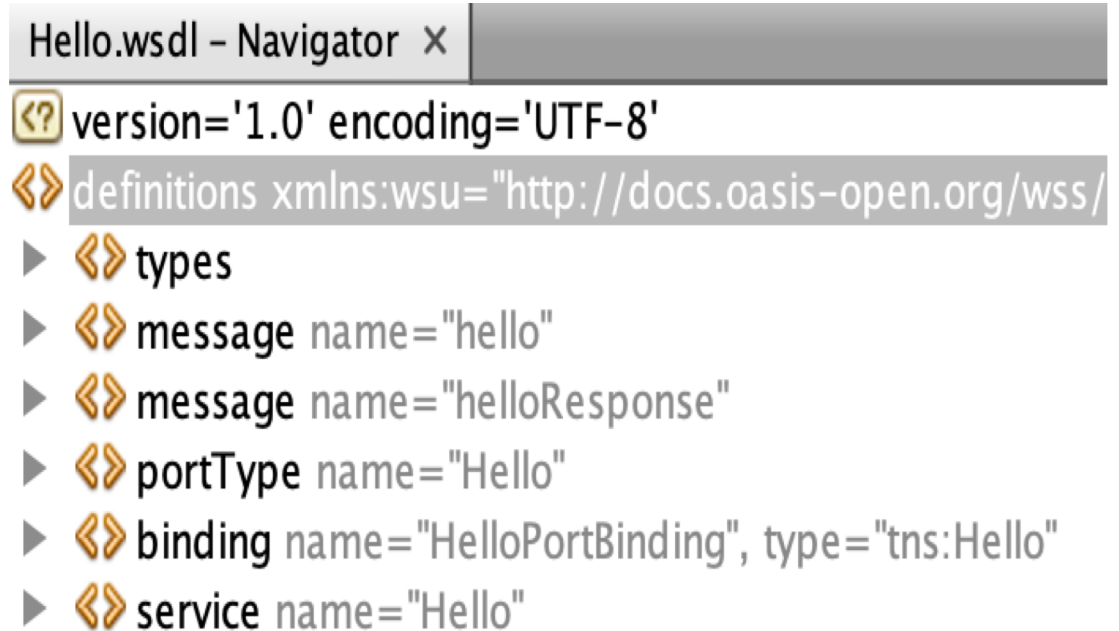
```
public interface Hello
{
    String hello(String name);
}
```

Java Interface Implementation

```
package wservice;
import javax.ws.WebService;
import javax.ws.WebMethod;
import javax.ws.WebParam;
/**
 * This is a sample web service
 */
@WebService(serviceName = "Hello")
public class Hello
{
    /**
     * This is a sample web service operation
     */
    @WebMethod(operationName = "hello")
    public String hello(@WebParam(name = "name") String name)
    {
        return "Hello " + name + " !";
    }
}
```

12/09/2021

WSDL Interface



The screenshot shows a WSDL file named 'Hello.wSDL' in a 'Navigator' window. The XML content is as follows:

```
<?xml version='1.0' encoding='UTF-8'>
<definitions xmlns:wsu="http://docs.oasis-open.org/wss/">
  <types/>
  <message name="hello"/>
  <message name="helloResponse"/>
  <portType name="Hello"/>
  <binding name="HelloPortBinding", type="tns:Hello"/>
  <service name="Hello"/>
</definitions>
```

tdi@cphbus

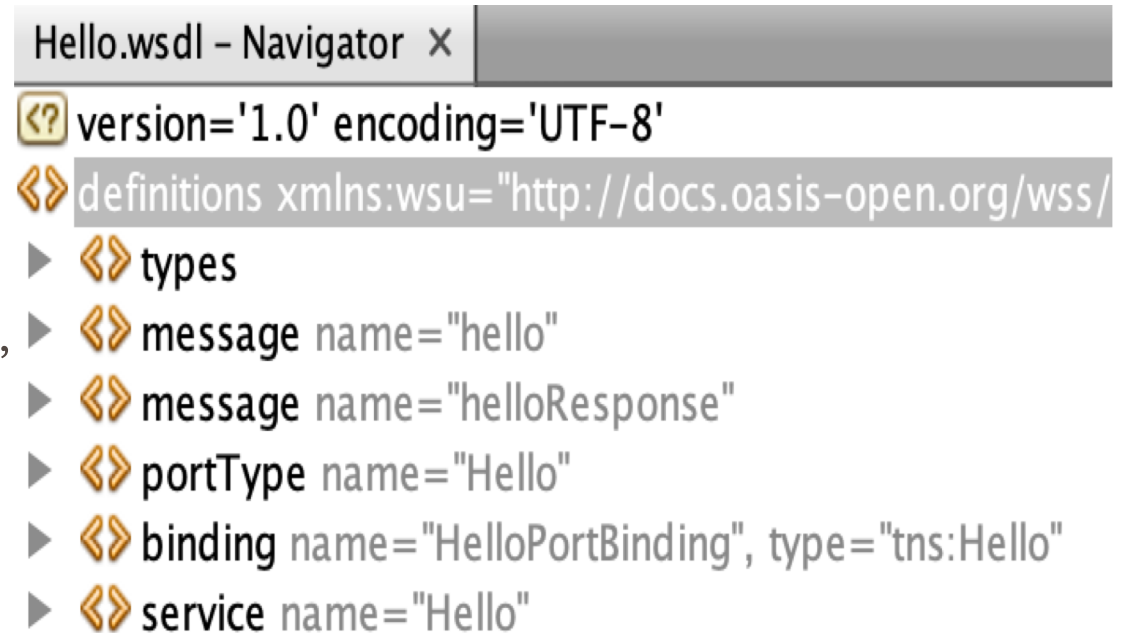
13

Java vs WSDL

interface => service
methods => portType, operations
data types, like String => types
input parameters => message "hello"
output result => message "helloResponse"

binding provides info on how to connect with the service, such as communication protocol and data format per port type

WSDL Interface



The screenshot shows a web browser window titled "Hello.wSDL - Navigator". The content displays the XML structure of a WSDL file. The root element is `definitions` with attributes `version='1.0'` and `encoding='UTF-8'`. It includes a namespace `xmlns:wsu="http://docs.oasis-open.org/wss/". The definitions element contains several child elements: types, message name="hello", message name="helloResponse", portType name="Hello", binding name="HelloPortBinding", type="tns:Hello", and service name="Hello".`

```
version='1.0' encoding='UTF-8'
definitions xmlns:wsu="http://docs.oasis-open.org/wss/
  types
  message name="hello"
  message name="helloResponse"
  portType name="Hello"
  binding name="HelloPortBinding", type="tns:Hello"
  service name="Hello"
```