Web Service Description Language (WSDL)

Prof. (Dr.) Vipul K. Dabhi
Assoc. Professor,
Department of Information Technology,
D. D. University

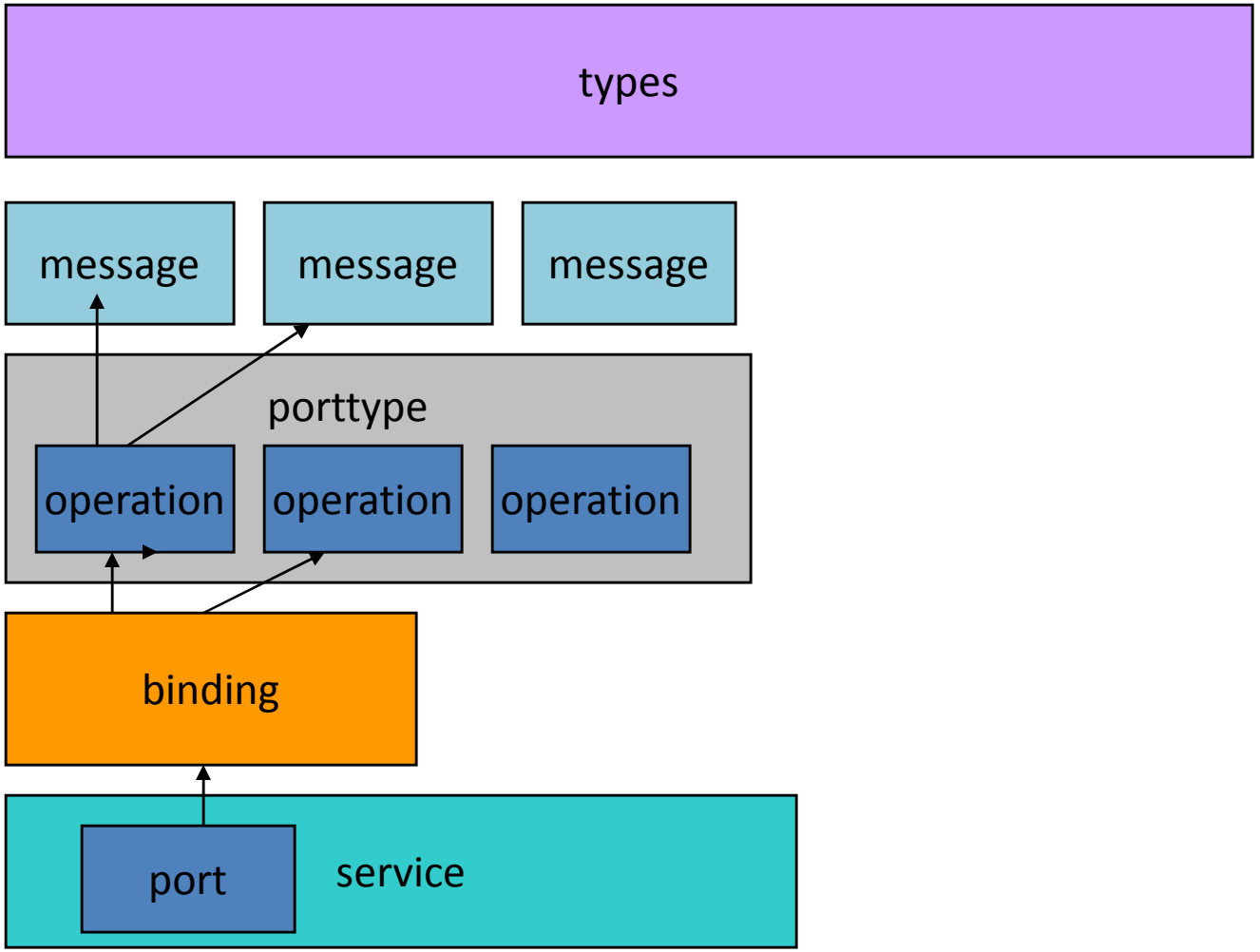
What is WSDL?

- Web Service Description Language
- WSDL is a document written in XML
- The document describes a Web service
- Specifies the location of the service and the methods the service exposes

Why WSDL?

- Without WSDL, calling syntax must be determined from documentation that must be provided, or from examining wire messages
- With WSDL, the generation of proxies for Web services is automated in a truly language- and platform-independent way

WSDL Structure



WSDL Structure

<definitions>: Root WSDL Element

- <types>: What data types will be transmitted?
- <message>: What messages will be transmitted?
- <portType>: What operations will be supported?
- <binding>: How will the messages be transmitted on the wire?
- <service>: Where is the service located?

WSDL Document Structure

- Written in XML
- Two types of sections
  - Abstract and Concrete
- *Abstract* section define SOAP messages in a platform- and language-independent manner
- Site-specific matters such as serialization are defined in the *Concrete* section

Abstract Definitions

- Types:
  - Machine- and language-independent type definitions.
- Messages:
  - Contains function parameters (inputs are separate from outputs) or document descriptions.
- PortTypes:
  - Refers to message definitions in Messages section that describe function signatures (operation name, input parameters, output parameters).

Concrete Descriptions

- Bindings:
  - Specifies binding(s) of each operation in the PortTypes section.
- Services:
  - Specifies port address(es) of each binding.

WSDL: Types Section

- The *type* element defines the data types that are used by the web service.
- <types>

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

WSDL: Message Section

- The <**message**> element is used to define the messages that will be exchanged between the client and the service.
- Messages consist of one or more logical parts <**part**> elements, which use types defined in the types element.
- The name of an output message element ends in "Response" by convention.

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

WSDL: PortTypes Section

- Defines a web service, the operations that can be performed, and the messages that are involved.
- A portType is analogous to a class
- An operation is analogous to a method in that class

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

WSDL: Binding Section

- Binding
  - The binding element provides concrete protocol and data format for a particular PortType operation
    - Protocol examples: SOAP over HTTP or SOAP over SMTP
- <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> </input>
  <output> </output>
  </operation>
  </binding>

WSDL : Service Section

- An endpoint is defined as a combination of a binding and an address. The <service> element defines <port> elements that specify where requests should be sent.
  - The <soap:address> sub-element identifies the URL of the service
  - <service name="StockQuoteService>
    <port name="StockQuotePort" binding="tns:StockQuoteSoapBinding">
    <soap:address location=http://example.com/stockquote"/>
    </port>
    </service>