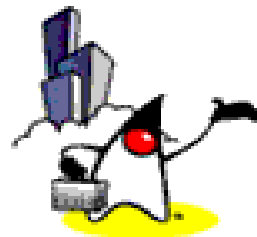




WSDL Basics (Web Services Description Language)

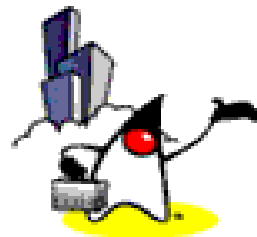


Agenda

- What and Why WSDL?
- Example WSDL Document
- WSDL Document Elements
- Importing & authoring style
- Application design & Tools
- Limitations of WSDL
- WSDL 1.2



What is and Why WSDL?

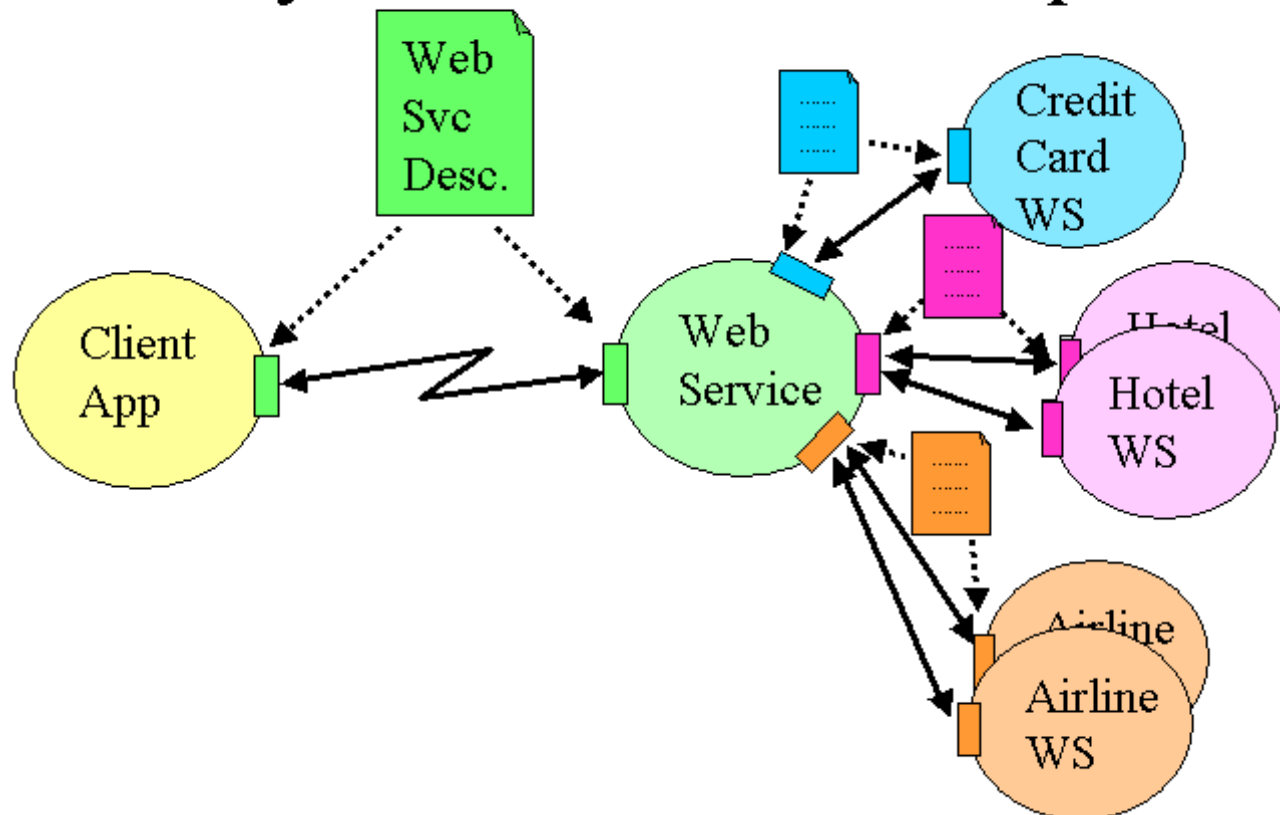


What is WSDL?

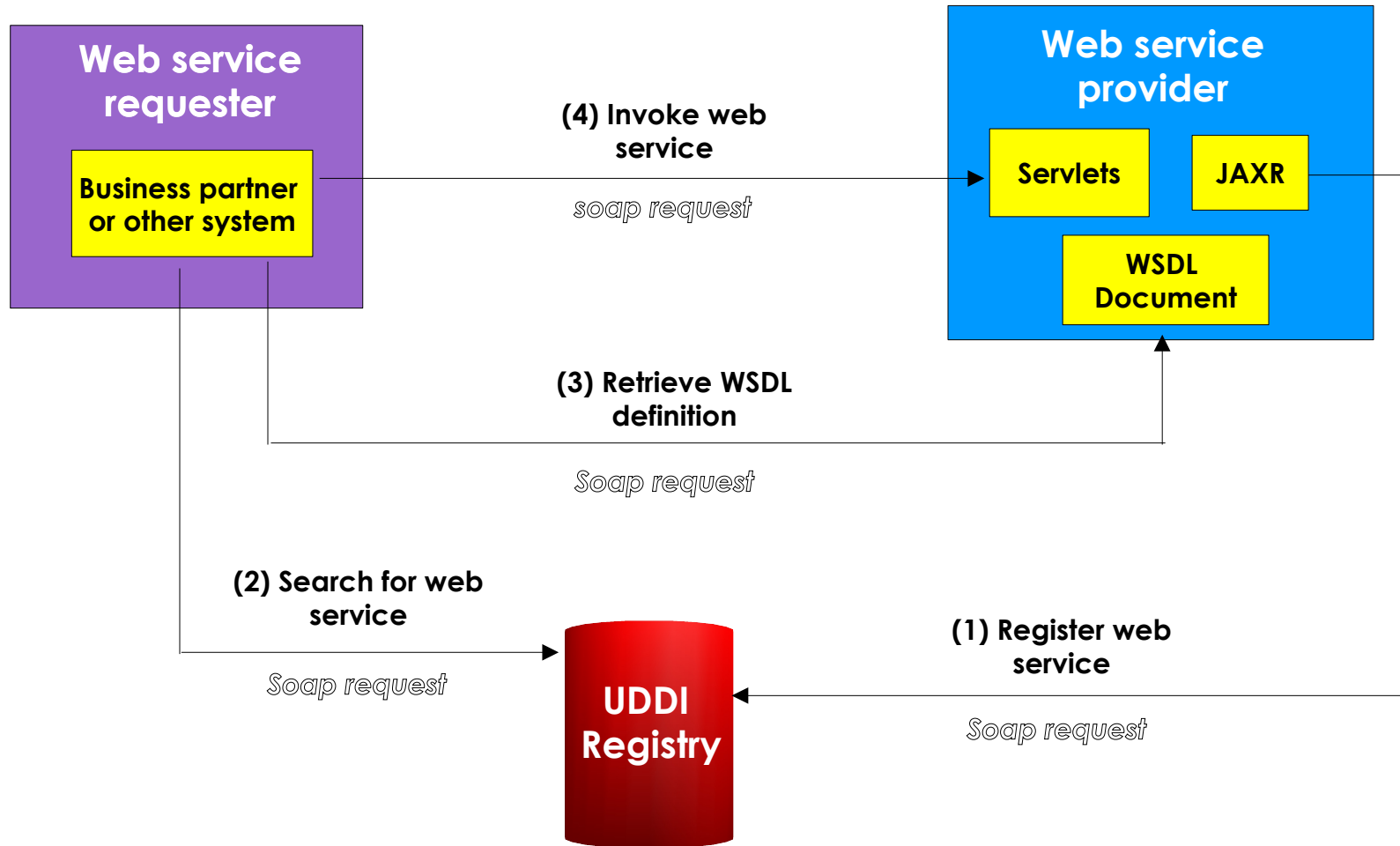
- XML language for **describing** web services
- Web service is described as
 - A set of **communication endpoints (ports)**
- Endpoint is made of two parts
 - **Abstract definitions of operations and messages**
 - **Concrete binding** to networking protocol (and corresponding endpoint address) and message encoding
- Why this separation?
 - Enhance **reusability** (of the abstract part, for example)

What is WSDL?

Many Web Service Descriptions



Where is WSDL Used?

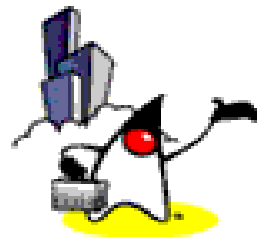


Why WSDL?

- Enables **automation** of communication details between communicating partners
 - Machines can read WSDL
 - Machines can invoke a service defined in WSDL
- Discoverable through registry
- Arbitration
 - 3rd party can verify if communication conforms to WSDL



WSDL Document Structure



WSDL Document Structure

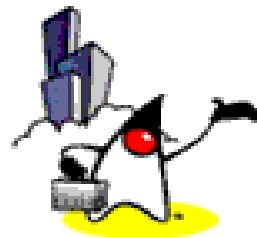
```
<wsdl:definitions xmlns:wsdl="http://schemas.xmlsoap.org/wsdl"
    targetNamespace="your namespace here"
    xmlns:tns="your namespace here"
    xmlns:soapbind="http://schemas.xmlsoap.org/wsdl/soap">
  <wsdl:types>
    <xs:schema targetNamespace="your namespace here (could be another) "
      xmlns:xsd="http://www.w3.org/2001/XMLSchema"
      <!-- Define types and possibly elements here -->
    </schema>
  </wsdl:types>
  <wsdl:message name="some operation input">
    <!-- part(s) here -->
  </wsdl:message>
  <wsdl:message name="some operation output">
    <!-- part(s) here -->
  </wsdl:message>
  <wsdl:portType name="your type name">
    <!-- define operations here in terms of their messages -->
  </wsdl:portType>
  <wsdl:binding name="your binding name" type="tns:port type name above">
    <!-- define style and transport in general and use per operation -->
  </wsdl:binding>
  <wsdl:service>
    <!-- define a port using the above binding and a URL -->
  </wsdl:service>
</wsdl:definitions>
```

WSDL Namespaces

- <http://schemas.xmlsoap.org/wSDL>
- <http://schemas.xmlsoap.org/wSDL/soap>
- <http://www.w3.org/2001/XMLSchema>



Example WSDL Document



WSDL Document Example

- Simple service providing stock quotes
- A single operation called **GetLastTradePrice**
- Deployed using **SOAP 1.1 over HTTP**
- Request takes a ticker symbol of type **string**
- Response returns price as a **float**

WSDL Elements

- Types
- Message
- Operation
- Port Type
- Binding
- Port
- Service

WSDL Elements

- Types
 - Data type definitions
 - Used to describe exchanged messages
 - Uses W3C XML Schema as canonical type system

WSDL Example: Types

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

WSDL Elements

- Messages
 - **Abstract**, typed definitions of **data** being exchanged
- Operations
 - **Abstract** description of an **action**
 - Refers to an **input** and/or **output messages**
- Port type
 - **Collection** of operations
 - **Abstract definition** of a service

Example:

Messages, Operation, Port type

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

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

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

WSDL Elements

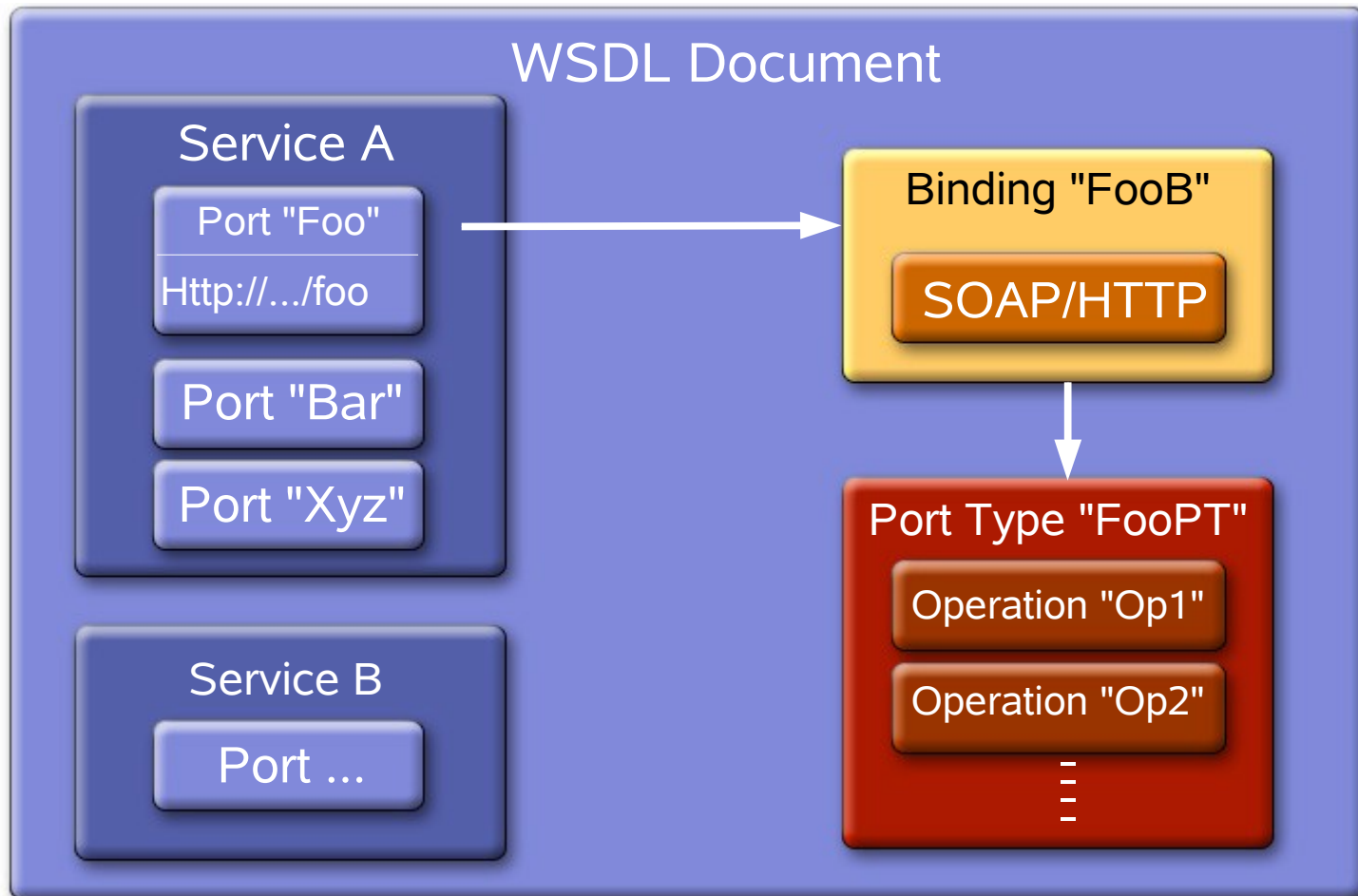
- Binding
 - **Concrete protocol and data format** (encoding) for a particular Port type
 - Protocol examples: SOAP 1.1 over HTTP or SOAP 1.1 over SMTP
 - Encoding examples: SOAP encoding, RDF encoding
- Port
 - Defines a single communication endpoint
 - **Endpoint address** for binding
 - URL for HTTP, email address for SMTP
- Service
 - **Aggregate set of related ports**

Example: Binding, Port, Service

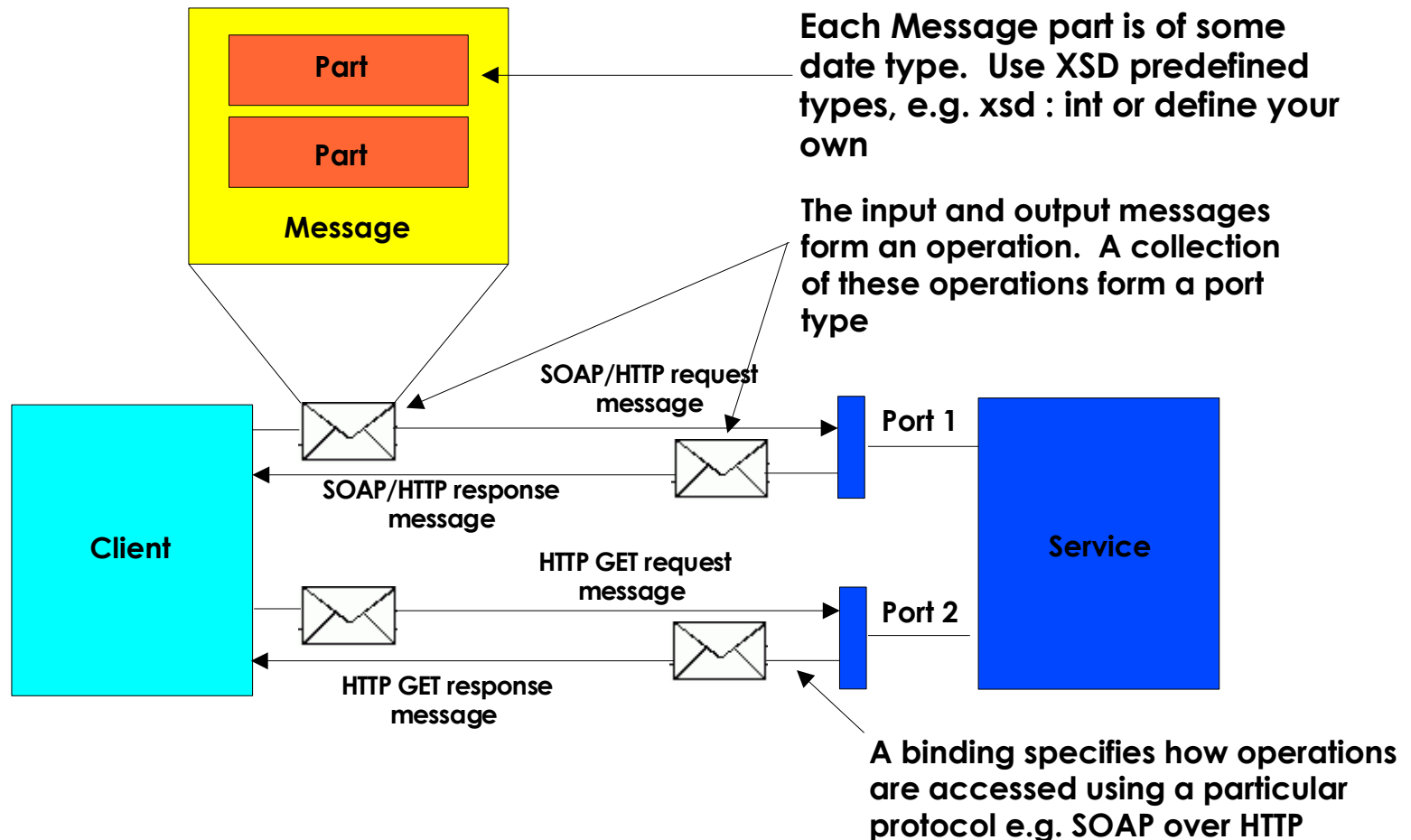
```
<binding name="StockQuoteSoapBinding" type="tns:StockQuotePortType">
  <soap:binding style="document"
    transport="http://schemas.xmlsoap.org/soap/http"/>
  <operation name="GetLastTradePrice">
    <soap:operation
      soapAction="http://example.com/GetLastTradePrice"/>
    <input> <soap:body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

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

WSDL View of a Web Service

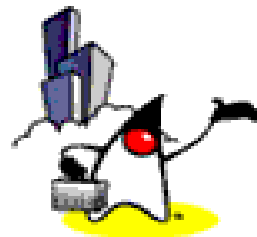


Web Service Invocation





More Detailed Description on WSDL Elements



Message Element

- Consist of one or more logical **parts**
- Syntax

```
<definitions .... >  
  <message name="nmtoken"> *  
    <part name="nmtoken" element="qname"? type="qname"?/> *  
  </message>  
</definitions>
```

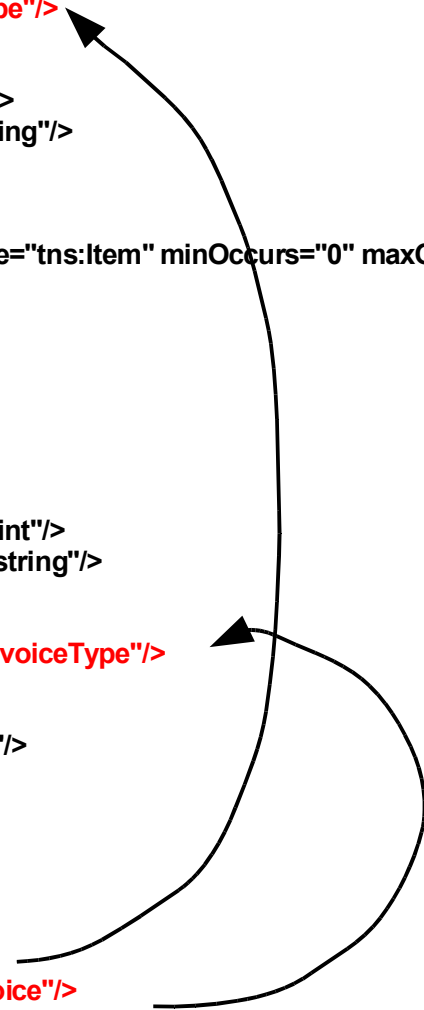
- *element* attribute refers to an XSD element using a QName
- *type* attribute refers to an XSD *simpleType* or *complexType* using a QName

Example: Message Element

```
<definitions .... >
  <types>
    <schema .... >
      <element name="PO" type="tns:POType"/>
      <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="0" maxOccurs="unbounded"/>
              </all>
            </complexType>
          </element>
        </all>
      </complexType>

      <complexType name="Item">
        <all>
          <element name="quantity" type="int"/>
          <element name="product" type="string"/>
        </all>
      </complexType>
      <element name="Invoice" type="tns:InvoiceType"/>
      <complexType name="InvoiceType">
        <all>
          <element name="id" type="string"/>
        </all>
      </complexType>
    </schema>
  </types>

  <message name="PO">
    <part name="po" element="tns:PO"/>
    <part name="invoice" element="tns:Invoice"/>
  </message>
</definitions>
```



The diagram consists of two curved arrows. The first arrow starts at the message definition `<part name="po" element="tns:PO"/>` and points to the `<element name="PO" type="tns:POType"/>` definition. The second arrow starts at the message definition `<part name="invoice" element="tns:Invoice"/>` and points to the `<element name="Invoice" type="tns:InvoiceType"/>` definition.

Types of Operations

- One-way
 - The endpoint receives a message
- Request/response
 - The endpoint receives a message, and sends a correlated message
- Notification
 - The endpoint sends a message
- Solicit/response
 - The endpoint sends a message, and receives a correlated message

One-way Operation

```
<operation name="submitPurchase">  
  <input message="purchase"/>  
</operation>
```

Request/Response Operation

```
<operation name="submitPurchase">  
  <input message="purchase"/>  
  <output message="confirmation"/>  
</operation>
```

```
<operation name="submitPurchase">  
  <input message="purchase"/>  
  <output message="confirmation"/>  
  <fault message="faultMessage"/>  
</operation>
```

Notification Operation

```
<operation name="deliveryStatus">  
  <output message="trackingInformation"/>  
</operation>
```

Solicit/Response Operation

```
<operation name="clientQuery">  
  <output message="bandwidthRequest"/>  
  <input message="bandwidthInfo"/>  
  <fault message="faultMessage"/>  
</operation>
```



Importing & Authoring Style

Authoring Style Recommendation

- Reusability and maintainability
- Maintain WSDL document in 3 separate parts
 - Data type definitions
 - Abstract definitions
 - Specific service bindings
- Use “import” element to import necessary part of WSDL document

Example7A:

<http://example.com/stockquote/stockquote.xsd>

```
<?xml version="1.0"?>
<schema targetNamespace="http://example.com/stockquote/schemas"
  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>
```

Example 7B:

<http://example.com/stockquote/stockquote.wsdl>

```
<?xml version="1.0"?>
<definitions name="StockQuote"

targetNamespace="http://example.com/stockquote/definitions"
  xmlns:tns="http://example.com/stockquote/definitions"
  xmlns:xsd1="http://example.com/stockquote/schemas"
  xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/"
  xmlns="http://schemas.xmlsoap.org/wsdl/">

  <import namespace="http://example.com/stockquote/schemas"
    location="http://example.com/stockquote/stockquote.xsd"/>

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

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

Example7C:

<http://example.com/stockquote/stockquoteservice.wsdl>

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

  <import namespace="http://example.com/stockquote/definitions"
    location="http://example.com/stockquote/stockquote.wsdl"/>

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



WSDL 1.2

WSDL 1.2

- Described in XML Information set
- Aligned with SOAP 1.2
 - Changes are mostly about SOAP binding
- MEP (Message Exchange Pattern)
- Clarifications
 - Solicit-response and notification model
- Still “work in progress” state