

Web Service Definition Language



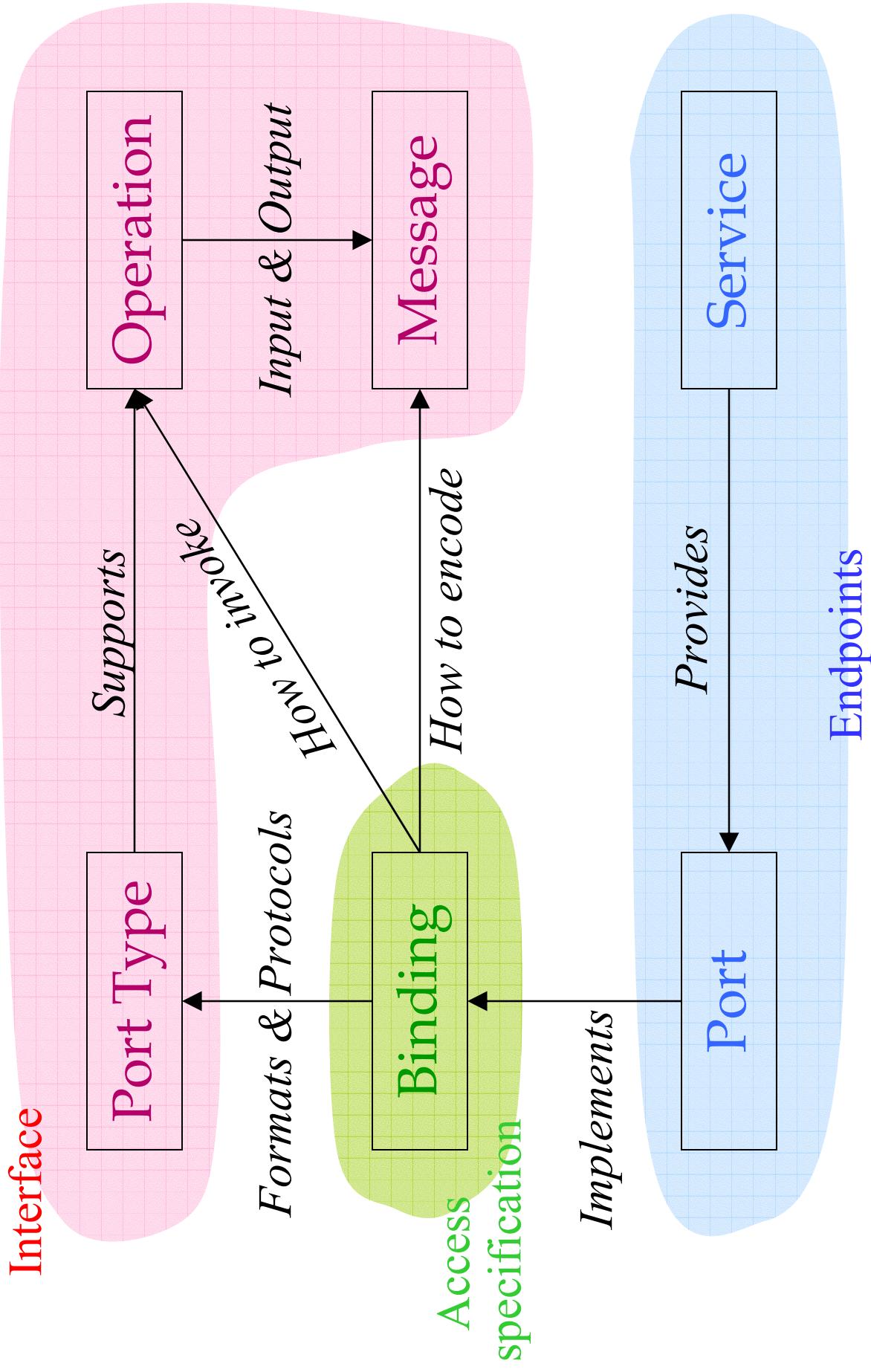
Web Service Definition Language (WSDL)

■ What is a web service?

[F. Leymann 2003]

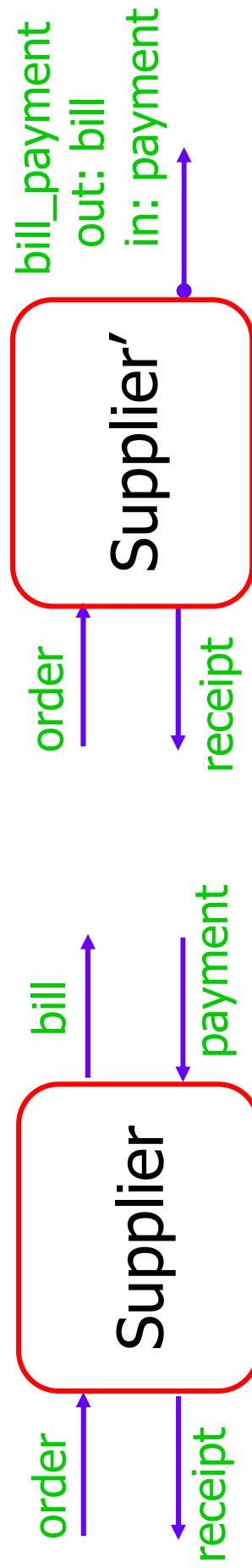
- A piece of code that can communicate with other pieces of code via the Internet
- Can hide “middleware ideoynchronies” (underlying component model, invocation protocol etc.) as far as possible
- WSDL defines
 - What the service is – **interface**
 - Access specification – **how**
 - Location of the service – **where**

Ingredients of WSDL



Web Services Definition Language (WSDL)

- Traditional I/O signatures
- Peer-to-peer: web service can act as client or server
 - Proactive : send request
 - send request, block till response
 - Reactive : receive request
 - receive request, send response
 - Reactive :
 - receive request, send response
- Port: mechanism to cluster operations
 - Port as unit of interoperation between services



Main Structure of WSDL

```
<definitions namespace = “http://...”>  
<types> XML schema types </type>  
<message> definition of a message </message>  
<portType> a set of operations </portType>  
<binding> communication protocols </binding>  
<service> a list of binding and ports </service>  
</definitions>
```

Types

- <types> define data types used in defining messages
- XML Schema, DTD, and etc.
- XML Schema must be supported by any vendor of WSDL conformant products

Type Examples

```
<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"
            minOccur = "1" maxOccur="10"/>
          <element name = "payment">
            <complexType>
              <choice>
                <element name = "account" type="string"/>
                <element name = "creditcard" type="string"/>
              </choice>
            </complexType>
          </element>
        </all>
      </complexType>
    </element>
  </schema>
</types>
```

WSDL Messages

- A <message> element defines the data elements of an operation
 - Each message can be the input or output of an operation, and may consist of one or more parts
 - A part resembles a parameter of a function

Message Examples

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

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

WSDL Ports

- The `<portType>` element is the most important WSDL element: it defines
 - a web service
 - the operations that can be performed, and
 - the messages that are involved
- The `<port>` defines the connection point to a web service, an instance of `<portType>`
 - It can be compared to a function library (or a module, or a class) in a traditional programming language
 - Each operation can be compared to a function in a traditional programming language

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

Operation Types

- The request-response type is the most common operation type, but WSDL defines four types:
 - **One-way:** The operation can receive a message but will not return a response
 - **Request-response:** The operation can receive a request and will return a response
 - **Solicit-response:** The operation can send a request and will wait for a response
 - **Notification:** The operation can send a message but will not wait for a response
- WSDL 1.2 adds: request – multiple response

One-Way Operation

```
<message name="newTermValues">
  <part name="term" type="xs:string"/>
  <part name="value" type="xs:string"/>
</message>

<portType name="glossaryTerms">
  <operation name="setTerm">
    <input name="newTerm"
      message="newTermValues" />
  </operation>
</portType>
```

Request-Response Operation

```
<message name="getTermRequest">
  <part name="term" type="xs:string"/>
</message>

<message name="getTermResponse">
  <part name="value" type="xs:string"/>
</message>

<portType name="glossaryTerms">
  <operation name="getTerm">
    <input message="getTermRequest"/>
    <output message="getTermResponse"/>
  </operation>
</portType>
```

One-way and Notification Example

```
<portType name="RegisterPort">
  <operation name="register">
    <input name="customerInfo" message="RegInfo"/>
  </operation>
  <operation name="register Response">
    <output name="response" message="ResponseInfo"/>
  </operation>
</portType>
```

Binding

- Binding defines how message are transmitted, and the location of the service
- <binding> element has two attributes:
 - type: the port type
 - name: name of the binding
- <soap:binding> has two attributes:
 - style: either "document" or "rpc"
 - transport: protocol to use, e.g., "http"

Binding Example

```
<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>
```

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

Services

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

Summary

- WSDL provides a practical framework for defining
 - Services: functions and input/output
 - Bindings using SOAP (document style or PRC style)
 - Service entry points
- Many WSDL services exist (www.xmethods.com), e.g.,
 - <http://soap.achhex.com/exec/btrnsoap.dll/wSDL/IBTRNSOAP>