

IT2605

Applications of Web Services

L02 Web Service Technology

Learning Outcome

- ▶ What are Web Services?
- ▶ Why Web Services?
- ▶ Web Services Technologies
 - Extensible Mark-Up Language (XML)
 - Web Service Description Language (WSDL)
 - Simple Object Access Protocol (SOAP)
 - Universal Description, Discovery, and Integration (UDDI)
- ▶ Components of WSDL



The Scenario

- ▶ Within an organisation, different departments often need data from other departments:
 - Finance needs employee information from HR for payroll purposes.
 - Sales need to know product stock levels from warehousing.
 - HQ HR may need employee information from regional HRs.



The Scenario

- ▶ Organisations also often communicate with other organisations:
 - A car manufacturer may want to order tires from their tire supplier.
- ▶ Many of these functions are increasingly done via IT systems over networks.



The Problem

- ▶ Different organisations implement their IT systems on different platforms.
 - Different programming frameworks, APIs & languages:
 - Java, C++, Visual C#, Visual Basic, PHP, ASP.NET, etc.
 - Different databases:
 - Oracle, MS SQL Server, MySQL Server, etc.
 - Different Operating Systems:
 - Microsoft Windows, Unix, etc.



The Problem

- ▶ Even within large organisations, there may be differences:
 - Different departments may use different IT systems for their purposes:
 - HR, Finance, Marketing, Sales, Warehousing, Shipping
 - Same departments in different International regions may use different IT systems for the same functions.



The Problem

- ▶ Difficult for the different systems to inter-operate because they:
 - Represent data differently.
 - Communicate in different "languages".
 - Communicate over different channels.
- ▶ A lot of effort are required to create CUSTOM "middle-man" software to allow systems to communicate.



The Problem

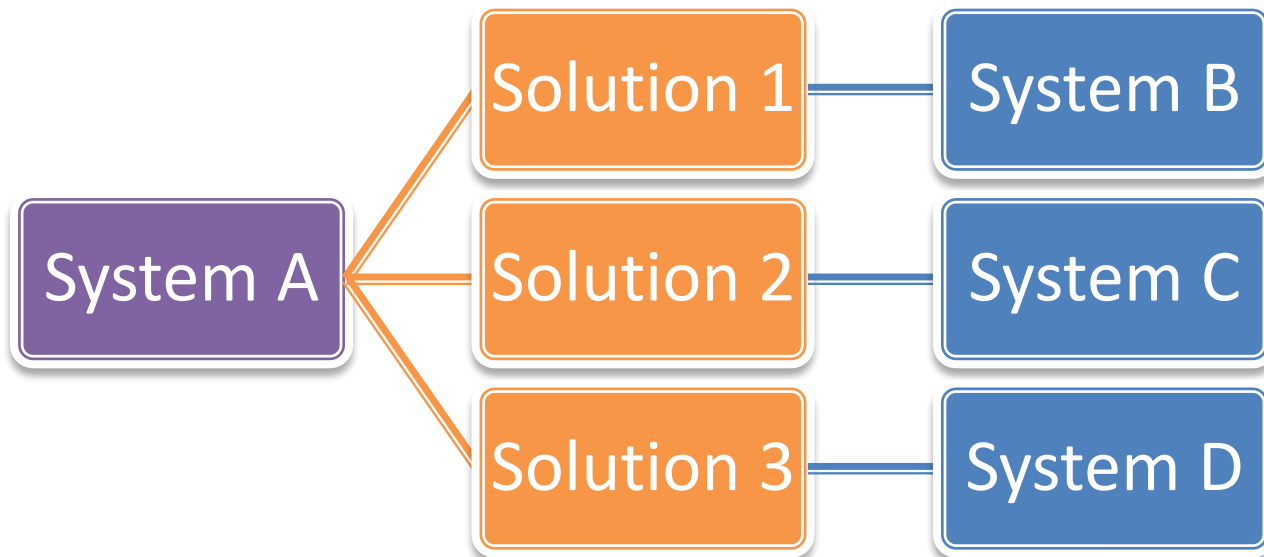
- ▶ Difficult to develop applications that could:
 - Access the systems from outside the company Firewalls:
 - Salesman using his laptop to get instant information while at the customer's office.
 - Run from a different platform or OS, like Smart Phones.



The Problem

▶ Other problems:

- Upgrading systems to better but different systems, will require changes to the "middle-man" software.
- For each new/different system encountered, a customise solution may be needed.



The Solution

- ▶ All systems use **ONE** standardised language and **ONE** standardised protocol to communicate. (Instead of multiple custom-built solutions.)
- ▶ The solution is: **WEB SERVICES**



What are Web Services?

The World Wide Web Consortium (W3C) defines a Web Service as:

- ▶ A software system designed to support interoperable machine-to-machine interaction over a network.

Web Services achieve this by using:

- ▶ A standard language to represent messages (requests and data): **eXtensible Markup Language (XML)**
- ▶ A standard transport protocol for transmitting messages: **Hypertext Transport Protocol (HTTP)**



Web Service Technologies

- ▶ XML
- ▶ Simple Object Access Protocol (SOAP)
- ▶ Web Services Description Language (WSDL)
- ▶ Universal Description, Discovery and Integration (UDDI)



Web Service Technologies

SOAP

- ▶ An XML-based **communication protocol** to let applications communicate over HTTP

WSDL

- ▶ An XML-based language for locating and describing Web services. It describes:
 - Location: URL of the web service
 - Operations available: What the web service can do
 - Types: Types of the data for input and output. E.g. simple types like strings and doubles, to complex types like lists and objects.



Web Service Technologies

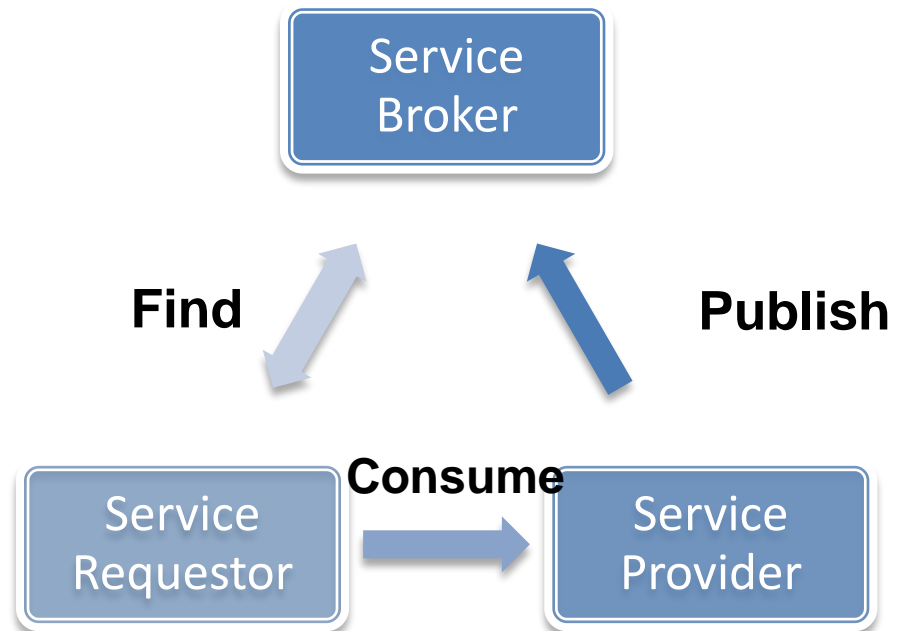
UDDI

- ▶ A directory service where organisation can register and search for Web services.



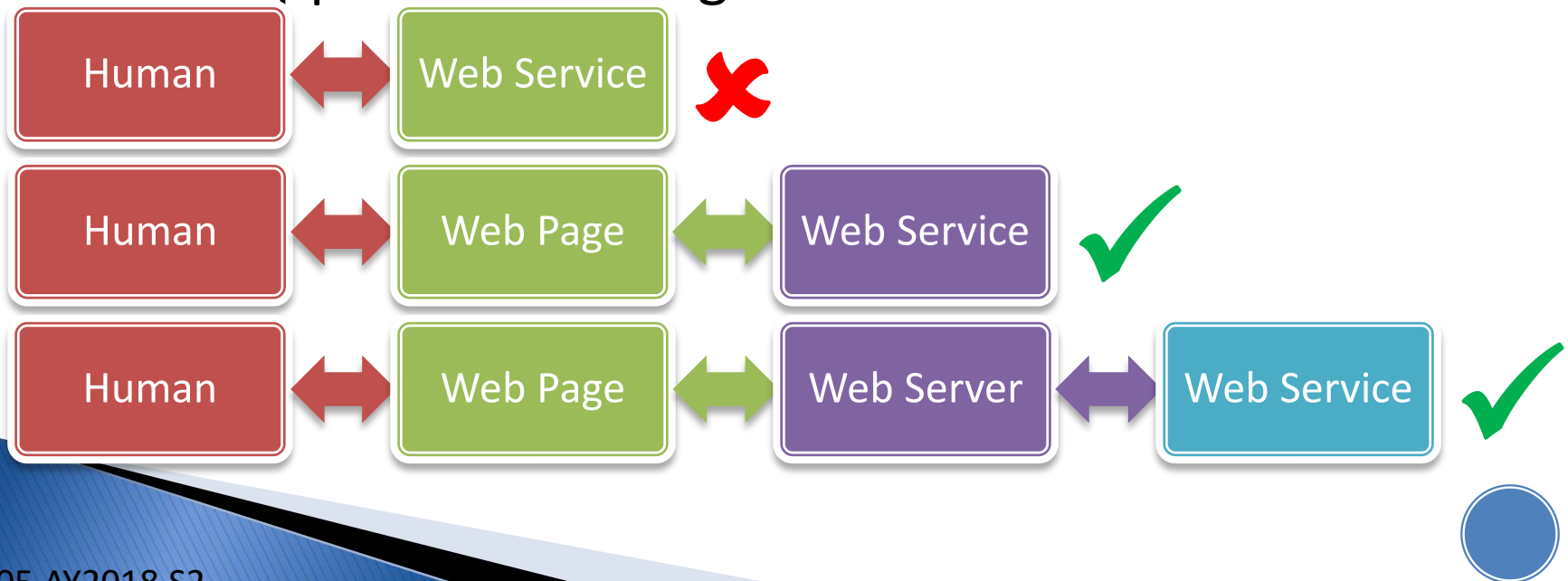
Web Services Components

- ▶ **Service Provider**
 - Provides web services
 - Publishes its web services to a Service Broker
- ▶ **Service Broker**
 - Maintains a registry of web services published to it.
- ▶ **Service Requestor**
 - Queries Service Brokers to find required web services.
 - Binds to the web service provided by the service provider.



What Web Services are NOT!

- ▶ Web Services are NOT web sites/pages.
- ▶ Web Services are NOT meant for human end-users: NO GUI.
- ▶ Web Services are for applications to communicate with other applications: Programmatic Interface.



Web Service Description Language (WSDL)

WSDL Overview

- ▶ Web Services Description Language
 - WSDL definitions describe how to access a web service and what operations it will perform.
 - It is based on XML
 - platform- and language-independent
 - Analogy: If a car is a Web Service, WSDL is the instructions for how to drive the car

<http://www.w3schools.com/wsdl/default.asp>



WSDL Syntax

- ▶ A WSDL document provides 4 important information about a web service

Interface Information

Describes publicly available web methods

Data Type Information

For all message requests and responses (String, number, complex types like DataSet)

Binding Information

Indicates transport protocol to be used

Address Information

Location of specified service (URL)

Currently WSDL is in its 2nd version i.e.. WSDL 2.0



Components of a WSDL Document

There are 8 major elements in a WSDL Document

Major Elements	Description
d efinitions	<ul style="list-style-type: none">▪ Root element in the WSDL document.▪ Declares multiple namespaces used by the web service
t ypes	<ul style="list-style-type: none">▪ Describes all data types used by the web services.
m essage	<ul style="list-style-type: none">▪ Define name and data element of the message.
o perations	<ul style="list-style-type: none">▪ Describe an action supported by the service.

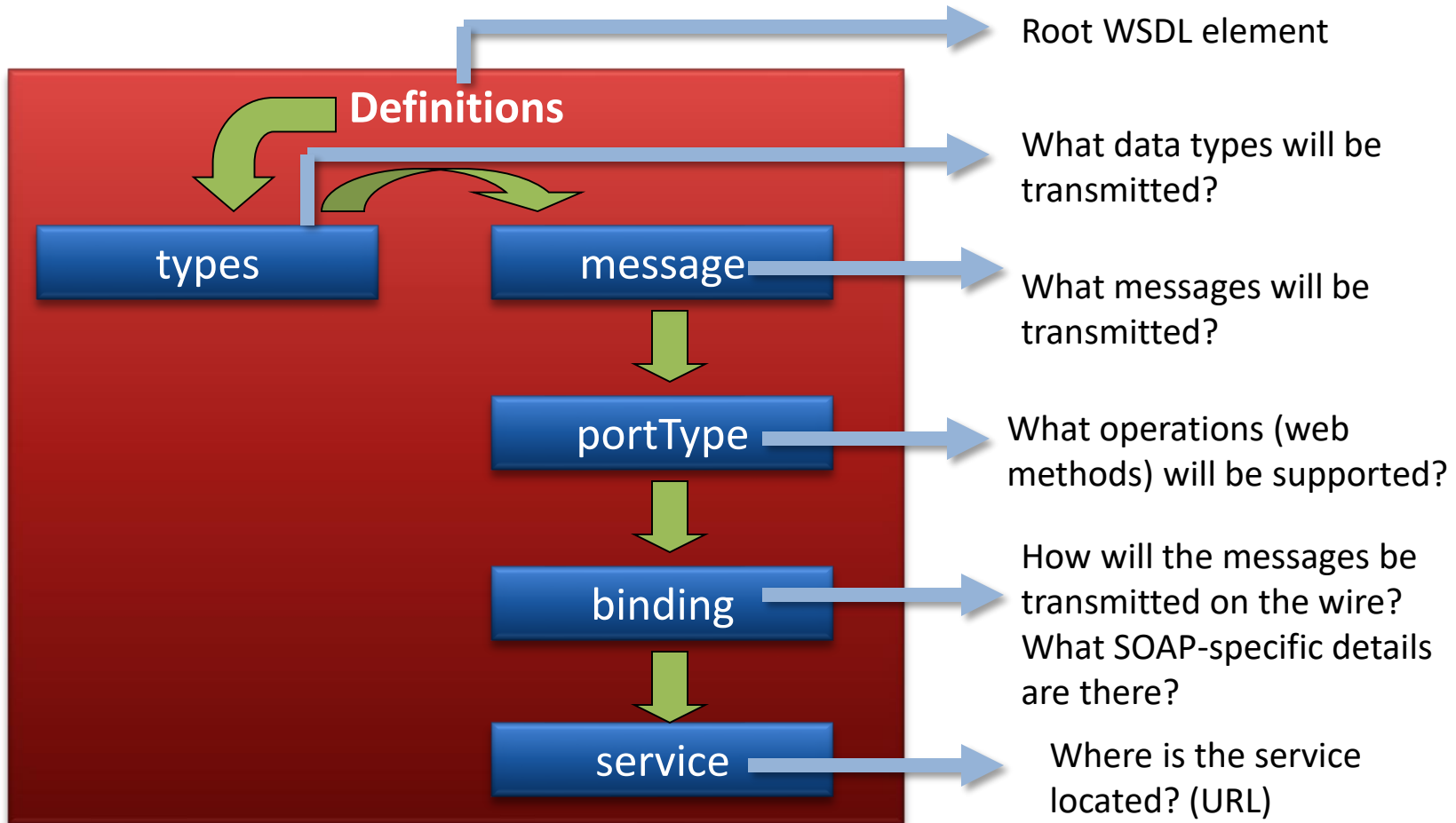


Components of a WSDL Document

Major Elements	Description
portType	<ul style="list-style-type: none">▪ Describe a set of operations supported in the web service▪ Specifies the messages used by each operation
binding	<ul style="list-style-type: none">▪ Defines the message format and protocol details for each port.
port	<ul style="list-style-type: none">▪ Specifies the url address of the binding
service	<ul style="list-style-type: none">▪ Mainly contains the address for invoking the web services.



WSDL Syntax



WSDL Syntax



definition Element

- ▶ We will now analyze the WSDL elements in detail:

```
<definitions xmlns:http="http://schemas.xmlsoap.org/wsdl/http/"
xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/"
xmlns:s="http://www.w3.org/2001/XMLSchema" xmlns:s0="http://tempuri.org/"
xmlns:soapenc="http://schemas.xmlsoap.org/soap/encoding/"
xmlns:tm="http://microsoft.com/wsdl/mime/textMatching/"
xmlns:mime="http://schemas.xmlsoap.org/wsdl/mime/"
targetNamespace="http://tempuri.org/"
xmlns="http://schemas.xmlsoap.org/wsdl/">
...types...
...messages...
...portType...
...binding...
...service...
</definitions>
```



definition Element

- ▶ The `definition` element is the **root** element of the WSDL document
- ▶ Serves as **parent** for the `types`, `message`, `portType`, `service` elements
- ▶ The `targetNamespace` attribute allows us to specify a namespace that the definition will belong to



types Element

- ▶ The `types` element allows us to define the data type that web service will use
- ▶ In the sample `types` element shown on the next slide, the element called `DoubleMyPay` is defined and has a data type with the following characteristics:
 - It can only contain at most one single element called `mypay`.
 - The value of the `mypay` element is of the type `double`.
- ▶ Note that the data type definition makes use of the XML schema



types Element

- ▶ The following shows a sample of the types element:

```
<wsdl:types>
  <s:schema elementFormDefault="qualified" targetNamespace="http://tempuri.org/">
    <s:element name="DoubleMyPay">
      <s:complexType>
        <s:sequence>
          <s:element minOccurs="1" maxOccurs="1" name="mypay" type="s:double"/>
        </s:sequence>
      </s:complexType>
    </s:element>
    <s:element name="DoubleMyPayResponse">
      <s:complexType>
        <s:sequence>
          <s:element minOccurs="1" maxOccurs="1" name="DoubleMyPayResult"
            type="s:double"/>
        </s:sequence>
      </s:complexType>
    </s:element>
  </s:schema>
</wsdl:types>
```

The data type of input message

The data type of output message



message Element

- ▶ The second part of the WSDL document contains the message elements
- ▶ message elements specify the messages that will be sent between the client and the web services
- ▶ Multiple message elements are commonly present in a WSDL document
- ▶ Each message element contains one or more part elements that describes a piece of content within the message. For example,
 - A parameter in a request string or
 - the body of a SOAP message



message Element

- Below are samples of the message elements:

A parameter in the request.

The element that describe the data type.

```
<wsdl:message name="DoubleMyPaySoapIn">
  <wsdl:part name="parameters" element="tns:DoubleMyPay"/>
</wsdl:message>
<wsdl:message name="DoubleMyPaySoapOut">
  <wsdl:part name="parameters" element="tns:DoubleMyPayResponse"/>
</wsdl:message>
```



portType Element

- ▶ The third part of the WSDL document is the `portType` element
- ▶ The `portType` element specifies **operations**
- ▶ Operations are the communications that can occur between the client and the server
 - You can think of `portType` as an interface while operations are the methods of the interface
- ▶ `operation` elements define **actions** to be performed



portType Element

There are 4 types of operations

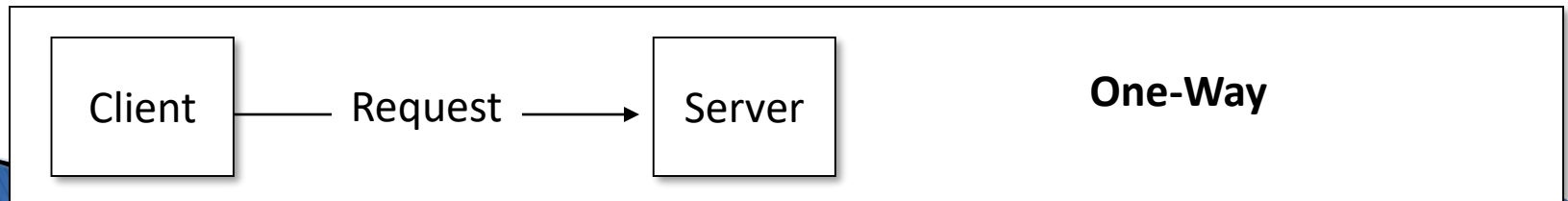
▶ Request-Response

- RPC style. Sending a request and receiving a response.



▶ One-Way

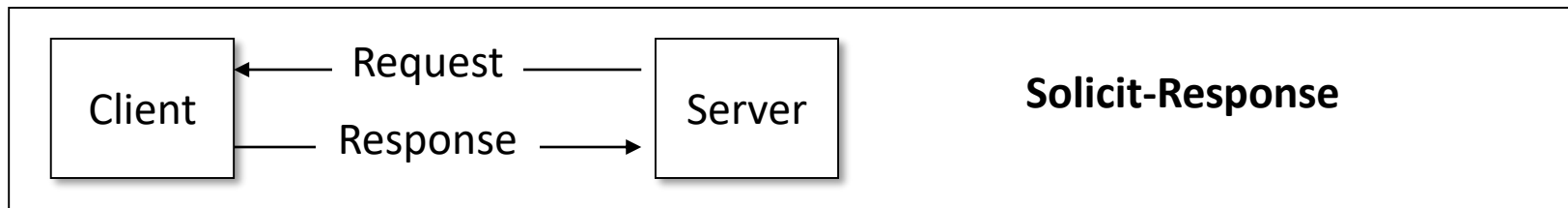
- Document style. Send information. May not receive acknowledgement.



portType Element

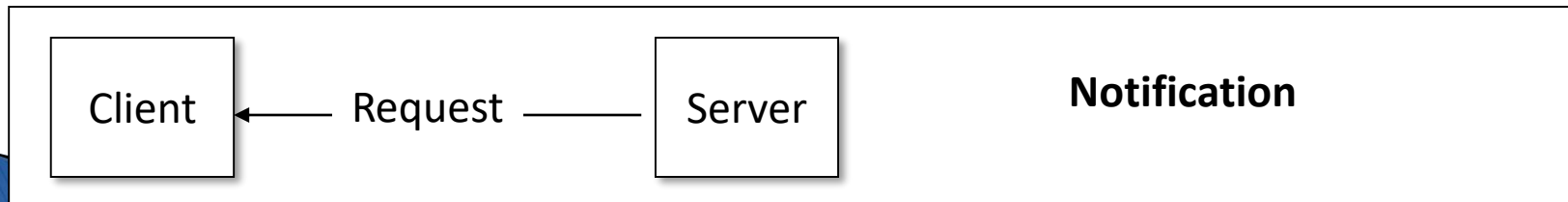
► Solicit-response

- Similar to Request-Response. Except that it is the server that sends the request while the client responds.



► Notification

- Similar to One-Way. Except that the information is sent from the server to the client



portType Element

- ▶ An operation is composed of a subset of `input`, `output`, `fault` elements
- ▶ The order of the `input`, `output` and `fault` elements determines the type of operation

Operation Type	input	output	fault
Request-response	1	2	3*
One-Way	1		
Solicit-response	2	1	3*
Notification		1	



portType Element

- ▶ Shown below is a sample of a `portType` element within the WSDL document

Calling the DoubleMyPay function or method.

The input message

```
<wsdl:portType name="PayrollServiceSoap">
  <wsdl:operation name="DoubleMyPay">
    <wsdl:input message="tns:DoubleMyPaySoapIn"/>
    <wsdl:output message="tns:DoubleMyPaySoapOut"/>
  </wsdl:operation>
</wsdl:portType>
```

This is a Request-Response operation

The output message



binding Element

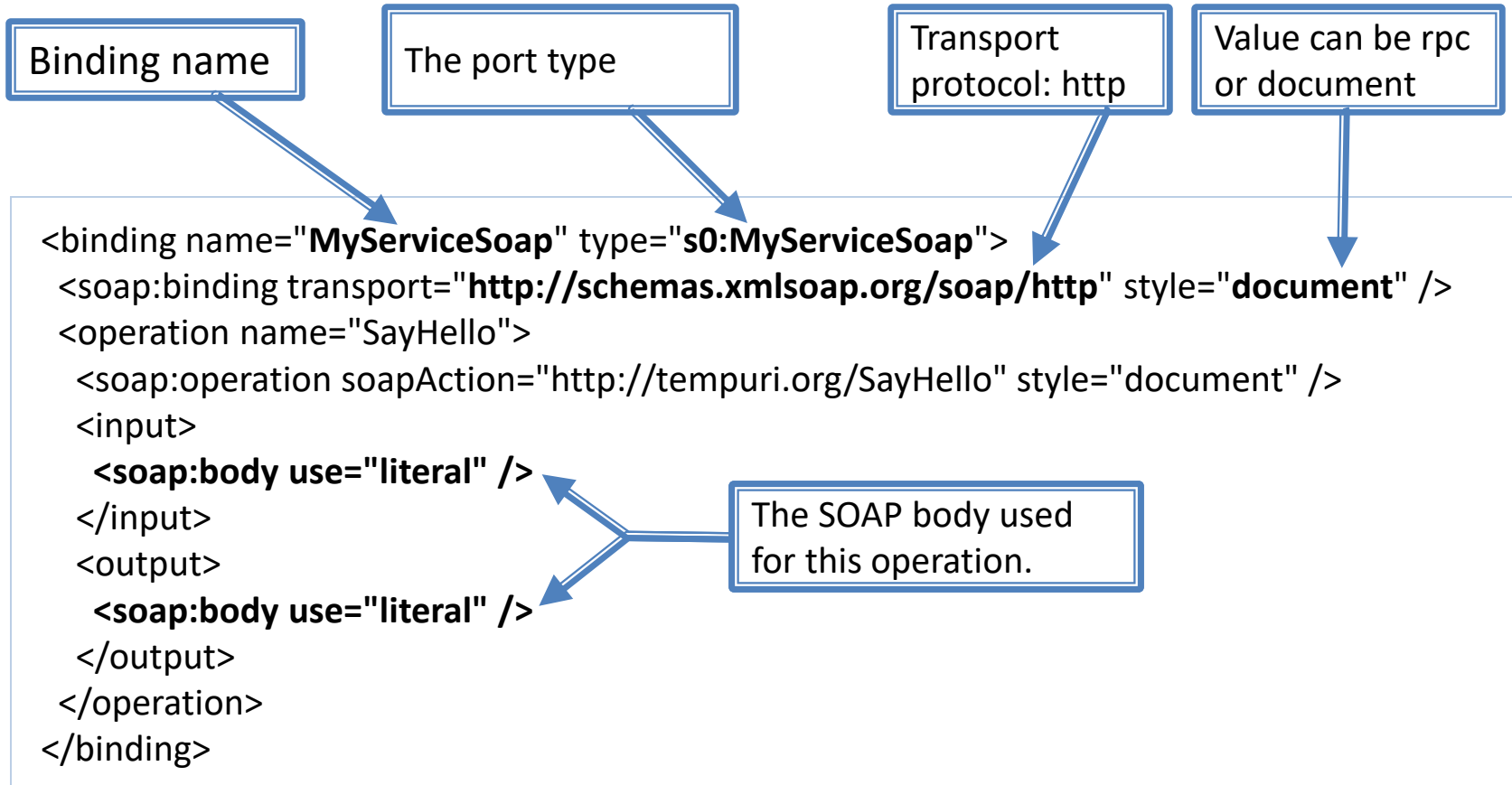
- ▶ The `binding` element
 - specifies how messages are to be transmitted.
 - contains *binding* definition.
- ▶ Binding definitions specify message formatting and protocol (SOAP, HTTP etc) details
- ▶ The following shows the general structure of a `binding` element.

```
<binding name="MyBindingName" type="MyPortType">  
  <operation name="Op1">  
    <input >...protocol and formatting Information... </input>  
    <output >... protocol and formatting Information ... </output>  
    ...Other Input/Output  
  </operation>  
  ...Other Operations...  
</portType>
```



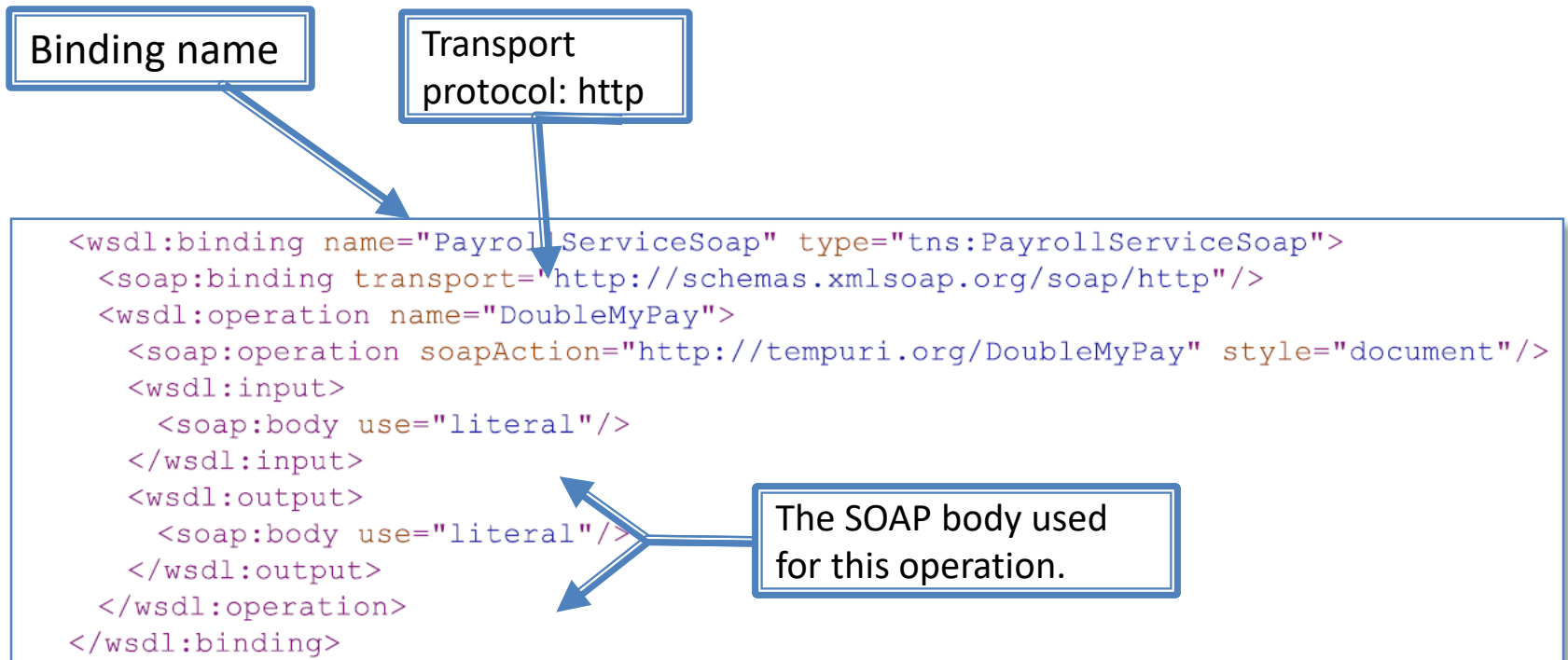
binding Element

- Below is a the binding part of the WSDL document:



binding Element

- ▶ Below is a the binding part of the WSDL document:



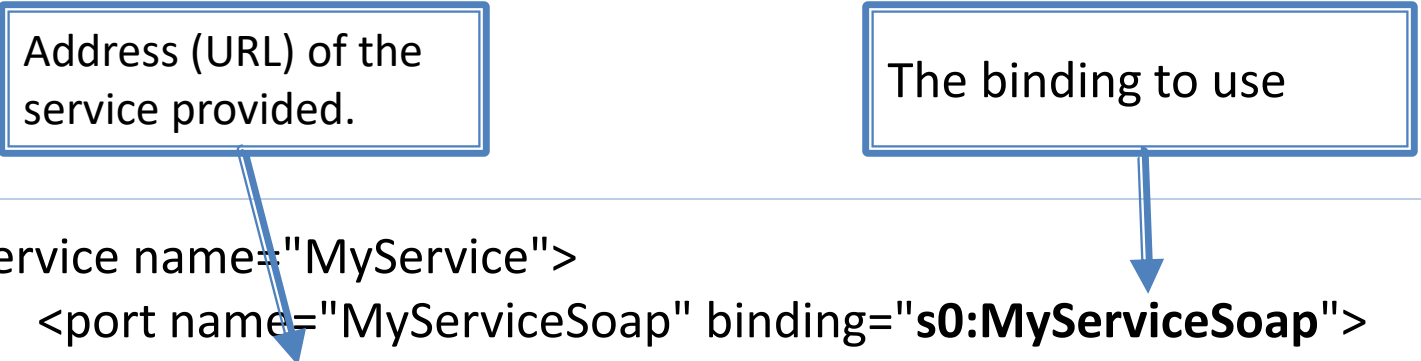
service Element

- ▶ The `service` element contains a group of related `port` elements
- ▶ Each `port` element contains URL of the service and which binding to be used.

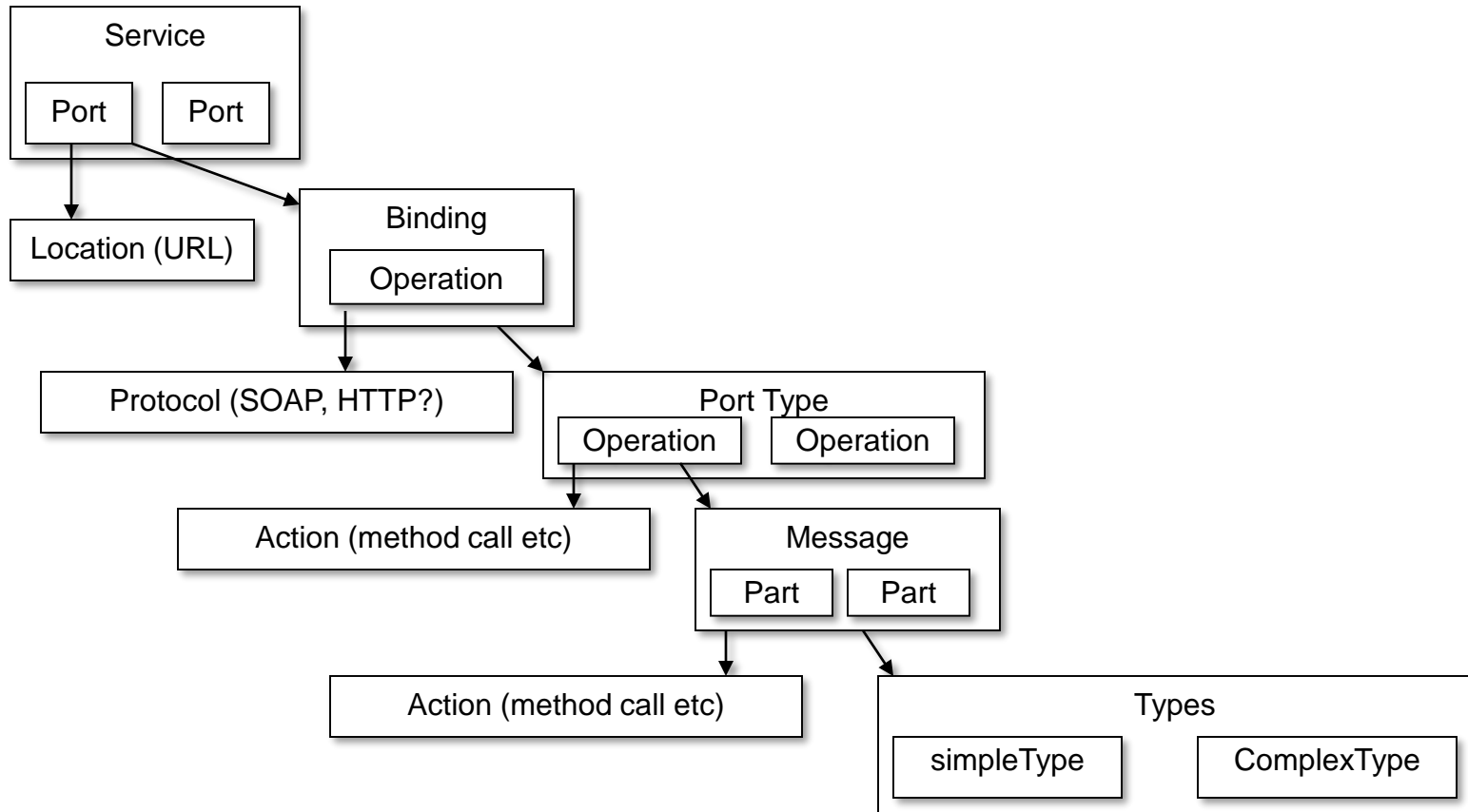
Address (URL) of the service provided.

The binding to use

```
<service name="MyService">  
  <port name="MyServiceSoap" binding="s0:MyServiceSoap">  
    <soap:address location="http://localhost/Service.asmx" />  
  </port>  
</service>
```



WSDL Syntax



Summary

- ▶ Web Service: A software system designed to support interoperable machine-to-machine interaction over a network.
- ▶ Web Services use XML as a language to represent messages.
- ▶ Web Services transmit messages over HTTP.
- ▶ Web Services consist of:
 - SOAP: A XML based communication protocol
 - WSDL: A XML based language for describing web services
 - UDDI: A directory of web services
- ▶ Components of WSDL



PRACTICAL TIME!