German Jordanian University الجامعة الألمانية الأردنية

CS355 Web Technologies

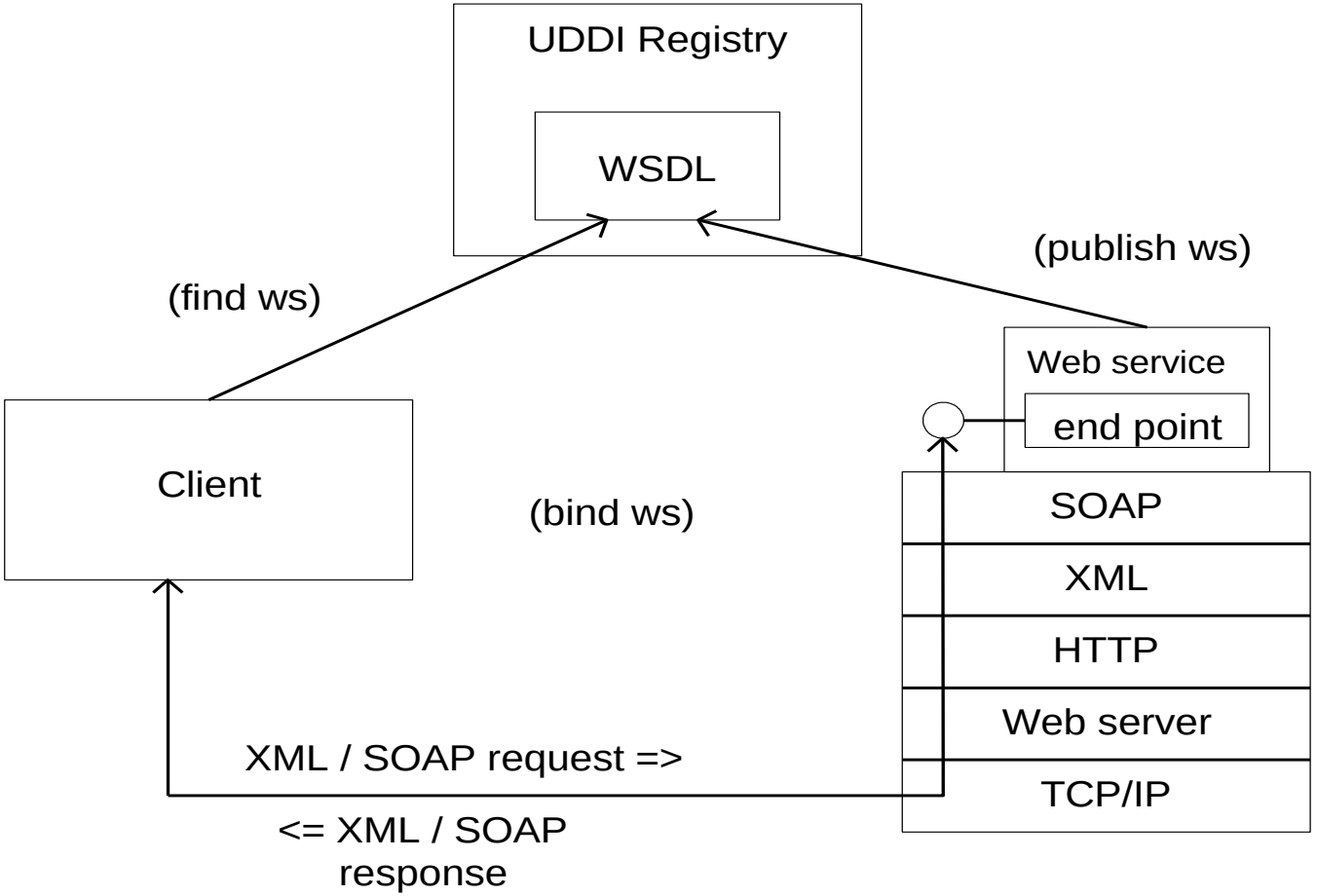
Dr. Ismail Hababeh

German-Jordanian University

Lecture 25

The Web Service Architecture

- Web service provider
- Web service requester (client)
- Web service registry. (Universal Description Discovery Integration)



The Web Service Architecture

- A Web service provider must publish/register a web service with a Universal Description Discovery Integration (UDDI) registry so that it can be accessed by any Web service requester globally.

The Web Service Architecture

- A web service requester (client) can lookup a specific Web service with help of UDDI by its provider (business) name, category info, Web service name, or even by the keys.
- The Web service - Apache eXtensible Interaction System (AXIS) run time engine is used to compile the web service automatically when this web service invoked by the client.

The Web Service Architecture

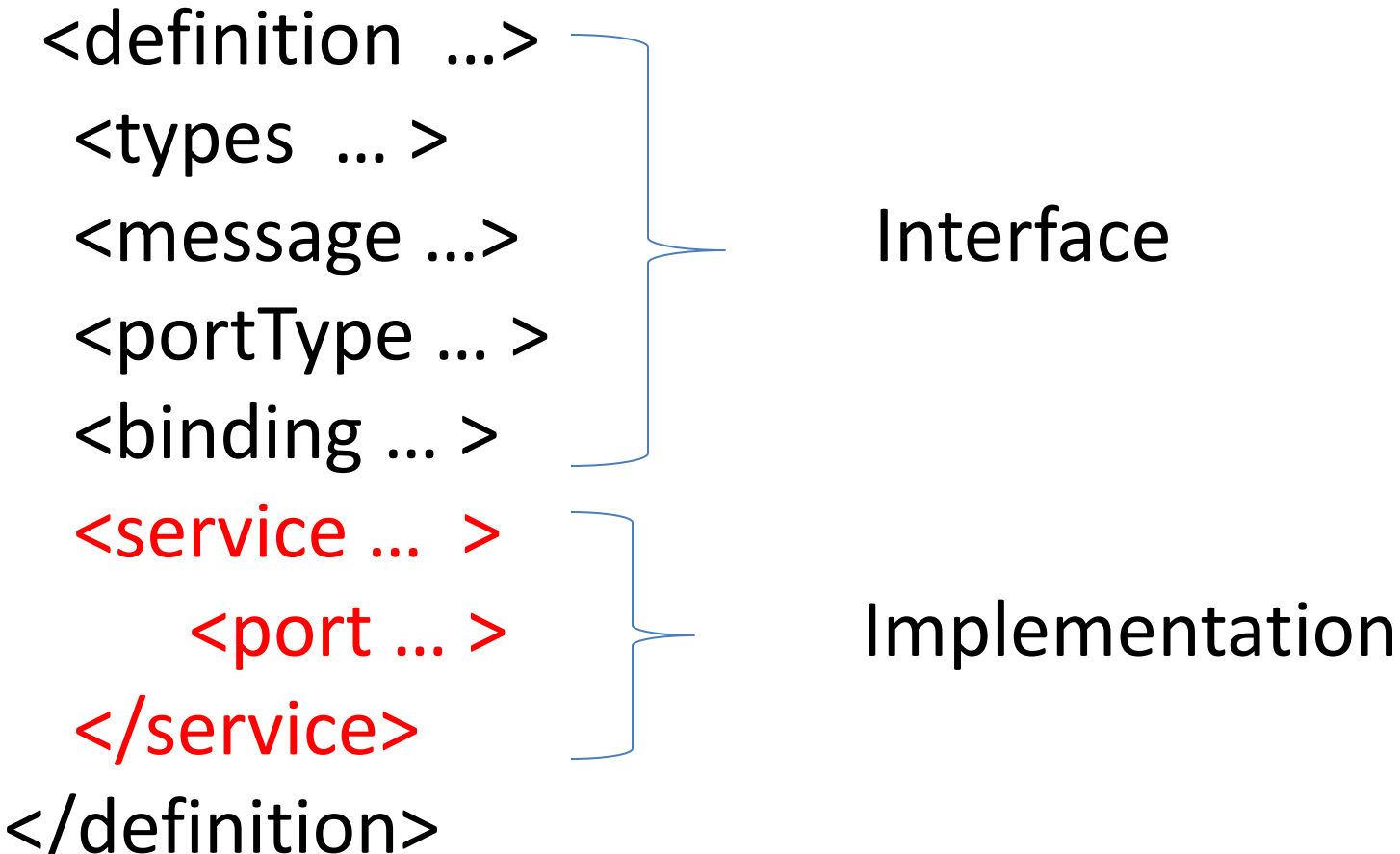
- A Web service can also be reached without any assistance from UDDI if the client knows the contact information such as:
  - Web service's URL
  - Method signature (method name, arguments, and return value).

The Web Service Architecture

- A Web service provider registers a Web Service Description Language (WSDL) interface at UDDI registry which is a contract interface of the web service to be used by its client.
- Both UDDI query requests and responses are in the SOAP formats.

WSDL Structure

The structure format of a WSDL as follows:



WSDL - Example

- WSDL definition part of Convert.wsdl element and sub-elements:

```
<wsdl:definitions . . . >
  <wsdl:message name="toFahrenheitResponse">
    <wsdl:port name="return" type="SOAP-ENC:string"/>
  </wsdl:message>
  <wsdl:message name="toFahrenheitRequest">
    <wsdl:port name="in0" type="SOAP-ENC:string"/>
  </wsdl:message>
```

```
<wsdl:portType name="Convert">
<wsdl:operation name="toFahrenheit" parameterOrder="in0">
      <wsdl:input message="intf:toFahrenheitRequest"/>
      <wsdl:output message="intf:toFahrenheitResponse"/>
   </wsdl:operation>          // intf: means interface
  </wsdl:portType>
  <wsdl:binding name="ConvertSoapBinding" type="intf:Convert">
   <wsdlsoap:binding style="rpc"  //SOAP binding can be Remote Procedure Call (RPC) style binding or a document style
        transport="http://schemas.xmlsoap.org/soap/http"/>
   <wsdl:operation name="toFahrenheit">
     <wsdlsoap:operation soapAction=""/>
```

```
<wsdl:input>
<wsdlsoap:body
    encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
      namespace="urn:myDirectory" use="encoded"/>
 </wsdl:input>
/* Uniform Resource Names (URNs) are resource identifiers with the specific requirements for
    enabling location independent identification of a resource.*/
 <wsdl:output>
 <wsdlsoap:body
    encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
        namespace="urn:myDirectory" use="encoded"/>
 </wsdl:output>
</wsdl:operation>
</wsdl:binding>
```

```
<wsdl:service name="ConvertService">
    <wsdl:port binding="intf:ConvertSoapBinding" name="Convert">
      <wsdlsoap:address
        location="http://localhost:8080/axis/services/Convert"/>
    </wsdl:port>
  </wsdl:service>
</wsdl:definitions>
```

Notes on the previous WSDL file

- The name of the web service “ConvertService” is specified in a service element.

```
<wsdl:service name="ConvertService">
```

- The port element of the service specifies an URL address of the web service and access point of this web service for a unique binding.

```
<wsdl:port binding="intf:ConvertSoapBinding" name="Convert">
    <wsdlsoap:address
location="http://localhost:8080/axis/services/Convert"/>
```

- The binding “ConvertSoapBinding” is defined in a binding element which is referenced in binding attribute of port sub-element of service in WSDL definition element. It could be multiple ports in one web service (example: input/output ports).

Notes on the previous WSDL file

- There are two message elements at the beginning of definition element:
  - The first message describes the argument type of requested Web service's remote method

```
<wsdl:message name="toFahrenheitRequest">
<wsdl:port name="in0" type="SOAP-ENC:string"/>
```

  - The second message is the return (response) type of the same method.

```
<wsdl:message name="toFahrenheitResponse">
<wsdl:port name="return" type="SOAP-ENC:string"/>
```

- The port sub-element specifies the name and data type of the message exchanged.

```
<wsdl:port name="in0" type="SOAP-ENC:string"/>
<wsdl:port name="return" type="SOAP-ENC:string"/>
```

Notes on the previous WSDL file

- The portType element describes an operation provided by the web service.

```
<wsdl:portType name="Convert">
<wsdl:operation name="toFahrenheit" parameterOrder="in0">
```

- A portType is a collection of operations.
- The operation describes the name of the method in web service called to perform the client request. The client must know about the input parameters and return data type.
- The two messages defined for a web service (request and response) are specified in the operation element.

```
<wsdl:input message="intf:toFahrenheitRequest"/>
<wsdl:output message="intf:toFahrenheitResponse"/>
```

Notes on the previous WSDL file

- The binding elements inside definition element specify the binding style; how a client and the web service should exchange messages.

- The binding sub-element specifies that input request and output response must be in a SOAP format.

```
<wsdlsoap:binding style="rpc"
transport="http://schemas.xmlsoap.org/soap/http"/>
```

- It also tells that this is a request/response (two ways) operation by the attribute style:
  - rpc // a Remote Procedure Call (RPC) style
  - document style
- There are four operation types in terms of patterns of inputs and outputs:
  - input only
  - output only
  - input/output
  - output/input.

Universal Description Discovery Integration (UDDI)

- A technical specification for building a distributed directory for business web services that enables companies publishing and marketing their web services.

- A group of specifications that lets web service providers publish information about their services and permit the clients search that information and run it.

- Web services are organized in a three-level nested structure model with:
  - business information
  - service information
  - binding information

- If a client knows in advance the location of the web service and the way to invoke the operations provided by this web service, then there is no need for web service registry.

UDDI Structure

- UDDI acts like a naming service in the distributed computing
  - Example: phone directory for phone service, or Google search engine for Internet service.

- UDDI consists of:
  - XML schema that defines UDDI's core data structures (business, service, binding and tModel*) programmatic interface.
  - A set of APIs that provide publishing and inquiry operations on those structures.

* tModel is a data structure representing a service type (a generic representation of a registered service) in the UDDI

UDDI Connection Model

1. A web service listing is created using WSDL and then sent to a UDDI registry which is mapped to a UDDI XML format document.

2. A web service client searches the service registry and finds the desired service description.

3. Through the registry interface, the client connects to the web service provider and invokes the service.

UDDI Connection Model

- A web service listing consists of three elements:
  - At the highest level there are *White* Pages, which contain basic information about the business including business name, descriptions, contact info (name, address, phone, fax, Web site) (business information).
  - Next are *Yellow* Pages, which organize services by industry codes, service type/business categories in product/services or geographical location taxonomy (service information).
  - Finally there are *Green* Pages, which specify how to bind to a service provider. It includes the technical information, such as interfaces and URL locations, and how to find and execute a Web service (binding information).
- An application requesting a service will use WSDL to programmatically interact with the Green Pages section of that service's listing.

UDDI Connection Model