Web-Services

Desarrollo de Aplicaciones Cliente-Servidor

Dr. Jorge E. Villaverde

Web Service definition

- A simple Definition:
- "A Web Service is an application component accessible over open protocols"

WS Definition

- A software **component** that can be described using a service-description language, which is in formal XML notation, covers all the details necessary to interact with the service (message formats for operations, transport protocols and location), and hides the implementation details of the service;
- Published to a registry of services;
- Discovered through a standard mechanism;
- Invoked through a declared API, usually through a network; and
- Composed with other services enabling loosely coupled, component-oriented, cross-technology application implementations.

Programming Interfaces and APIs

- API = Application Programming Interface
- Don’t confuse this with a user Interface
- Defines a set of functions that may be called by application programs like a library
- But application developer may not have access to the code implementing the functions
- E.g., being an eBay developer means you can use the API, not that you can see how it is implemented and the functions may even be executed on a different computer!
- E.g., eBay developers are using functions located on eBay servers, not their own computers

Evolution

- Web services evolved from previous technologies that served the same purpose such as RPC, ORPC (DCOM, CORBA and JAVA RMI).

- Web Services were intended to solve three main problems:

1. Interoperability
2. Firewall traversal
3. Complexity

Interoperability

- Earlier distributed systems suffered from interoperability issues because each vendor implemented its own on-wire format for distributed object messaging.
- Development of DCOM apps strictly bound to Windows Operating system.
- Development of RMI bound to Java programming language.

Distributed Object models

- DCOM is a Protocol that enables software components to communicate directly over a network in a reliable, secure, and efficient manner. Previously called OLE, DCOM is designed for use across multiple network transports, including Internet Protocols such as HTTP.

- RMI is an RPC mechanism enabling Java programmers to create distributed applications, in which the methods of remote Java objects can be invoked from another JVM, possibly on a different host.

- CORBA describes the architecture of a middleware platform that supports the implementation of applications in distributed and heterogeneous environments. The CORBA standard is issued by OMG. In contrast to other middleware platforms such as Microsoft's DCOM, CORBA is a specification that does not prescribe any specific technology.

Firewall traversal

- Collaboration across corporations was an issue because distributed systems such as CORBA and DCOM used non-standard ports.
- Web Services use HTTP as a transport protocol and most of the firewalls allow access though port 80 (HTTP), leading to easier and dynamic collaboration.

Complexity

- Web Services is a developer-friendly service system.
- Most of the above-mentioned technologies such as RMI, COM, and CORBA involve a whole learning curve.
- New technologies and languages have to be learnt to implement these services.

Web Service definition revisited

A more precise definition:

- an application component that:
- Communicates via open protocols (HTTP, SMTP, etc.)
- Processes XML messages framed using SOAP
- Describes its messages using XML Schema
- Provides an endpoint description using WSDL
- Can be discovered using UDDI

Web Services Components

XML – eXtensible Markup Language – A uniform data representation and exchange mechanism.
SOAP – Simple Object Access Protocol – A standard way for communication.
UDDI – Universal Description, Discovery and Integration specification – A mechanism to register and locate WS based application.
WSDL – Web Services Description Language – A standard meta language to described the services offered.

The Web Service Model

The Web Services architecture is based upon the interactions between three roles:

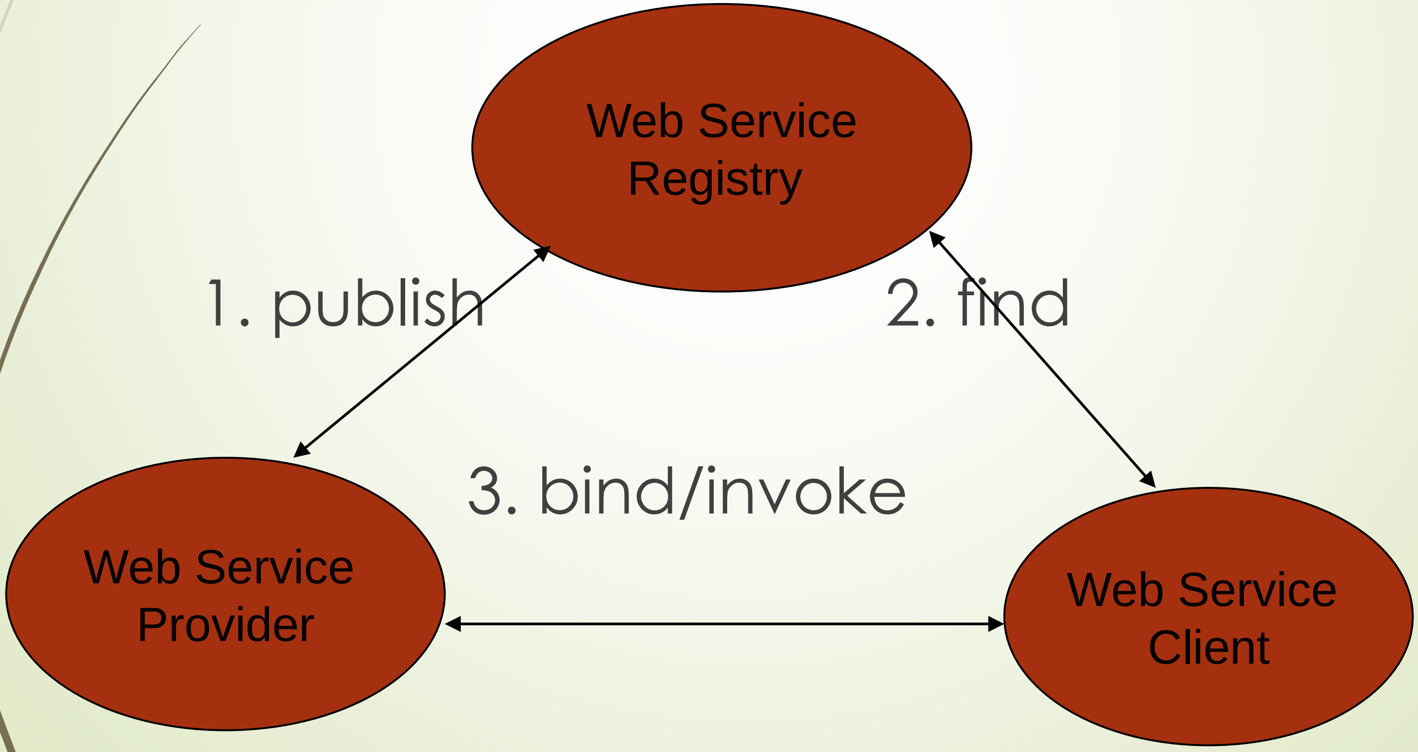
- Service provider
- Service registry
- Service requestor

The interactions involve the:

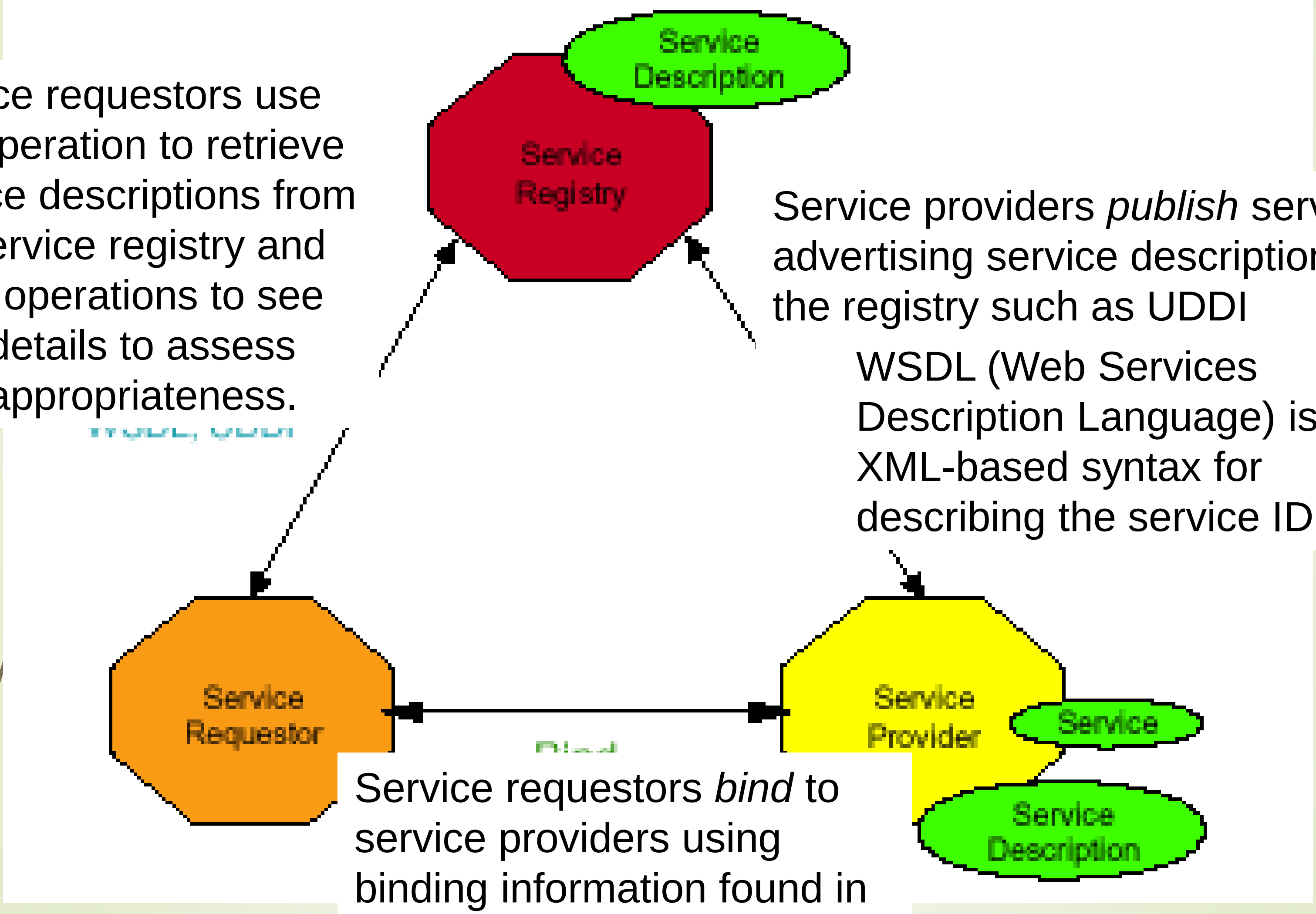
- Publish operations
- Find operation
- Bind operations.

The Web Service Model (cont)

The Web Services model follows the *publish*, *find*, and *bind* paradigm.



Service-Oriented Architectures



Service requestors use *find* operation to retrieve service descriptions from the service registry and *show* operations to see their details to assess their appropriateness.

Service providers *publish* services by advertising service descriptions in the registry such as UDDI

WSDL (Web Services Description Language) is an XML-based syntax for describing the service IDL

Service requestors *bind* to service providers using binding information found in service descriptions to locate and invoke a service.

XML

- XML stands for E**X**tensible **M**arkup **L**anguage.
- XML is a **markup language** much like HTML.
- XML was designed to **describe data**.
- XML tags are not predefined. You must **define your own tags.**
- The prefect choice for enabling cross-platform data communication in Web Services.

SOAP

- SOAP originally stood for "Simple Object Access Protocol" .
- Web Services expose useful functionality to Web users through a standard Web protocol called SOAP.
- Soap is an XML vocabulary standard to enable programs on separate computers to interact across any network. SOAP is a simple markup language for describing messages between applications.
- Soap uses mainly HTTP as a transport protocol. That is, HTTP message contains a SOAP message as its payload section.

SOAP Characteristics

- SOAP has three major characteristics:
  - Extensibility – security and WS-routing are among the extensions under development.
  - Neutrality - SOAP can be used over any transport protocol such as HTTP, SMTP or even TCP.
  - Independent - SOAP allows for any programming model.

SOAP Building Blocks

A SOAP message is an ordinary XML document containing the following elements:

- A required Envelope element that identifies the XML document as a SOAP message.
- An optional Header element that contains header information.
- A required Body element that contains call and response information.
- An optional Fault element that provides information about errors that occurred while processing the message.

SOAP Security

- SOAP uses HTTP as a transport protocol and hence can use HTTP security mainly HTTP over SSL.
- But, since SOAP can run over a number of application protocols (such as SMTP) security had to be considered.
- The WS-Security specification defines a complete encryption system.

WSDL

- WSDL stands for Web Services Description Language.
- WSDL is an XML vocabulary for describing Web services. It allows developers to describe Web Services and their capabilities, in a standard manner.
- WSDL specifies what a request message must contain and what the response message will look like in unambiguous notation. In other words, it is a contract between the XML Web service and the client who wishes to use this service.
- In addition to describing message contents, WSDL defines where the service is available and what communications protocol is used to talk to the service.

What is REST ?

REST is a term coined by Roy Fielding to describe an **architecture style** of networked systems. REST is an acronym standing for Representational State Transfer.

Rest – An architectural Style

- **Elements**
  - **Components** – Proxy , gateway etc
  - **Connectors** – client , server etc
  - Data – resource , representation etc

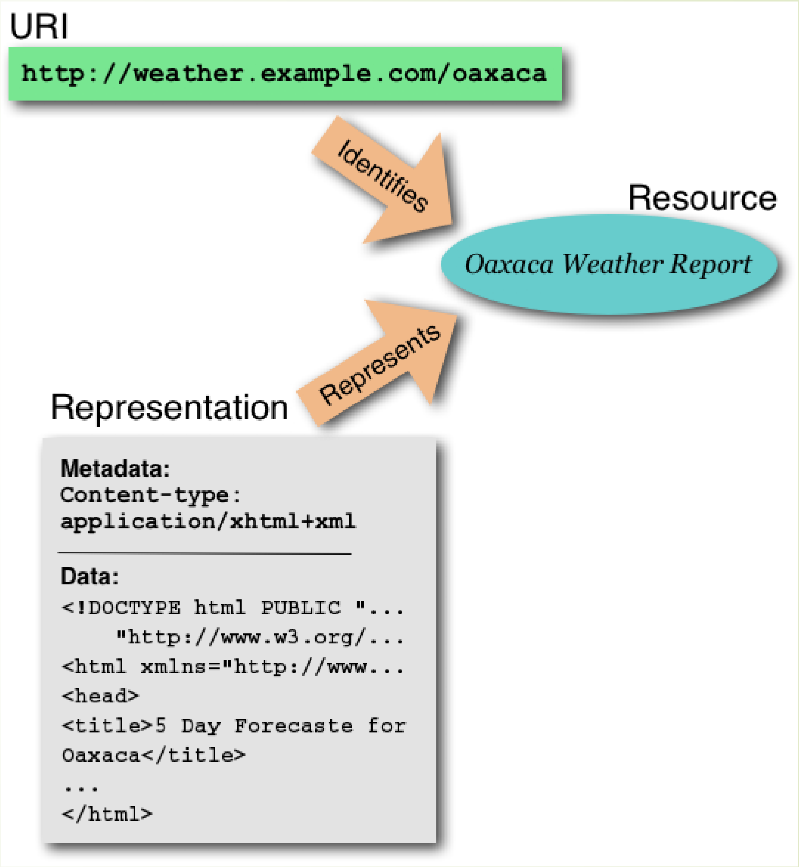
- **REST**
  - Ignores component implementation details.
  - Focus on roles of components,their interactions and their interpretation of data elements.

Example

Resource

URI-Uniform Resource Identifier (or URL)

Web Page (HTML Page)



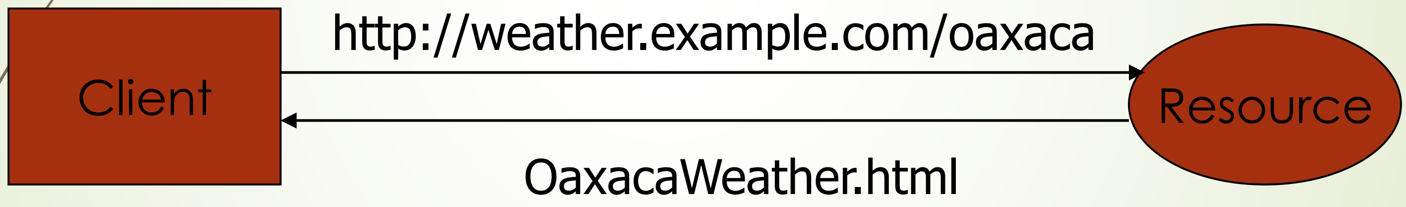
Metadata:

```
Content-type:
application/xhtml+xml
```

Data:

```
<!DOCTYPE html PUBLIC "...
    "http://www.w3.org/...
<html xmlns="http://www...
<head>
<title>5 Day Forecaste for
Oaxaca</title>
...
</html>
```

Why is it called Representational State Transfer?



"Representational State Transfer is intended to evoke an image of how a well-designed Web application behaves: a network of web pages (a virtual state-machine), where the user progresses through an application by selecting links (state transitions), resulting in the next page (representing the next state of the application) being transferred to the user and rendered for their use."

Roy Fielding.

REST - An Architectural Style of Networked System

- Underlying Architectural model of the world wide web.
- Guiding framework for Web protocol standards.
- REST based web services
  - Online shopping
  - Search services
  - Dictionary services

Characteristics of a REST based network

Client-Server: a pull-based interaction style(Client request data from servers as and when needed).

Stateless: each request from client to server must contain all the information necessary to understand the request, and cannot take advantage of any stored context on the server.

Cache: to improve network efficiency, responses must be capable of being labeled as cacheable or non-cacheable.

Uniform interface: all resources are accessed with a generic interface (e.g., HTTP GET, POST, PUT, DELETE).

Named resources - the system is comprised of resources which are named using a URL.

Interconnected resource representations - the representations of the resources are interconnected using URLs, thereby enabling a client to progress from one state to another.

Principles of REST web service design

1.Identify all the conceptual entities that we wish to expose as services. (Examples we saw include resources such as : parts list, detailed part data, purchase order)
2.Create a URL to each resource.
3.Categorize our resources according to whether clients can just receive a representation of the resource (using an HTTP GET), or whether clients can modify (add to) the resource using HTTP POST, PUT, and/or DELETE).
4.All resources accessible via HTTP GET should be side-effect free. That is, the resource should just return a representation of the resource. Invoking the resource should not result in modifying the resource.
5.Put hyperlinks within resource representations to enable clients to drill down for more information, and/or to obtain related information.
6.Design to reveal data gradually. Don't reveal everything in a single response document. Provide hyperlinks to obtain more details.
7.Specify the format of response data using a schema (DTD, W3C Schema, RelaxNG, or Schematron). For those services that require a POST or PUT to it, also provide a schema to specify the format of the response.
8.Describe how our services are to be invoked using either a WSDL document, or simply an HTML document.

Summary

REST – Is an architectural style.

It is the architectural style of the WEB

Resource

http://www.ics.uci.edu/~fielding/pubs/dissertation/rest_arch_style.htm

Preguntas