NOM POUPA

Prénom Adrien

Promo M1 2018

Date 13/01/17

POUPA Adrien
M1 - 2016

1/2

MATIÈRE XML Web Servic

Questions about XML

100 %

1. Unlike DTDs, XML schemas are richer and more complete. They allow data typing (simple and complex types, simple attributs). They can be used to define constraints (optional or mandatory occurence, number of occurences). Finally, they offer a namespace feature to prevent code duplication and inheritance.

10 pts.

2. One should use SAX over DOM when speed and a low memory footprint is required since the whole XML file is not loaded into memory.

Exercise XML schema and Xpath

1. In section 2, "Chapitre 1" should be between quotes.
In Section 2, Chapitre 2, the second paragraph starts with a closing tag </paragraphe> instead of <paragraphe>.

10 pts.

10

70 %

2. livres.xsd file :

```
<?xml version="1.0" ?>
  <xsd:schema>
    <xsd:attribute name="titre" type="typeInfo"/>
    <xsd:complexType name="titreInfo">
      <xsd:sequence>
        <xsd:attribute name="titre" type="xsd:string"/>
      </xsd:sequence>
    </xsd:complexType>

    <xsd:complexType name="auteursInfo">
      <xsd:sequence>
        <xsd:element name="auteur">
          <xsd:attribute name="nom" type="xsd:string"/>
          <xsd:attribute name="prenom" type="xsd:string"/>
        </xsd:element>
      </xsd:sequence>
    </xsd:complexType>

    <xsd:complexType name="chapitreInfo">
      <xsd:extension base="titreInfo">
        <xsd:sequence>
          <xsd:element name="paragraphe" type="xsd:sting"/>
        </xsd:sequence>
      </xsd:extension>
    </xsd:complexType>
```

```
<xsd:complexType name="sectionInfo">
  <xsd:extension base="titreInfo">
    <xsd:sequence>
      <xsd:element name="chapitre" type="chapitre
                                         Info"/>
    </xsd:sequence>
  </xsd:extension>
</xsd:complexType>

<xsd:complexType name="sectionsInfo">
  <xsd:sequence>
    <xsd:element name="section" type="sectionInfo"
  </xsd:sequence>
</xsd:complexType>

<xsd:complexType name="livreInfo">
  <xsd:extension base="titreInfo">
    <xsd:sequence>
      <xsd:element name="auteurs" type="auteurs
                                        Info"/>

      <xsd:element name="sections" type=
                                  "sectionsInfo"/>
    </xsd:sequence>
  </xsd:extension>
</xsd:complexType>
```

```
<xsd:element name="livre">
    <xsd:complexType>
        <xsd:sequence>
            <xsd:element name="livre" type="livreInfo"/>
        </xsd:sequence>
    </xsd:complexType>
</xsd:element>

</xsd:schema>
```

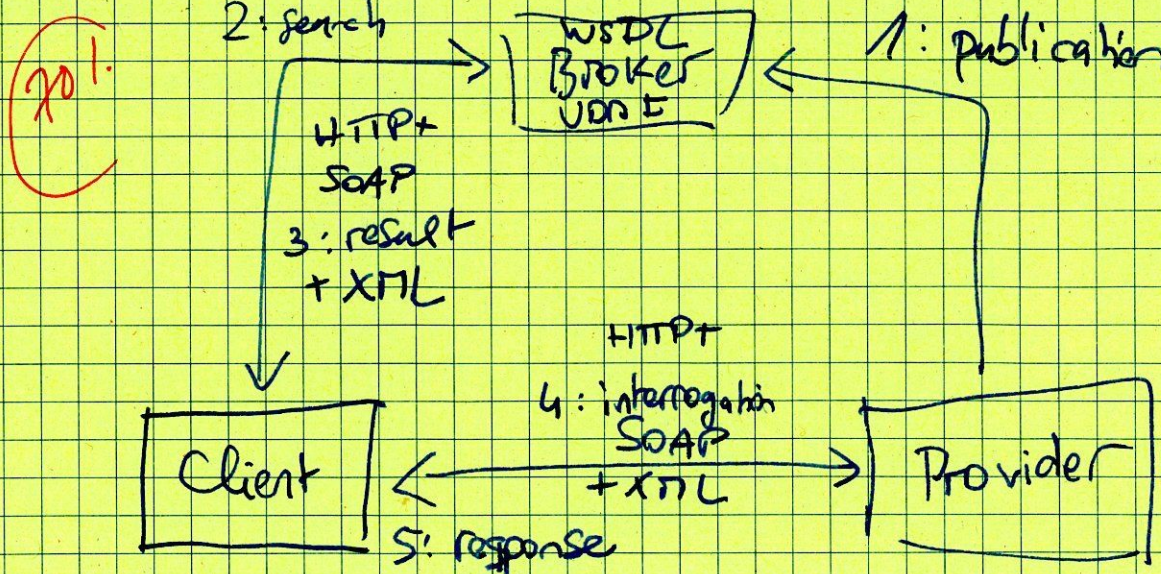
3. First expression :

- livre/livre/section[@titre="Section 2"]/chapitre[@titre="chapitre 1"]/paragraphe[last()]

Second expression :

- livre/livre/sections/section[1]/@titre

Questions SOAP

- 

NOM POURA

Prénom Adrien

Promo M1 2018

Date 13/01/17

MATIÈRE XML

- WSDL is a language defining the interface of a web service. It describes the public interface of web services using XML.

- If we use the API JAX-WS we can use a simple POJO class. There is no need for an interface or EJB but we have to generate a WSDL file, unlike if we used a stateless session EJB.

- We have to regenerate the WSDL because it describes what the service do / return.
If we add a parameter, the contract will change and the WSDL must be aware of it.
To do it automatically, we can use a dynamic client.

REST Questions.

- Jersey is the reference application of Jax-RS, a framework used to write RESTful web services.
Jackson has not been mentioned during class.(

- REST webservices are bookmarkables because they can be accessed from a unique URL that can be bookmarked.
because each resource has an unique

(101)

- At the top of the POJO class, put a simple @XmlRootElement and in the service class, add a @Produces({MediaType.TEXT_XML}) or any type wanted at the top of the function.

- Rest is not really a technology or a protocol because it is entirely based on HTTP and its own protocols.
eg POST - create a resource
DELETE - delete a resource
PATCH - edit a resource
PUT - replace a resource
GET - read a resource
all accessible from a simple URL with the right HTTP protocol
It works with a simple client/server.
It is also stateless.

(501.)

not really