

Contrôle services Web

vendredi 27 novembre 2009

1 APIs Web services et Java (3 pts)

1. Expliquer en quelques phrases ce qu'est une correspondance XML/Objet et comment elle peut être utilisée dans le cadre de l'implémentation de services Web.
2. On considère un service web dont les messages correspondent à un certain schéma XML. On suppose que l'on a un client capable d'utiliser un autre service offrant les mêmes fonctionnalités, mais utilisant un schéma XML différent pour les messages. On suppose que la structure des messages reste la même, mais que les espaces de nommage et le nom des éléments et des attributs changent. On vous demande d'implémenter un proxy permettant au client d'utiliser le service. Donner les technologies/APIs que vous emploieriez et pourquoi.

2 XML Schema (5 pts)

Définir en XML Schema un format XML pouvant être utilisé pour l'export d'une base de données comportant:

- Un ensemble de personnes dont on connaît le nom, le prénom, le salaire et le numéro d'employé.
- Un ensemble de départements ayant un nom et dans lequel se trouvent les employés, chaque département pouvant éventuellement être un sous-département d'un département plus gros, et ceci avec une profondeur arbitraire.
- Un ensemble de projets ayant un titre. Chaque projet est rattaché à un département et un ensemble arbitraire de personnes peut travailler sur le projet.

On demande à ce que les éléments soient définis dans l'espace de nommage `urn:entreprise:departements-projets`.

3 SOAP (3 pts)

Donner un exemple de messages SOAP échangés lors de l'utilisation de l'opération `rechercheHotelOperation` du fichier `reservation.wsdl`.

4 BPEL (9 pts)

On souhaite réaliser en BPEL un service de réservation couplant un billet de train avec une nuit à l'hôtel. L'ensemble des interfaces des services entrant en jeu sont décrites dans le fichier `reservation.wsdl`.

On précise que le `lookupPortType` est l'interface d'un service combinant les fonctions de recherche de disponibilité (train/hôtel) et d'annuaire de service. On précise également que les adresses des services de réservation de train et d'hôtel dépendent de la demande de réservation du client.

- Donner la liste des différents acteurs, en dehors du service de réservation à implémenter, avec les interfaces (`portType`) utilisées ou exposées par ces services. Dire pour chaque acteur si ses interactions avec le processus BPEL sont synchrones ou asynchrones.
- Décrire, avec leur enchaînement, les activités BPEL à utiliser pour implémenter le service.

- On suppose que l'identifiant transmis avec la demande est unique. Définir un ou plusieurs ensembles de corrélation, expliquer avec quels envois et/ou réception de messages ils sont utilisés et dans le(s)quel(s) ils prennent leur valeur de départ.

Fichier reservation.wsdl

```
<?xml version="1.0" encoding="UTF-8"?>
<definitions name="reservation"
  targetNamespace="http://master-info.univ-lyon1.fr/reservation"
  xmlns="http://schemas.xmlsoap.org/wsdl/"
  xmlns:wSDL="http://schemas.xmlsoap.org/wsdl/"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:tns="http://master-info.univ-lyon1.fr/reservation">
  <types>
    <xsd:schema targetNamespace="http://master-info.univ-lyon1.fr/reservation"
      xmlns:tns="http://master-info.univ-lyon1.fr/reservation">
      <xsd:element name="demandeReservation">
        <xsd:complexType>
          <xsd:sequence>
            <xsd:element name="villeDepart" type="xsd:string"/>
            <xsd:element name="villeSejour" type="xsd:string"/>
            <xsd:element name="date" type="xsd:date"/>
          </xsd:sequence>
          <xsd:attribute name="idDemande" type="xsd:string"/>
        </xsd:complexType>
      </xsd:element>
      <xsd:element name="reservation">
        <xsd:complexType>
          <xsd:sequence>
            <xsd:element name="train">
              <xsd:complexType>
                <xsd:attribute name="numero" type="xsd:integer"/>
                <xsd:attribute name="place" type="xsd:integer"/>
              </xsd:complexType>
            </xsd:element>
            <xsd:element name="hotel" type="xsd:string"/>
            <xsd:element name="prix" type="xsd:double"/>
          </xsd:sequence>
        </xsd:complexType>
      </xsd:element>
      <xsd:element name="reservationHotel">
        <xsd:complexType>
          <xsd:sequence>
            <xsd:element name="date" type="xsd:date"/>
          </xsd:sequence>
        </xsd:complexType>
      </xsd:element>
      <xsd:element name="confirmationHotel">
        <xsd:complexType>
          <xsd:sequence>
            <xsd:element name="prix" type="xsd:double"/>
          </xsd:sequence>
        </xsd:complexType>
      </xsd:element>
    </xsd:schema>
  </types>

```

```

</xsd:element>
<xsd:element name="reservationTrain">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="date" type="xsd:date"/></xsd:element>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
<xsd:element name="confirmationTrain">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="prix" type="xsd:double"/></xsd:element>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
<xsd:element name="rechercheHotel">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element ref="tns:demandeReservation"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
<xsd:element name="hotelService" type="tns:adresseType"/></xsd:element>
<xsd:element name="rechercheCompanie">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element ref="tns:demandeReservation"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
<xsd:element name="companieService" type="tns:adresseType"/></xsd:element>
<xsd:complexType name="adresseType">
  <xsd:sequence>
    <xsd:element name="adresse" type="xsd:string"/></xsd:element>
  </xsd:sequence>
</xsd:complexType>
</xsd:schema>
</types>
<message name="demandeReservationRequest">
  <part name="part1" element="tns:demandeReservation"/>
</message>
<message name="demandeReservationResponse">
  <part name="part1" element="tns:reservation"/>
</message>
<message name="reservationHotelRequest">
  <part name="part1" element="tns:reservationHotel"/>
</message>
<message name="reservationHotelResponse">
  <part name="part1" element="tns:confirmationHotel"/>
</message>
<message name="reservationTrainRequest">
  <part name="part1" element="tns:reservationTrain"/>
</message>
<message name="reservationTrainResponse">

```

```

    <part name="part1" element="tns:confirmationTrain"/>
</message>
<message name="rechercheHotelRequest">
    <part name="part1" element="tns:rechercheHotel"/>
</message>
<message name="rechercheHotelResponse">
    <part name="part1" element="tns:hotelService"/>
</message>
<message name="rechercheCompanieTrainRequest">
    <part name="part1" element="tns:rechercheCompanie"/>
</message>
<message name="rechercheCompanieTrainResponse">
    <part name="part1" element="tns:companieService"/>
</message>
<portType name="reservationServicePortType">
    <operation name="reservationOperation">
        <input name="input1" message="tns:demandeReservationRequest"/>
    </operation>
</portType>
<portType name="reservationClientPortType">
    <operation name="reservationReponseOperation">
        <input name="input1" message="tns:demandeReservationResponse"/>
    </operation>
</portType>
<portType name="lookupPortType">
    <operation name="rechercheHotelOperation">
        <input name="input1" message="tns:rechercheHotelRequest"/>
        <output name="output1" message="tns:rechercheHotelResponse"/>
    </operation>
    <operation name="rechercheCompanieOperation">
        <input name="input2" message="tns:rechercheCompanieTrainRequest"/>
        <output name="output2" message="tns:rechercheCompanieTrainResponse"/>
    </operation>
</portType>
<portType name="trainPortType">
    <operation name="reservationTrainOperation">
        <input name="input1" message="tns:reservationTrainRequest"/>
        <output name="output1" message="tns:reservationTrainResponse"/>
    </operation>
</portType>
<portType name="hotelServicePortType">
    <operation name="reservationHotelOperation">
        <input name="input1" message="tns:reservationHotelRequest"/>
    </operation>
</portType>
<portType name="hotelClientPortType">
    <operation name="reservationHotelReponseOperation">
        <input name="input1" message="tns:reservationHotelResponse"/>
    </operation>
</portType>
</definitions>

```