

Labo 3 : Web Service SOAP

Raphaël Javaux

Création du projet Dans NetBeans, créer un projet *Application Web*. Il est ensuite possible de rajouter des Web Services SOAP via *Nouveau fichier*. Sélectionner création d'un nouveau Web Service et non basé sur un fichier WSDL existant.

Définition du Web Service La classe qui contiendra le Web Service aura l'annotation *@WebService* et toutes ses méthodes d'opération devront porter l'annotation *@WebMethod*. Ces méthodes doivent être publiques.

```
@WebService(serviceName = "WebServiceSOAP")
public class WebServiceSOAP {
    @WebMethod(operationName = "HistoriqueVols")
    public @WebResult(name="Vol") Vol[] HistoriqueVol()
    {
        [...]
    }
}
```

Classes de retour Les classes utilisées comme valeur de retour pour les opérations du Web Service devront comporter un constructeur par défaut ainsi qu'une annotation *@XmlElement* sur les accesseurs.

```
public class Vol {
    private Date _depart;
    @XmlElement(name="depart")
    public Date getDepart() {
        return _depart;
    }

    public void setDepart(Date depart) {
        this._depart = depart;
    }
}
```

Handler La classe utilisée comme handler pour les messages SOAP doit dériver de la classe générique *SOAPHandler* (le paramètre générique de la classe définit le type du handler). Le handler doit ensuite être enregistré. Cela peut se faire à l'aide de l'opération *Configurer les handlers* du menu clic-droit du Web Service.

```
public class AuthHandler
    implements SOAPHandler<SOAPMessageContext>
{
    public boolean handleMessage(SOAPMessageContext context)
    {
        [...]
        return true;
    }
}
```

Test du Web Service Netbeans permet de directement déployer et tester le Web Service sur un serveur d'application (Glassfish, Tomcat ...) dans une application JSP (via un clic droit sur le Web Service). Les exceptions qui seraient apparues lors d'exécution du Web Service se trouveront dans les logs du serveur d'application.

Utilisation du Web Service en Ruby La bibliothèque *Savon* peut être utilisée pour accéder à des Web Services SOAP en Ruby. Il suffit de lui fournir l'URI du fichier WSDL. Les exemples sur le site officiel sont suffisants pour utiliser l'API.

```
require "savon"

client = Savon::Client.new do |wsdl, http, wsse|
  wsdl.document = wsdl_uri
  wsse.credentials user, pass
end

client.request(:wsdl, :historique_vols)
```