

Déploiement et Invocation de services web Java avec Axis

Dans ce TP, il s'agit de découvrir comment développer et déployer un service web simple en utilisant les outils Axis et Tomcat.

I- Lancement et configuration d'Axis

Le serveur de déploiement « Axis » est installé comme une application Web au sein du serveur web et du moteur de servlets et de JSPs « Apache Tomcat ». Pour installer Axis (on va utiliser version 1.4), décompresser le fichier « axis-web.zip » (disponible sur <http://www.redcad.org/members/tarak.chaari/cours/qos/axis-web.zip>) dans le répertoire « webapps » de tomcat. Comme ça Axis sera une application web déployée sous Tomcat.

Pour vérifier la bonne installation et éventuellement accéder aux services web existants sur « Axis », Vous devez démarrer Tomcat et vous tapez dans un navigateur : `http://localhost:8080/axis`

Ensuite, vous pouvez cliquer sur « Validation » pour vérifier que Axis a été bien installé dans le serveur Tomcat. En plus, vous pouvez lister les services déjà hébergé grâce au lien « List ».

II- Développement et déploiement : méthode 1

a- Création du web service

La première étape consiste à définir la classe Calculator du service web qui offre deux opérations simples à ses clients : l'addition et la multiplication de deux entiers. Cette classe est une classe Java standard sans aucune différence par rapport au développement Orienté objet classique. La seule exception dans cette première étape est que le fichier source doit être enregistré sous l'extension jws et non java. Ici, le fichier de sauvegarde sera donc : « Calculator.jws ».

b- Déploiement du web service

La deuxième étape consiste à déployer le service (le fichier Calculator.jws) au sein d'un fournisseur de services web. L'environnement d'exécution et de déploiement des services web que nous utilisons est, comme vous l'avez déjà constaté, l'outil « Axis ». Le premier mode de déploiement sous « Axis » que nous allons mettre en œuvre est le plus simple qui soit ; c'est le déploiement instantané par fichier « JWS ».

Pour réaliser le déploiement, il suffit de copier le fichier « Calculator.jws » dans le domaine applicatif de « Axis ». Ce domaine n'est que le répertoire axis que vous venez de copier dans Tomcat. Vous êtes désormais en mesure d'accéder à votre service à l'URL suivant : <http://localhost:8080/axis/Calculator.jws>.

c- Test du web service

La dernière étape consiste à mettre en œuvre votre service qui est désormais accessible à travers tout le net ! Pour exécuter une méthode de votre service et obtenir la réponse « SOAP » correspondante, vous tapez un URL de ce style dans votre navigateur : <http://localhost:8080/axis/Calculator.jws?method=add&a=2&b=3>

Vous pouvez aussi tester le fonctionnement de ce service en développant un Client Java basé sur le descripteur WDSL généré par Axis sur la page d'accueil de ce service.

III- Développement et déploiement : méthode 2

Dans la partie précédente, nous avons mis en œuvre un mode de déploiement entièrement pris en charge par « Axis ». Ce mode automatique de déploiement présente la contrainte suivante : Nécessité de disposer des sources des classes du service. En effet, le mode de déploiement automatique d'Axis travaille sur la source des classes Java, Impossibilité de décrire des particularités de déploiement (classes publiques et privées...). Pour cette raison, que nous devons parfois réaliser un déploiement explicite. Cela implique la définition d'un fichier particulier, appelé descripteur de déploiement du service web. Ce fichier porte l'extension « wsdd » pour « Web Service Deployment Descriptor ».

a- définition du descripteur de déploiement

Le descripteur de déploiement d'un service web sous Axis « deploy.wsdd » est un fichier XML contenant les balises minimales suivantes :

```
<deployment xmlns="http://xml.apache.org/axis/wsdd/"
  xmlns:java="http://xml.apache.org/axis/wsdd/providers/java">
  <service name="Calculator" provider="java:RPC" style="rpc">
    <parameter name="className" value="Calculator"/>
    <parameter name="allowedMethods" value="*" />
  </service>
</deployment>
```

b- déploiement du web service

Le descripteur de déploiement doit être maintenant pris en compte par le fournisseur de services « Axis ». Pour ce faire, il convient d'utiliser l'utilitaire « AdminClient » de ce fournisseur (voir le fichier « deployServiceWSDD.bat » dans le répertoire « WEB-INF/lib » de Axis). La classe du service doit être présente dans ce même répertoire pour que l'outil AdminClient puisse la trouver.

Cela signifie donc qu'à la réception d'une requête « HTTP-SOAP », le serveur pourra appliquer la méthode spécifiée dans la requête à une instance de la classe correspondant à notre service (en lui passant, le cas échéant des valeurs). Vous êtes désormais en mesure d'accéder à votre service à l'URL suivante :

<http://localhost:8080/axis/services/Calculator>.

Vous pouvez alors constater que votre service a bien été déployé sur « Axis ». Le nom « Calculator » correspond au nom du service que nous avons indiqué dans le descripteur de déploiement.

c- *exécution du web service par son alias*

La dernière étape consiste à mettre en œuvre votre service qui est désormais accessible à travers tout le net ! Pour exécuter une méthode de votre service et obtenir la réponse « SOAP » correspondante, vous tapez l'expression suivante dans votre navigateur :

<http://localhost:8080/axis/services/Calculator?method=add&a=2&b=3>

Par rapport à la méthode précédente, vous précisez ici le nom du web service par son alias (ici « Calculator »).

IV- Consommer le service web en Java

Jusqu'à ce stade, nous avons toujours utilisé le navigateur Web pour invoquer les services et visualiser (sous le format SOAP) les résultats retournés. Nous allons étudier maintenant, un autre mode de communication avec les services plus adapté à leur mise en œuvre, et surtout à l'intégration avec d'autres applications. Ce deuxième modèle consiste à la mise en œuvre des Services Web depuis le langage Java en générant automatiquement la définition du Stub côté client. Cela sous-entend la création de toutes les interfaces et classes nécessaires à la mise en œuvre du service depuis Java.

Ce modèle est caractérisé par :

- Utilisation du stub côté client et du skeleton côté serveur
- Utilisation des interfaces représentant les types des objets Java à manipuler

L'outil « Axis » qui permet la génération des définitions Java côté client et côté serveur, s'appelle « org.apache.axis.wsdl.WSDL2Java ». L'invocation depuis une fenêtre DOS, est la commande : « java org.apache.axis.wsdl.WSDL2Java -s calculator.wsdl ». Veuillez consulter le fichier wsdl2java.bat (dans le répertoire WEB-INF/lib) pour voir les jars nécessaires à mettre dans le classpath.

Remarque : le fichier « calculator.wsdl » peut-être obtenu grâce à l'URL : <http://localhost:8080/axis/services/Calculator?wsdl> .

A partir de ce moment, vous pouvez reprendre le TP précédent « IP2Geo » pour développer et tester un client Java du service Calculator.

V- Désactivation du service web

Enfin pour retirer un service déployé, il suffit de ré-appliquer l'utilitaire « AdminClient » sur un fichier « undeploy.wsdd » correspondant au service.

```
<undeployment xmlns="http://xml.apache.org/axis/wsdd/">  
  <service name="Calculator"/>  
</undeployment>
```

Bibliographies:

<http://ws.apache.org/axis/>

<http://www.developpez.net/forums/>

<http://dico.developpez.com>