

# TP Web Services

Préparé par  
M.G. BELKASMI

# Tomcat

- Aller sur <http://tomcat.apache.org/> pour télécharger Tomcat
- Décompresser le fichier et créer les deux variables d'environnement
  - CATALINA\_HOME = chemin vers dossier Tomcat
  - JAVA\_HOME = chemin vers dossier jdk ou jre
- Lancer Tomcat : Tomcat\bin\startup.bat
- Consulter: <http://localhost:8080/>

# Tomcat

[Home](#) [Documentation](#) [Configuration](#) [Examples](#) [Wiki](#) [Mailing Lists](#)

[Find Help](#)

## Apache Tomcat/7.0.67



If you're seeing this, you've successfully installed Tomcat. Congratulations!



### Recommended Reading:

[Security Considerations HOW-TO](#)

[Manager Application HOW-TO](#)

[Clustering/Session Replication HOW-TO](#)

[Server Status](#)

[Manager App](#)

[Host Manager](#)

### Developer Quick Start

[Tomcat Setup](#)

[First Web Application](#)

[Realms & AAA](#)

[JDBC DataSources](#)

[Examples](#)

[Servlet Specifications](#)

[Tomcat Versions](#)

# Axis

- Aller sur <http://axis.apache.org/axis/> pour télécharger Axis
- Décompresser le fichier
- copier l'application Web d'Axis (webapps/axis) dans le répertoire d'applications Web de Tomcat (Repertoire Tomcat/webapps/)
- Consulter: <http://localhost:8080/axis/>

# Axis

## Apache-AXIS

Hello! Welcome to Apache-Axis.

What do you want to do today?

- [Validation](#) - Validate the local installation's configuration  
*see below if this does not work.*
- [List](#) - View the list of deployed Web services
- [Call](#) - Call a local endpoint that list's the caller's http headers (or see its [WSDL](#)).
- [Visit](#) - Visit the Apache-Axis Home Page
- [Administer Axis](#) - [disabled by default for security reasons]
- [SOAPMonitor](#) - [disabled by default for security reasons]

To enable the disabled features, uncomment the appropriate declarations in WEB-INF/web.xml in the webapplication and restart it.

# Axis

- Tester la configuration avec **Validation**:  
<http://localhost:8080/axis/happyaxis.jsp>

## Axis Happiness Page

### Examining webapp configuration

#### Needed Components

- Found SAAJ API ( javax.xml.soap.SOAPMessage ) at an unknown location
- Found JAX-RPC API ( javax.xml.rpc.Service ) at C:\ws\apache-tomcat-7.0.67\webapps\axis\WE

# Création du service

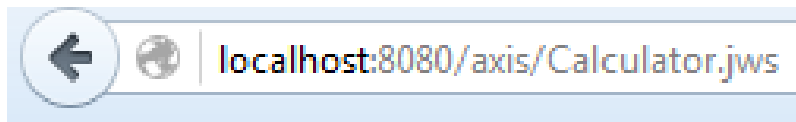
- Code du service Calculator

```
public class Calculator {  
    public int add(int p1, int p2) {  
        return p1 + p2;  
    }  
    public int subtract(int p1, int p2) {  
        return p1 - p2;  
    }  
}
```

# Déploiement du service

- Enregistrer le code sou **Calculator.jws**.
- Placer ce fichier dans le répertoire webapps\axis de Tomcat.
- Le fichier WSDL est généré à la volée par AXIS.  
On peut le consulter sur :

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



There is a Web Service here

[Click to see the WSDL](#)



# Déploiement du service

- Fichier WSDL

```
– <wsdl:definitions targetNamespace="http://localhost:8080/axis/Calculator.jws">
  – <wsdl:message name="addRequest">
    <wsdl:part name="p1" type="xsd:int"/>
    <wsdl:part name="p2" type="xsd:int"/>
  </wsdl:message>
  – <wsdl:message name="subtractRequest">
    <wsdl:part name="p1" type="xsd:int"/>
    <wsdl:part name="p2" type="xsd:int"/>
  </wsdl:message>
  – <wsdl:message name="addResponse">
    <wsdl:part name="addReturn" type="xsd:int"/>
  </wsdl:message>
```

# Création d'un client

- Ce client invoquera les méthodes du service via la classe principale **Call**.

```
import org.apache.axis.client.Call;  
import org.apache.axis.client.Service;  
import org.apache.axis.encoding.XMLType;  
import org.apache.axis.utils.Options;  
  
import javax.xml.rpc.ParameterMode;
```

# Création d'un client

- Code du client CalcClient

```
String endpoint = "http://localhost:8080/axis/Calculator.jws";
```

```
String method = args[0];
```

```
Integer i1 = new Integer(args[1]);
```

```
Integer i2 = new Integer(args[2]);
```

# Création d'un client

- Code du client CalcClient

```
Service  service = new Service();  
Call     call     = (Call) service.createCall();  
  
call.setTargetEndpointAddress(new java.net.URL(endpoint));  
call.setOperationName( method );  
call.addParameter("op1", XMLType.XSD_INT, ParameterMode.IN);  
call.addParameter("op2", XMLType.XSD_INT, ParameterMode.IN);  
call.setReturnType(XMLType.XSD_INT);  
  
Integer ret = (Integer) call.invoke( new Object [] { i1, i2 } );  
  
System.out.println("Got result : " + ret);
```

# Test du client

- Classpath : (variable d'env AXISCLASSPATH)

set AXISCLASSPATH=

.;axis-1\_4\lib\axis.jar;

axis-1\_4\lib\axis-ant.jar;

axis-1\_4\lib\commons-discovery-0.2.jar;

axis-1\_4\lib\commons-logging-1.0.4.jar;

axis-1\_4\lib\jaxrpc.jar;

axis-1\_4\lib\log4j-1.2.8.jar;

axis-1\_4\lib\saaj.jar;

axis-1\_4\lib\wsdl4j-1.5.1.jar

# Test du client

- Compiler :

```
javac -cp %AXISCLASSPATH% CalcClient.java
```

- Exécuter :

```
java -cp %AXISCLASSPATH% CalcClient add 1 2
```

```
Got result : 3
```

```
java -cp %AXISCLASSPATH% CalcClient subtract 4 2
```

```
Got result : 2
```