How to install and run wsig-examples in 5 steps

Author : Sebastien Heymann <sebastien.heymann@tremplin-utc.net>
Date : 2008-29-12

Requirements :

- Be familiar with Eclipse IDE
- Be familiar with Jade
- Knowing how to install and set up Tomcat (out of consideration here)

Notes :

- This document is about the setting of a development environment for WSIG, do not use it in production mode.
- The tutorial is based on the WSIG guide (v2.1), especially the MathAgent example.

Distribution :

- JADE v3.6.1 - http://jade.tilab.com
- WSIG v2.1 - http://jade.tilab.com/community-addons.php

Softwares installed during this tutorial :

- Java Run Time Environment v1.5 necessary (tested in v1.6)
- Eclipse 3.4+ - http://www.eclipse.org
- Eclipse Web Tools Project (WTP) 3.0.3+
- Java EE Developer Tools minimum installation
- Subclipse 1.4.7+ - http://subclipse.tigris.org
- Apache Tomcat 5.5 - http://tomcat.apache.org

Step 1 – Installing Tomcat and Eclipse

First install Tomcat. It is available in official repositories for Ubuntu. In this system, do not forget to remove the autostart daemon by the following command line :

```
sudo update-rc.d -f tomcat5.5 remove
```

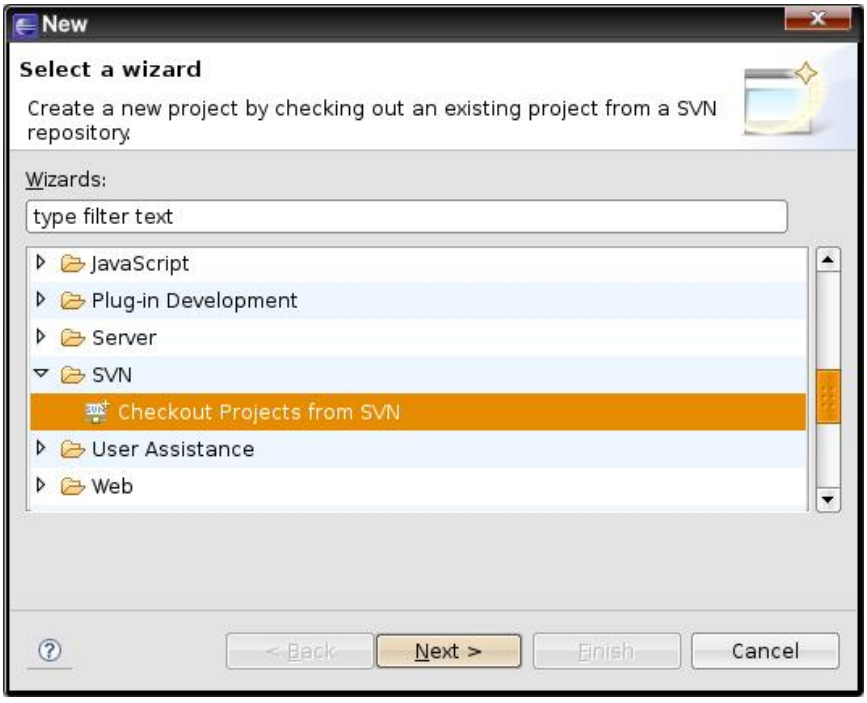
Install the Eclipse you want (http://www.eclipse.org/downloads/). We use the Classic distribution in this tutorial.

Illustration 1: http://www.eclipse.org/downloads/

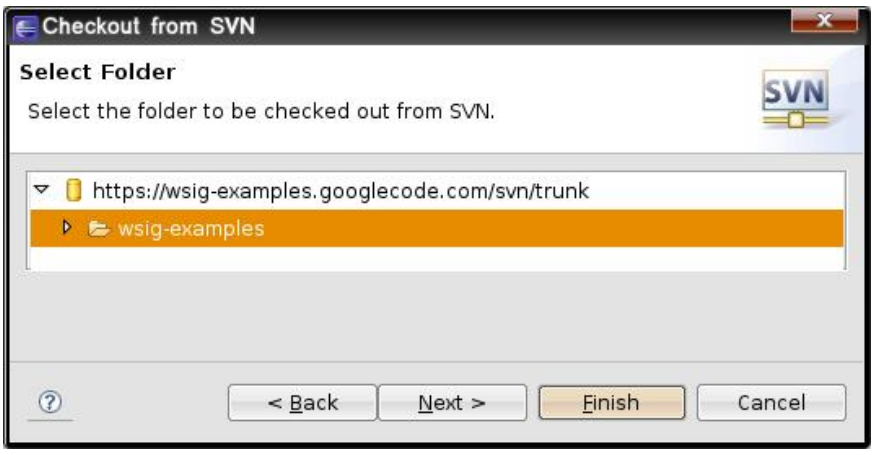
1. Create an empty workspace dedicated to the project, and restart Eclipse to use it.
2. Go to *Help* > *Software Updates*, then add http://download.eclipse.org/webtools/updates.
3. Install Java EE Developer Tools (in Eclipse Web Tools Project (WTP) 3.0.3). Note that Axis is not required because WSIG distribution contains it.
4. Finally add the site http://subclipse.tigris.org/update_1.4.x, and download Subclipse, an integrated SVN tool.

Step 2 – Getting the project

1. Click on *File* > *New* > *Other...* > *SVN* > *Checkout Projects from SVN*



2. Click on *create a new repository location*
3. Then add https://wsig-examples.googlecode.com/svn/trunk/.
4. Select *wsig-examples-read-only* (or *wsig-examples* if you have the right perrmissions for on the SVN) and finish.



This project is a Dynamic Web Project. Once the repository content downloaded and the workspace automatically updated, Eclipse declares an error :

"*Target runtime Apache Tomcat v5.5 is not defined. wsig-examples Unknown Faceted Project Problem*"

This is normal at this stage, we will define a server in step 4.

Step 3 – Jade platform and agents

Create and start the following *Debug Configurations* as *Java Application* :

runJadePlatform

Project	wsig-examples
Main class	jade.Boot
Program arguments	-gui

Click on *Debug* to launch the Jade platform.

runMathAgent

Project	wsig-examples
Main class	jade.Boot
Program arguments	-gui -container "MathAgent1:com.tilab.wsig.examples.MathAgent(MathFunctions false)" -name "WSIGTestPlatform"

Click on *Debug* to launch the MathAgent1 agent, instance of MathFunctions.

runMathAgentPrefix

Project	wsig-examples
Main class	jade.Boot
Program arguments	-gui -container "MathAgent2:com.tilab.wsig.examples.MathAgent(MathFunctions false second)" -name "WSIGTestPlatform"

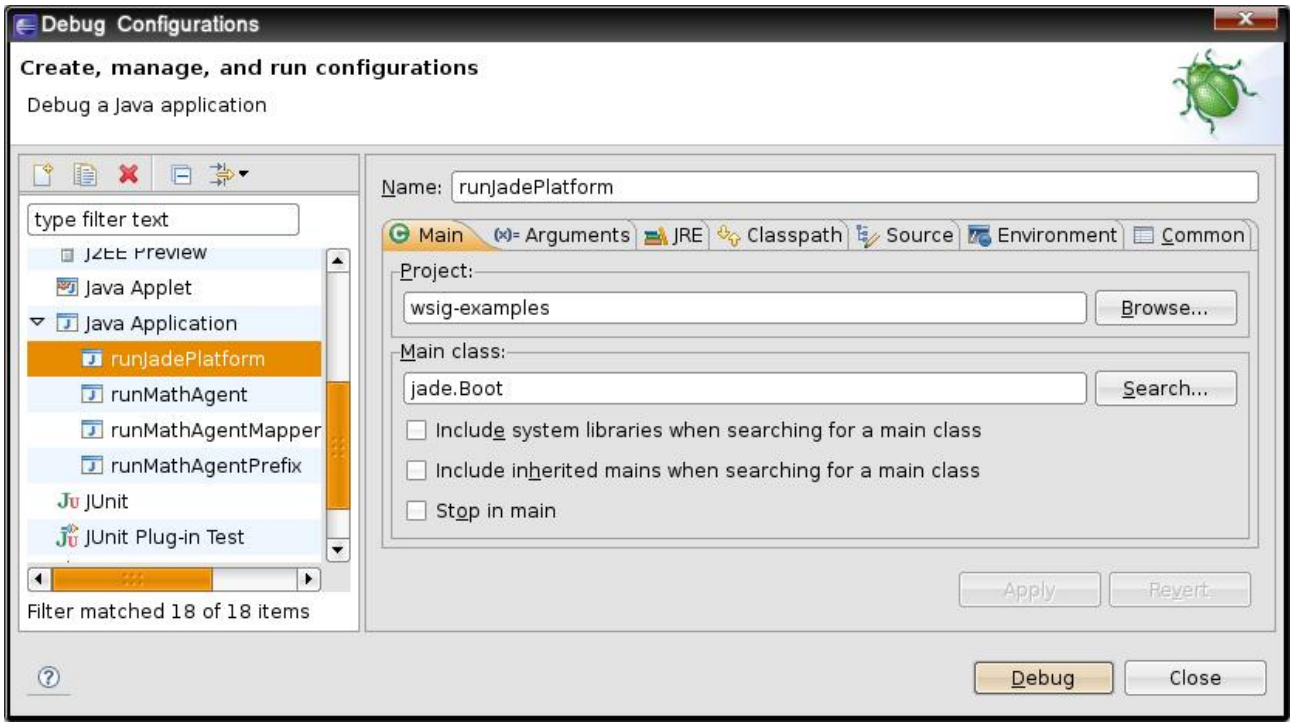
Click on *Debug* to launch the MathAgent2 agent, instance of MathFunctions.

runMathAgentMapper

Project	wsig-examples
Main class	jade.Boot
Program arguments	-gui -container "MathAgent3:com.tilab.wsig.examples.MathAgent(MathFunctionsMapper true)" -name "WSIGTestPlatform"

Click on *Debug* to launch the MathAgent3 agent, instance of MathFunctionsMapper.

Please refer to the WSIG guide for more details on these agents.



Launch

In the future, agents will be directly launchable through the toolbar :
Debug > runJadePlatform
Debug > runMathAgent
Debug > runMathAgentPrefix
Debug > runMathAgentMapper

The Jade Platform GUI must have appeared. You can now see the 3 agents evolving in 3 different containers. These agents are independent from each other.

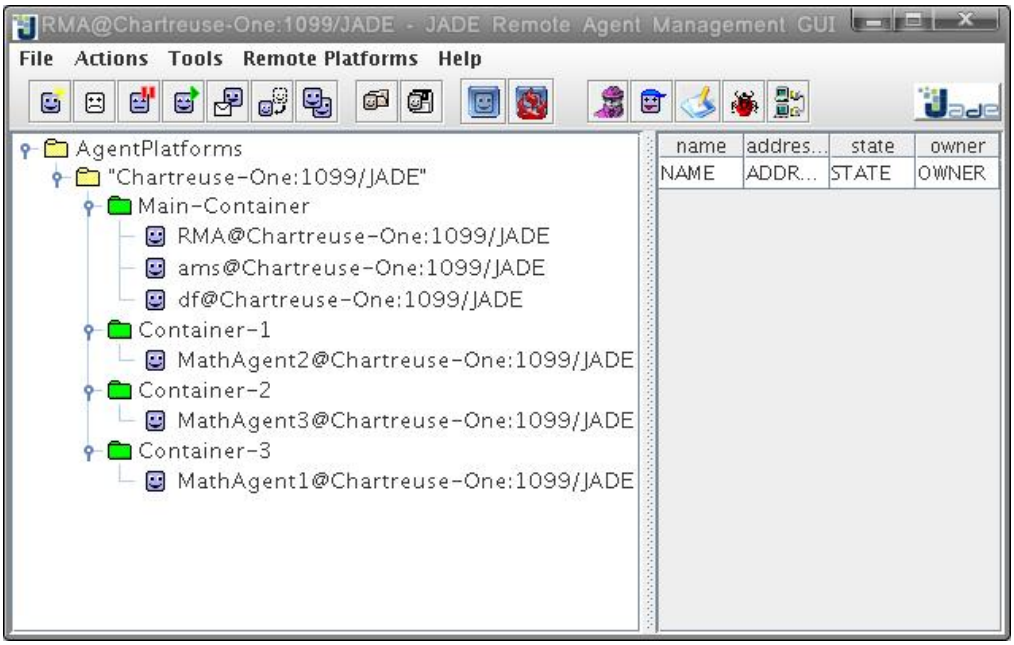
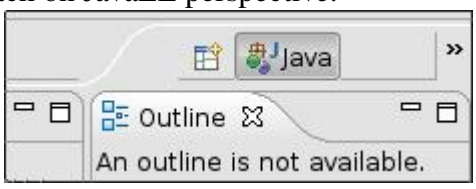


Illustration 2: Jade Platform GUI

Step 4 – Adding the serveur

Procedure

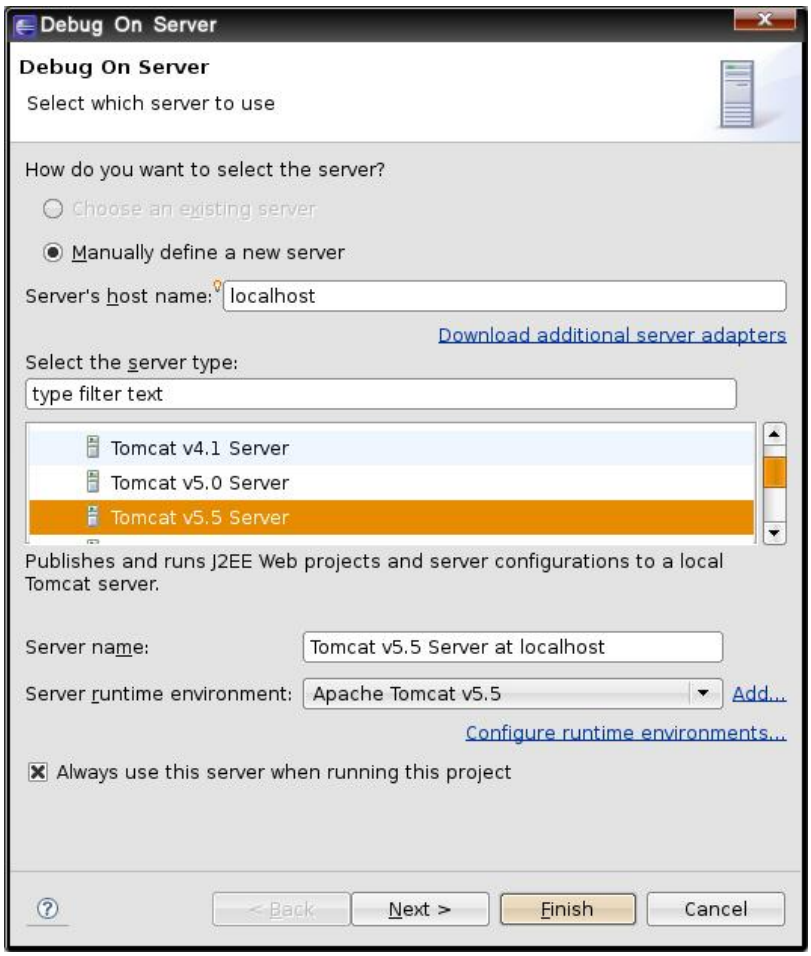
1. Switch on JavaEE perspective.



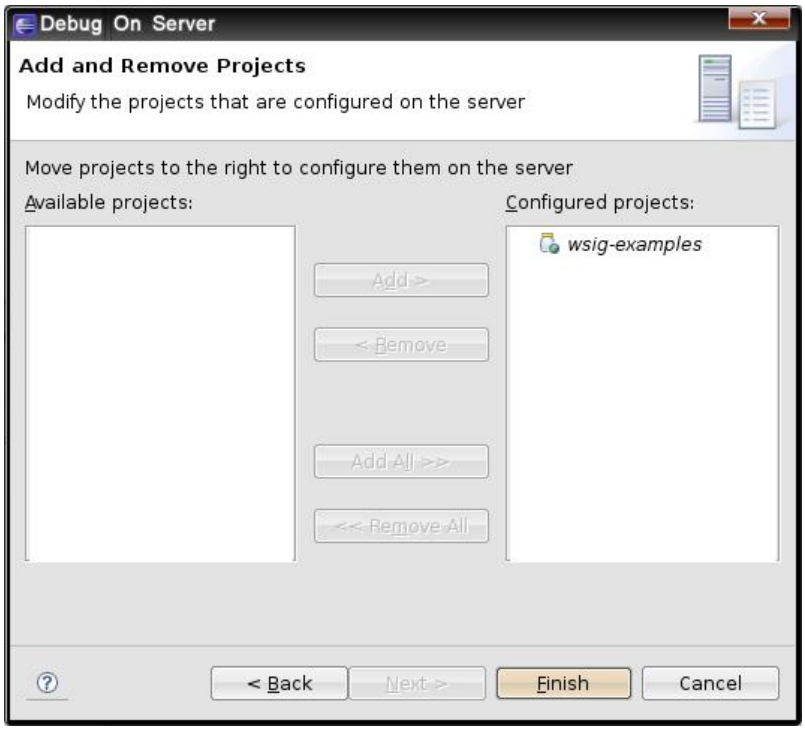
=>

2. Select the *wsig-examples* project in the projects list.

3. Create the application's *Debug Configuration* (*Debug* > *Debug As* > *Debug on server*) : choose *Manually define a new server* of type *Apache* > *Tomcat v5.5 Server*, tick *Always use this server when running this project*, and click on Next.

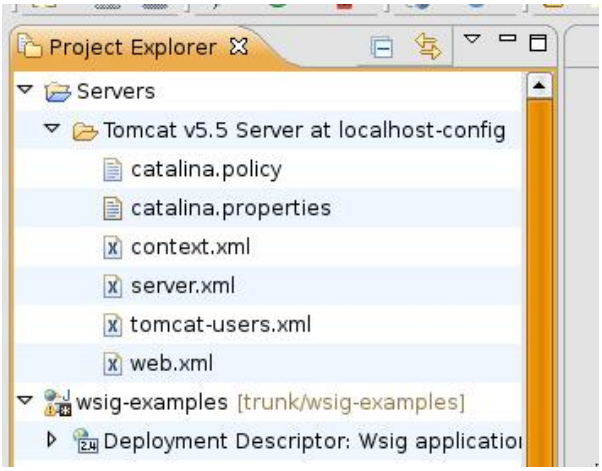


4. Fill in the installation path of Tomcat, */usr/share/tomcat5.5* on Ubuntu 7.10, and click on Next.
5. Add *wsig-examples* to the projects configured for this server, and finish.



Settings

In projects view, a new project appeared : *Servers*. It contains a directory with Tomcat setting files inside (the XML files).



To know the listened port by the server, open *server.xml*. This is my default configuration :

<!-- Define a non-SSL HTTP/1.1 Connector on port 8180 -->
<Connector acceptCount="100" connectionTimeout="20000" disableUploadTimeout="true" enableLookups="false" maxHttpHeaderSize="8192" maxSpareThreads="75" maxThreads="150" minSpareThreads="25" port="8180" redirectPort="8443"/>
...
<!-- Define an AJP 1.3 Connector on port 8009 -->
<Connector enableLookups="false" port="8009" protocol="AJP/1.3" redirectPort="8443"/>

If you change the ports, modifications will be synchronized immediately.

The wsig-examples application is declared at the end of the file :

<Context docBase="wsig-examples" path="/wsig-examples" reloadable="true" source="org.eclipse.jst.j2ee.server:wsig-examples"/></Host>

When *reloadable* is *true*, the modifications are directly propagated on the server, which is very useful to test source code on the fly. Without this mechanism, you would have to package the application in a war archive and deploy it on the server each time you want to test it.

Step 5 – Deployment and use

The server starts immediately after having been added, and deploys the application. The *ControlWSIG* agent is then created inside *WSIG-Container*. The WSIG console is available on http://localhost:8180/wsig-examples/ (depends on the port you set) with your favorite browser.

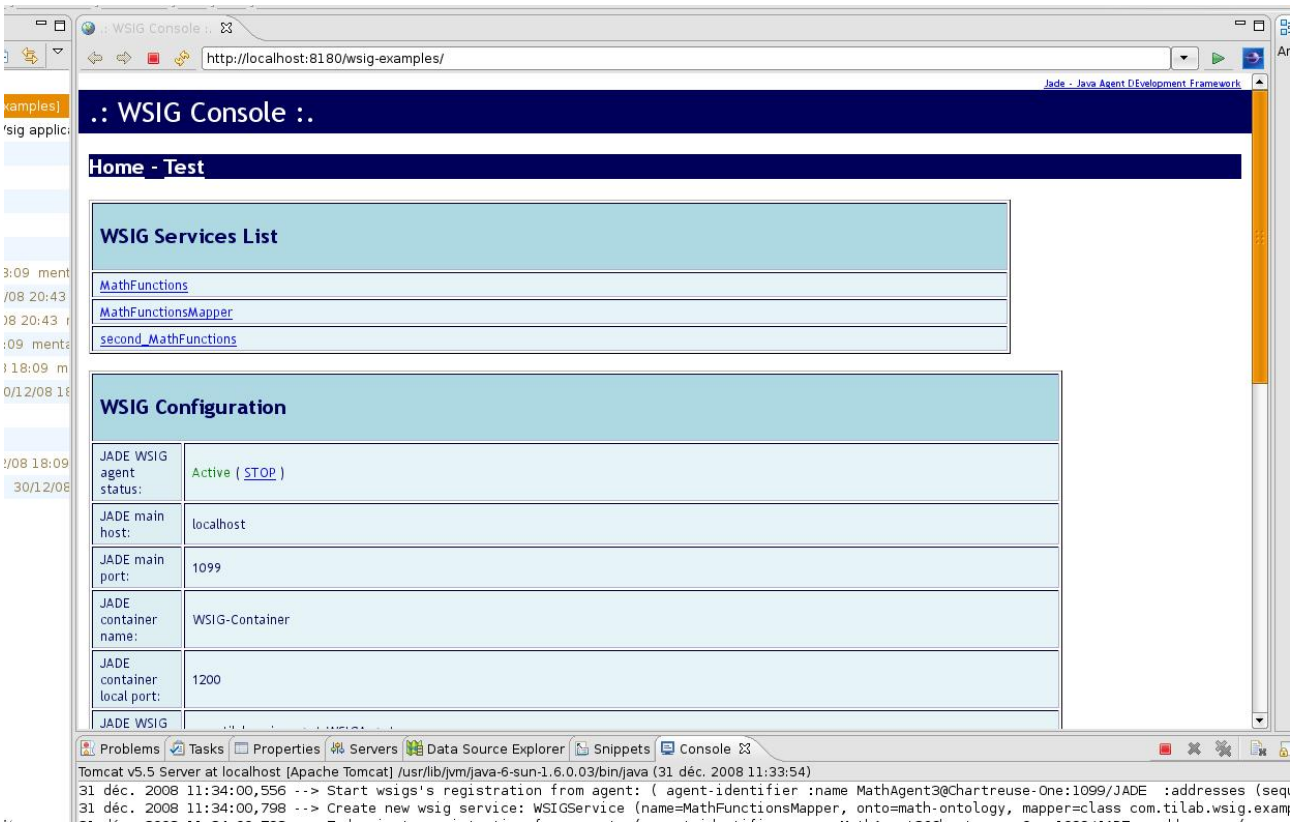
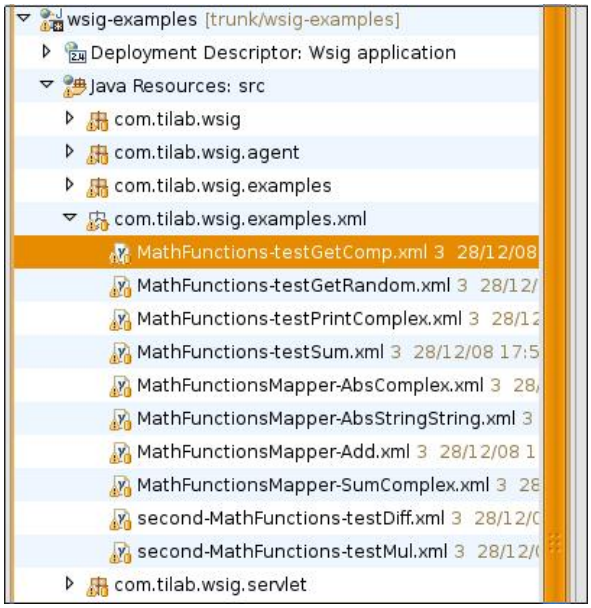


Illustration 3: WSIG's web console

Use

The web services corresponding to our agents are listed in *Home* page, in *WSIG Services List*. The sample SOAP requests are in *src/com.tilab.wsig.examples.xml :*



Go to the *Test* page to execute the sample requests proposed in this package : copy the XML content to test in *SOAP request*. Send the request, and that's it !

Note that :

- If the server replies that the web service doesn't exist ("*Service xxx not present in wsig*"), you should check that the corresponding agent is started.
- If you deactivate the *ControlWSIG* agent through the web console and reactivate it, the other agents will be stopped. Start them again (without stopping the server), and refresh the *Home* page to see them again in the list.

Illustration 4: SOAP request test

Manual launch

1. Launch the Jade platform
2. Launch the agents
3. In the Servers tab on the bottom of the screen, right click on the server name and click on *Debug*, or directly launch wsig-examples through the *Run* > *Debug* menu or the toolbar.

Stop

To stop everything properly :

1. Right click on the server name and click on *Stop*.
2. Then stop Jade through the GUI by clicking on *File* > *Shut down Agent Platform*.

For more details on Jade and WSIG, please refer to their guides.