

## Manuel d'installation

### Prérequis:

- Eclipse avec java 1,7
- Un serveur web avec apache 2, php 5 et mysql

### Etape 1:

Prendre le projet github dans votre dossier de destination

### Etape 1 bis :

Copier le contenu du dossier "SMAAlpha\_html" a la racine de votre serveur web.

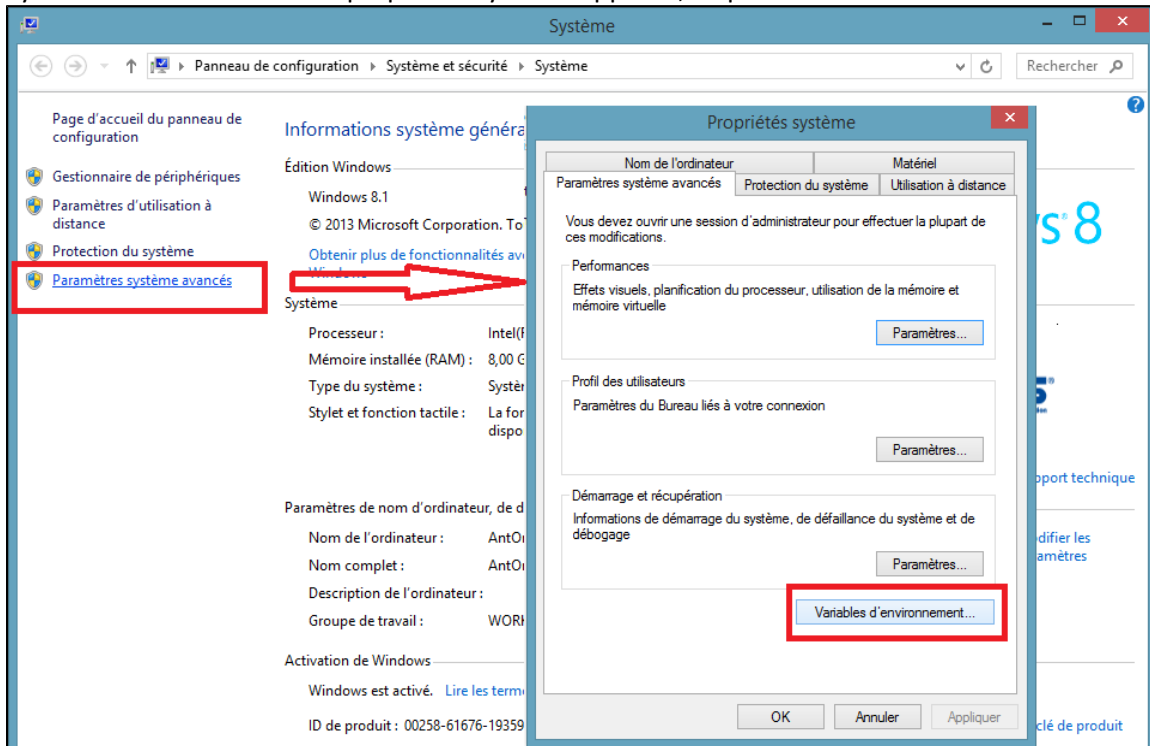
Importer la base de donnée via le fichier c3po.sql qui se situe à la racine du dossier "SMAAlpha\_html".

Modifier le fichier SMAAlpha\_html\model\sql.php et renseigner les nom de compte et mot de passe valides.

### Etape 2:

Mettre à jour ou créer la variable classpath en faisant comme suit :

Faire un clic droit sur le poste de travail/ordinateur , choisissez propriétés, puis Paramètres Système avancés. La fenêtre propriétés système apparaît, cliquez sur variables d'environnement.



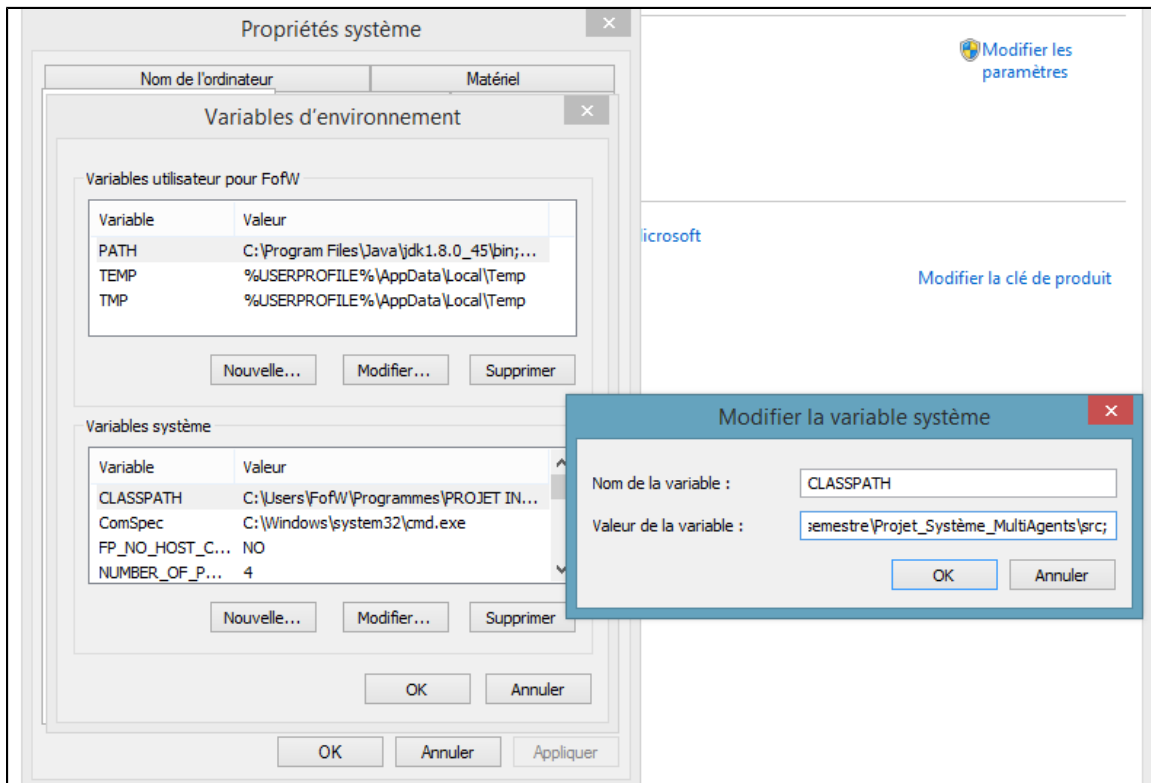
Une petite fenêtre intitulée " variables d'environnement " apparaît.

Dans la zone variables système, essayer de trouver la variable d'environnement qui porte le nom CLASSPATH. Si vous ne la trouvez pas, il faut la créer.

Maintenant que la variable est trouvée/créée ,on doit lui attribuer une valeur. Cette valeur est la concaténation des chemins:

```
[chemin_du_projet]\SMAAlpha\Source_Externes\Jade\lib\commons-codec\ commons-codec-1.3.jar;[chemin_du_projet]\SMAAlpha\Source_Externes\Jade\lib\jade.jar;
```

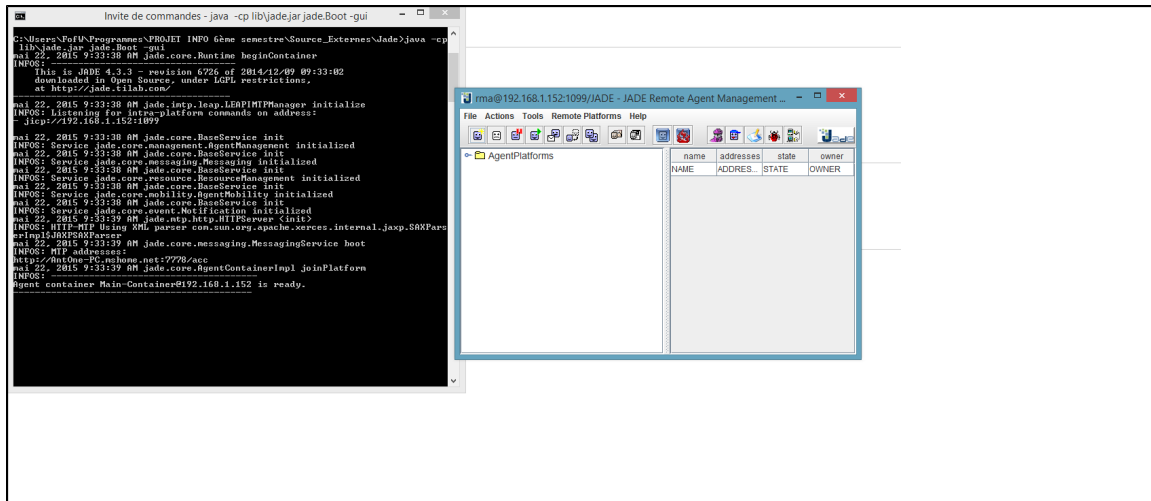
Séparez les chemins par des ; et Sauvegardez les modifications.



Pour vérifier que l'opération est bien réalisée, ouvrir l'invite de commande, se déplacer vers le dossier (commande « cd .. » pour aller à la racine du repertoire actuel, et cd nomdedossier pour aller dans le dossier présent dans le dossier actuel) puis entrez la commande

```
java -cp lib\jade.jar jade.Boot -gui
```

La fenêtre suivante apparaît :



fermez la fenetre avec file/shut down Agent Platform

### Etape 3:

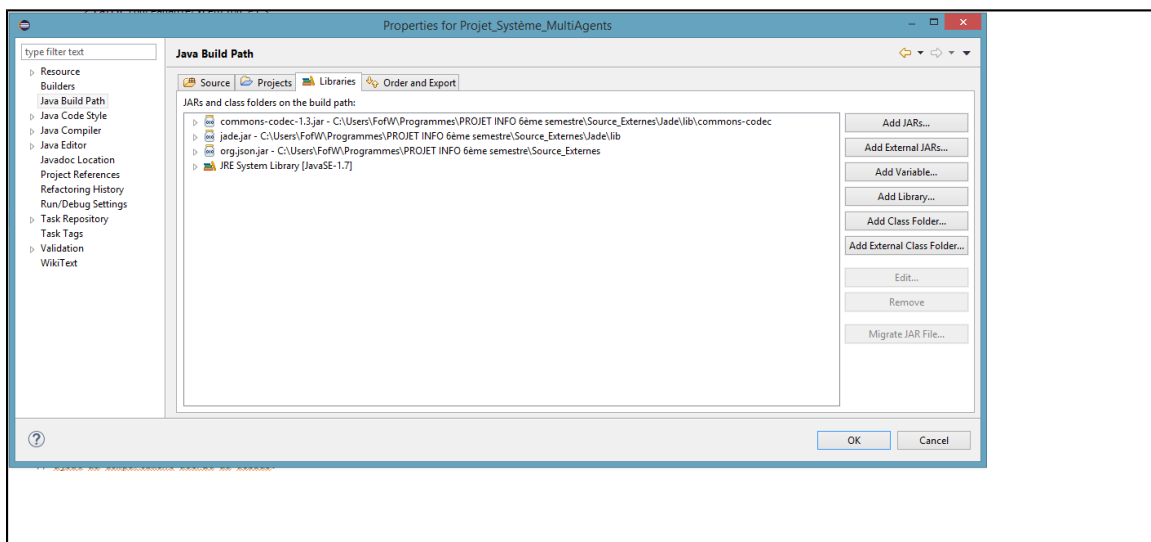
Ouvrir le systeme multi agents avec eclipse.

Ouvrir eclipse, importer le projet et ensuite, faire un clic droit sur le dossier du projet, puis cliquer sur propriétés, java build path, librairies, Add External Jars. Dans le dossier sources externes, importer :

[chemin\_du\_projet]\SMAAlpha\Source\_Externes\Jade\lib\jade.jar

[chemin\_du\_projet]\SMAAlpha\Source\_Externes\org.json.jar

[chemin\_du\_projet]\SMAAlpha\Source\_Externes\Jade\lib\commons-codec\ commons-codec-1.3.jar



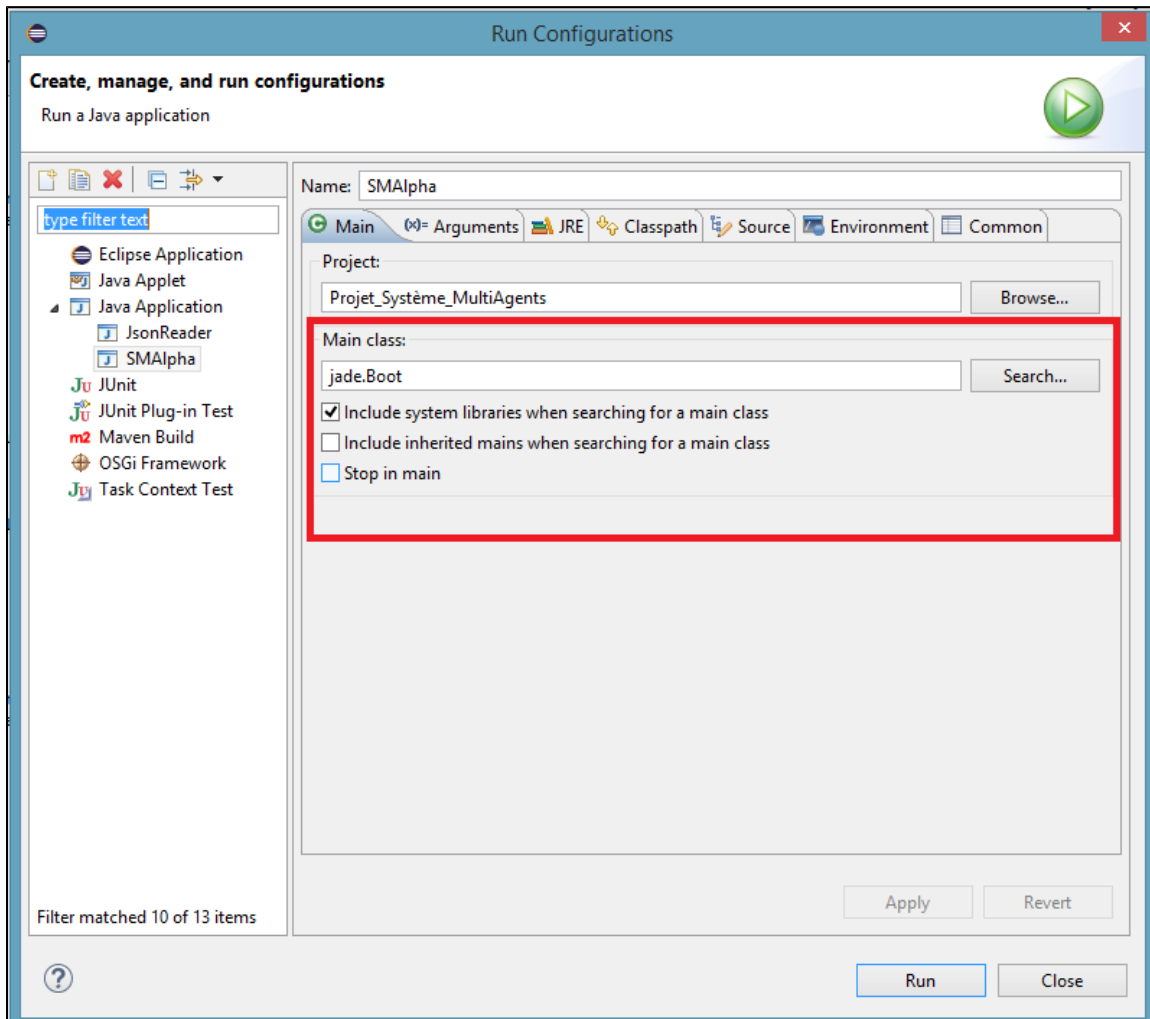
Les erreurs de projet devraient disparaître avec l'import des fonctions dont elles ont besoin, et les erreurs de package lors de la première sauvegarde avant lancement.

#### Etape 4:

Lancer le systeme multi agents avec Eclipse

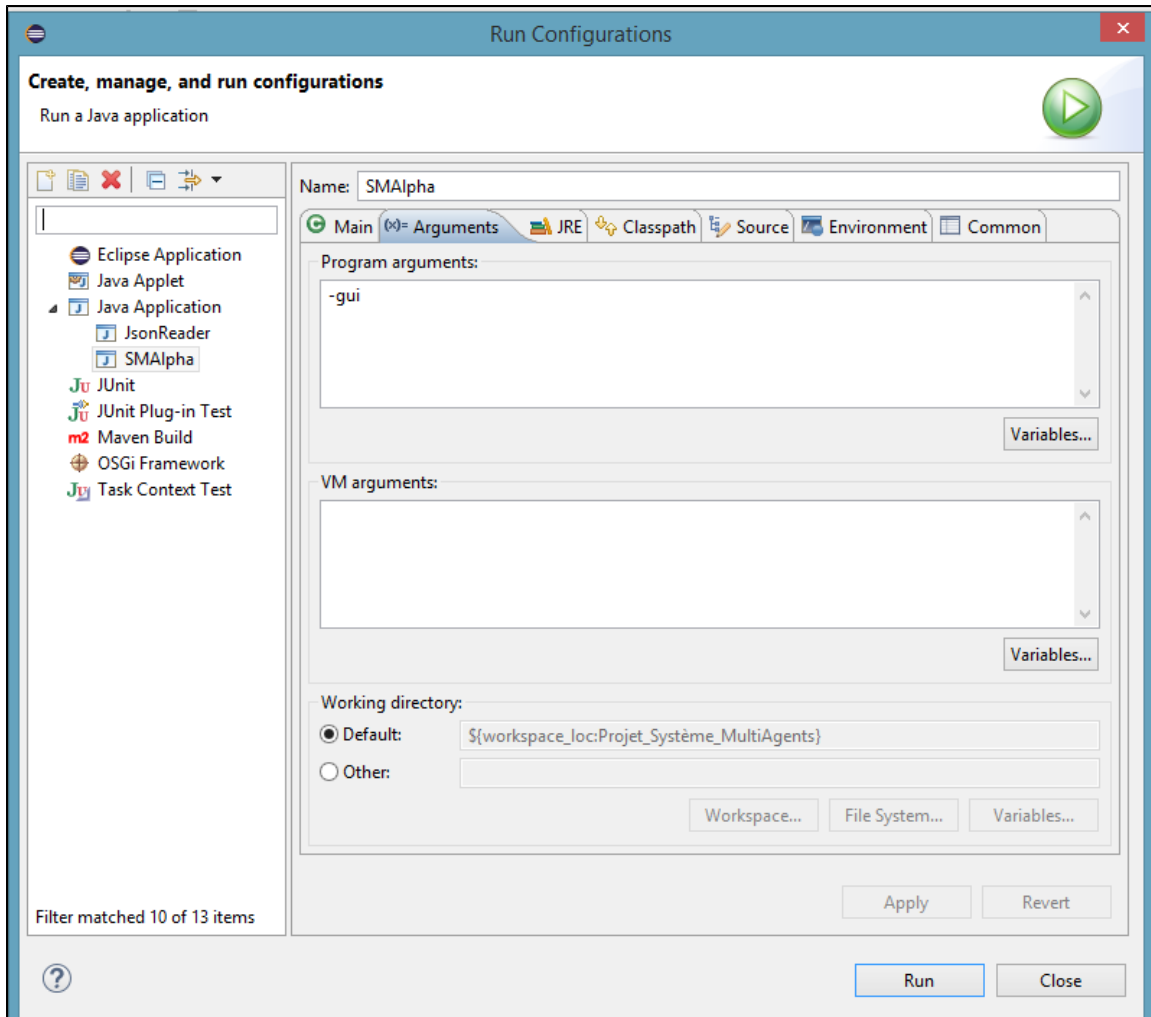
Sous Eclipse, déplier l'onglet run et cliquez sur "Run Configuration".

Double cliquer sur Java Application. Dans l'onglet Main (normalement actif), marquer "jade.Boot" dans la zone de saisie Main class et cocher la case " Include librairies when searching for a main class".



Puis, dans l'onglet argument, tapez le code suivant:

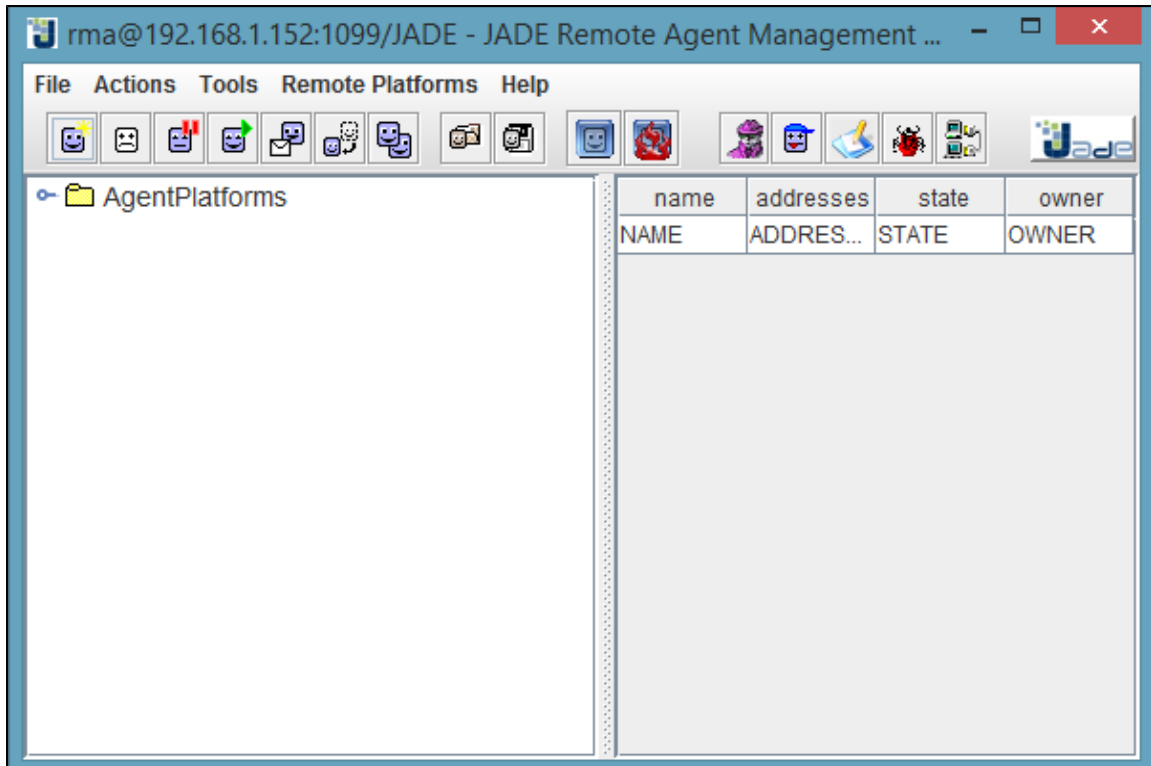
-gui



Jade est prête à être lancée, vous pouvez cliquer sur Run, Jade se lance en quelques secondes.

Lancer le Système Multi-Agent :

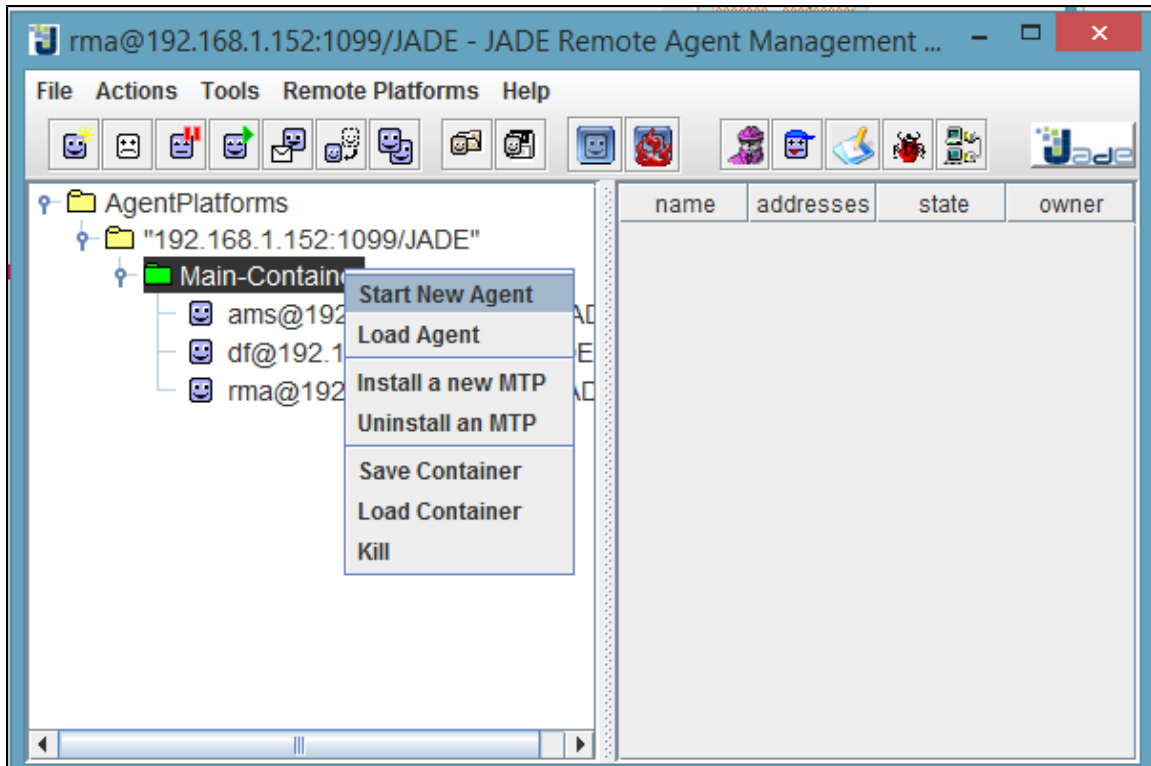
une fois Jade lancée, la fenêtre suivante apparait:



( Si vous rencontrez l'erreur **Communication failure while joining agent platform: No ICP active**, c'est qu'une instance de Jade est encore active. Auquel cas, la fermer avec File > Shut down Agent Plateform. Dans le cas ou la fenêtre aurait été fermé d'une autre manière, il se peut que le processus soit encore actif, et qu'il faille le terminer avec le gestionnaire des tâches (Java(TM) Platform SE binary) )

Derouler "AgentPlatforms=>[IP\_PRIVÉE]:1099/JADE"

Clique droit sur "Main-Container" puis cliquer sur "Start New Agent"



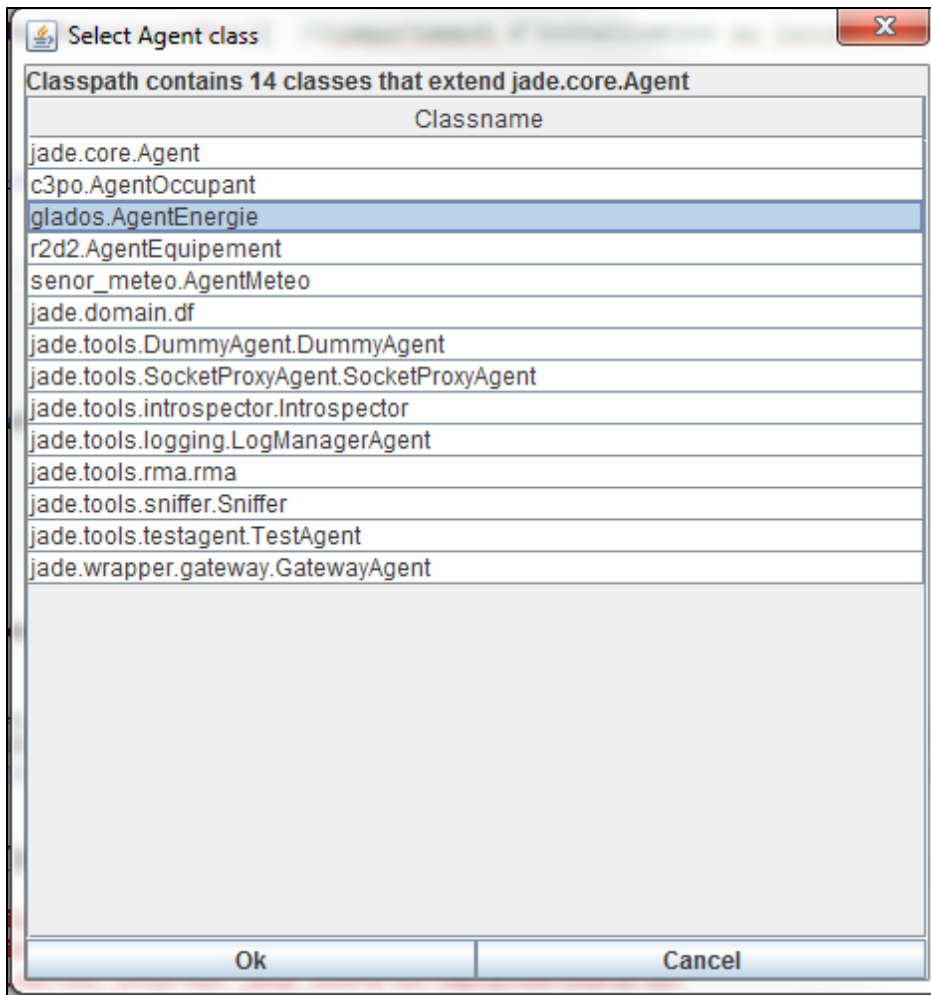
Agent Name: glados

classe Name: cliquer sur "Search in classpath for classes extending Agent"

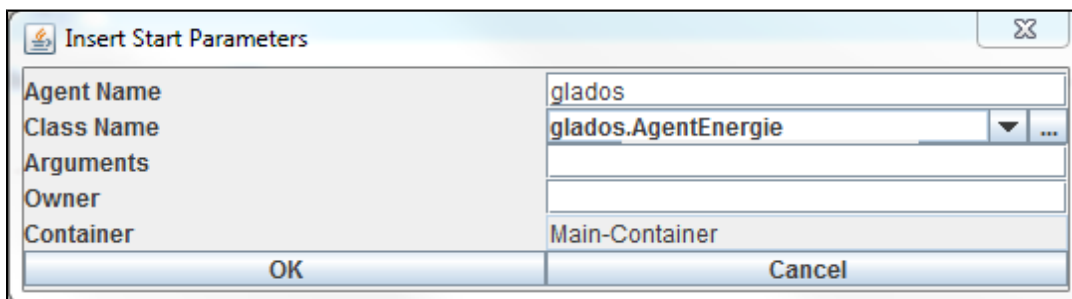
(veillez à bien avoir lancé le serveur, et configurer le fichier

"...\SMAAlpha\Projet\_Système\_MultiAgents\serveur.txt" avec son adresse auparavant.)

selectionner "glados.AgentEnergie" puis "Ok"



cliquer sur OK à nouveau



L'Agent Energie (Glados) est maintenant lancé(e), se rendant compte que les autres agents ne le sont pas, il (ou elle) va les lancer.  
Le système est maintenant opérationnel.