**CENTRE UNIVERSITAIRE DE MILA**

**INSTITUT DES SCIENCES ET DE LA TECHNOLOGIE**

**2ere Année Master STIC et I2A**

**Module: SMA (Résolution distribuée des prolèmes**

**TP1**

**Créez votre premier agent avec JADE et ECLIPSE**

**Introduction**

Jade est un middleware qui facilite le développement des systèmes multi agents (SMA). JADE contient :

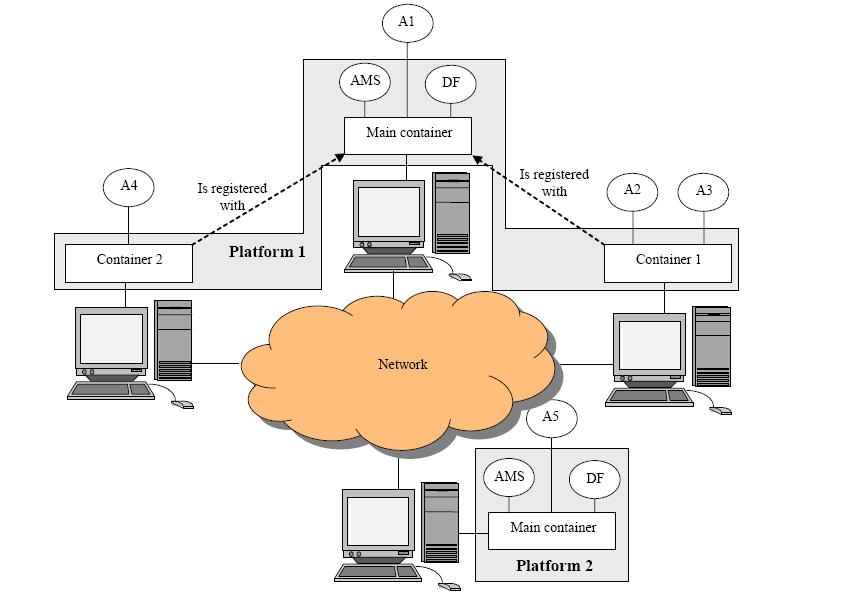
• Un runtime Environnement : l'environnement ou les agents peuvent vivre. Ce runtime environnement doit être activé pour pouvoir lancer les agents.

• Une librairie de classes : que les développeurs utilisent pour écrire leurs agents

• Une suite d'outils graphiques : qui facilitent la gestion et la supervision de la plateforme des agents

Chaque instance du JADE est appelée conteneur " container ", et peut contenir plusieurs agents. Un ensemble de conteneurs constituent une plateforme. Chaque plateforme doit contenir un conteneur spécial appelé main-container et tous les autres conteneurs s'enregistrent auprès de celui-là dés leur lancement.

La figure suivante illustre les concepts de base du jade en montrant un petit exemple de deux plateformes jade composées respectivement de trois et un conteneur Chaque agent est identifié par un identifiant unique et peut communiquer avec n'importe quel autre agent sans avoir besoin de connaître son emplacement :



* Dans le même conteneur (exemple agents A2 et A3)
* Dans la même plateforme mais dans des conteneurs différant (ex A1 et A2)
* Dans deux plateformes différentes (ex. A4 and A5).

Un main-container se distingue des autres " simples " conteneurs par une autre chose; il contient toujours deux agents spéciaux appelés AMS et DF qui sont lancés automatiquement au lancement du main-container

1. AMS (Agent Management System) qui fournit le service de nommage (pour assurer par exemple que chaque agent possède un identifiant unique dans la plateforme) et qui représente l'autorité de la plateforme (par exemple il est possible de créer/arrêter des agents en envoyant des requêtes à l'AMS)
2. DF (Directory Facilitator) qui fournit un système de pages jaunes qui permet aux agents de retrouver lesnagents fournisseurs de services.

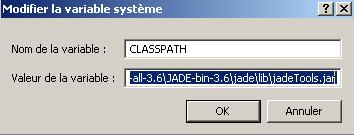
**Installation**

Voici les étapes à suivre pour installer JADE :

1. Téléchargez le fichier JADE-all-3.6.zip de l'adresse suivante <http://jade.tilab.com/download.php> ou bien une version plus récente.
2. Décompressez le fichier (on va supposer tout au long de ce tutorial que le chemin du répertoire JADE-all-3.6 est le c:\ JADE-all-3.6 ). Apres avoir décompressé le fichier vous retrouvez quatre autres fichiers ZIP (JADEbin-3.6.zip , JADE-doc-3.6.zip, JADE-examples-3.6.zip, JADE-src-3.6.zip). Décompressez ces 4 fichiers
3. On doit maintenant mettre à jour la variable classpath (si elle n'existe pas encore il faut la créer) En faisant comme suit :

Par Clic droit sur le poste de travail, choisissez propriétés. La fenêtre propriétés système apparaît, choisissez l'onglet Avancé Puis cliquez sur variables d'environnement Une petite fenêtre intitulée " variables d'environnement " apparaît

Dans la zone variable système, essayez 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 des quatre fichiers jar http.jar, iiop.jar, jade.jar, jadeTools.jar situés dans le chemin c:\JADE-all-3.6\JADE-bin-3.6\jade\lib



**Création du premier agent avec JADE et ECLIPSE**

Maintenant, nous allons créer notre premier agent. Nous allons utiliser l'exemple fourni avec la plateforme situé dans : C:\JADE-all-3.6\JADE-examples-3.6\jade\src\examples\hello.

* Ouvrez eclipse et créez un nouveau projet (MyFirstAgent par exemple).
* Ajoutez un package (firstAgent)
* créez une nouvelle classe appelée HelloWorldAgent.
* Collez le code suivant dans la classe.

package firstAgent;

import jade.core.Agent;

/\*\*

This example shows a minimal agent that just prints "Hello World!"

and then terminates.

\*/

public class HelloWorldAgent extends Agent {

protected void setup() {

System.out.println("Hello World! My name is "+getLocalName());

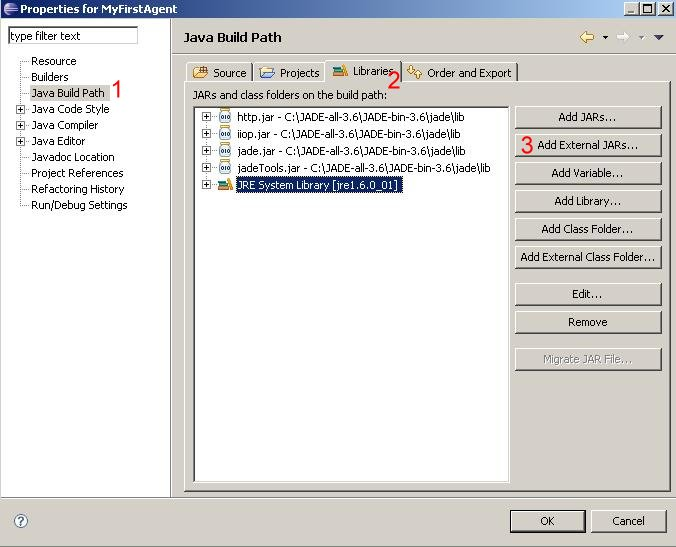
// Make this agent terminate

doDelete();

}

}

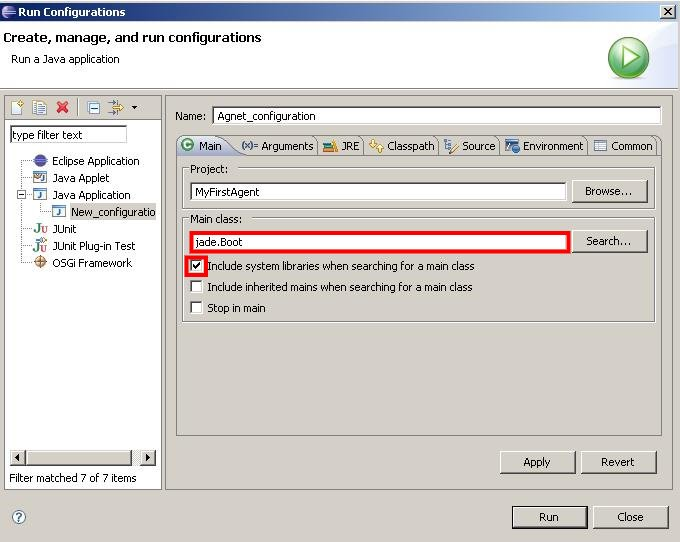
* Vous remarquez l'existence de plusieurs erreurs dans ce petit code.
* Pour résoudre ce petit problème, effectuez un clic droit sur le nom du projet Puis choisissez propriétés. Cliquez sur java build path >> Libraries>> add external JARs



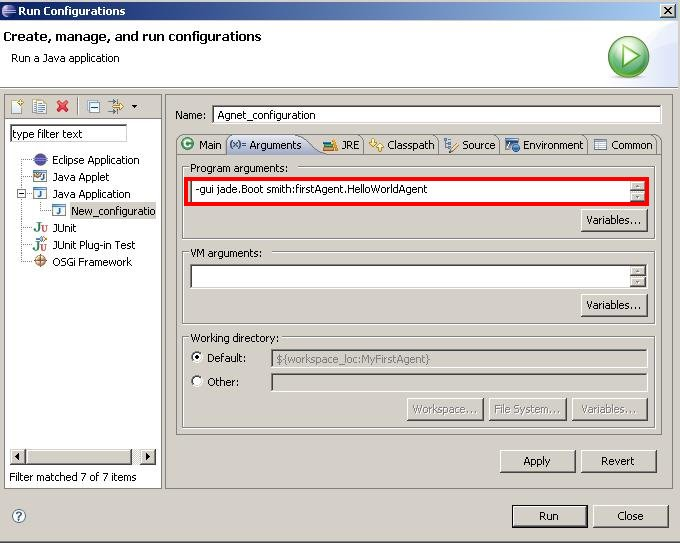
* Ajoutez les quatre jar "http.jar" "iiop.jar" "jade.jar" "jadeTools.jar" situés dans C:\JADE-all-3.6\JADE-bin-3.6\jade\lib
* puis cliquez sur Ok

**Compilation et exécution de l'agent**

Il reste à compiler et lancer l'agent pour cela : Allez dans run>>Run configuration puis double-cliquez sur java application Dans l'onglet " main " et dans la zone de saisie Main class, tapez le code suivant : jade.Boot puis cochezla case : " Include librairies when searching for a main class



Puis cliquez sur apply pour ne pas refaire cette configuration plusieurs fois dans le même projet



Cliquez sur run pour voir le résultat (affichage de Hello World!)