

# An Introduction to Software Agents

## Setup guide

### 1. Starting containers

Before you start developing your agents you should set up a few containers to use for testing. You can either run all containers on your local computer, or you can work together with other students to create a local agent network.

Together with the files for this workshop you will find two packages; main-container.zip and container.zip. These are pre-configured agent containers which you can simply extract to separate folders and start up. Remember that the main-container has to be launched first as this is the connection point for all other containers!

#### 1.1. Connect to a main container on another computer

If you want to work together with other students and create an agent platform across several computers you need to slightly modify the configuration of the container.zip package.

Please follow these steps:

- a) Open the conf/jade.properties file
- b) Change the “host=localhost” line to “host=<address of the main container>”.
- c) Save the file and launch your container.

### 2. Creating a new agent project

For the exercises I have created a base project which you may use to start your agent development. This base project is available in the agent-projectbase.zip package. Unzip this package to your Eclipse workspace and use the Import existing projects into workspace feature. After the project has been successfully imported you need to add a few libraries to the classpath. These libraries are available in the lib/ folder. To add these to your classpath, follow these steps:

- a) Right click on your project in the Package Explorer in Eclipse and choose Properties.
- b) Go to the “Java Build Path” category and the “Libraries” tab.
- c) Click the “Add JARs” button and browse to the lib folder.
- d) Select all the JAR files and click OK.
- e) Click OK on the project properties window.

You may now start developing your agents.

The `MyAgent` and `MyBehaviour` classes show a very basic agent and behaviour implementation and is meant to be a small guideline for getting you started.

For the exercises there are a few utility classes that you may use. These are located in the `no.uia.ikt407.exercises.utils` package.

### 3. Configuring your project so that the agents are launched when you run the application

You run your agent container (containing your agent) by running the `ContainerLauncher` class as a “Java Application” from Eclipse. The base project also contains configuration in the same way the main-container and container packages do and are configured accordingly.

In order for the container to find your agent and start it you need to make some changes to the Jade4Spring configuration located in `conf/jade4spring.xml`. The base project contains some example configuration for the `MyAgent` class and you may replace this with configuration for your own agent.

#### 3.1. Define the agent bean

First you have to define your agent bean – your agent instance. This is done by the following line to the `jade4spring.xml` file:

```
<bean id="myAgent" class="no.uia.ikt407.exercises.MyAgent" />
```

Where the `id` parameter is a unique identifier and the `class` parameter is the full name of your agent class. This line will make sure your agent is instantiated.

#### 3.2. Make Jade4Spring launch your agent when you start your container

The previous section only instantiated your agent. Now we need to tell Jade4Spring to add it to the container and run it, so that it will actually do something useful.

To add your agent to the list of agents that Jade4Spring should launch, add the following element to the `<property name="autostartAgents">` part of the `jadeContainer` bean configuration:

```
<entry key="MyAgent">
  <ref bean="myAgent" />
</entry>
```