



Universidade do Minho

Escola de Engenharia

Departamento de Informática

**Mestrado Integrado em Engenharia Informática
Mestrado em Engenharia Informática
Agentes Inteligentes
2019/2020**

Paulo Novais, Filipe Gonçalves

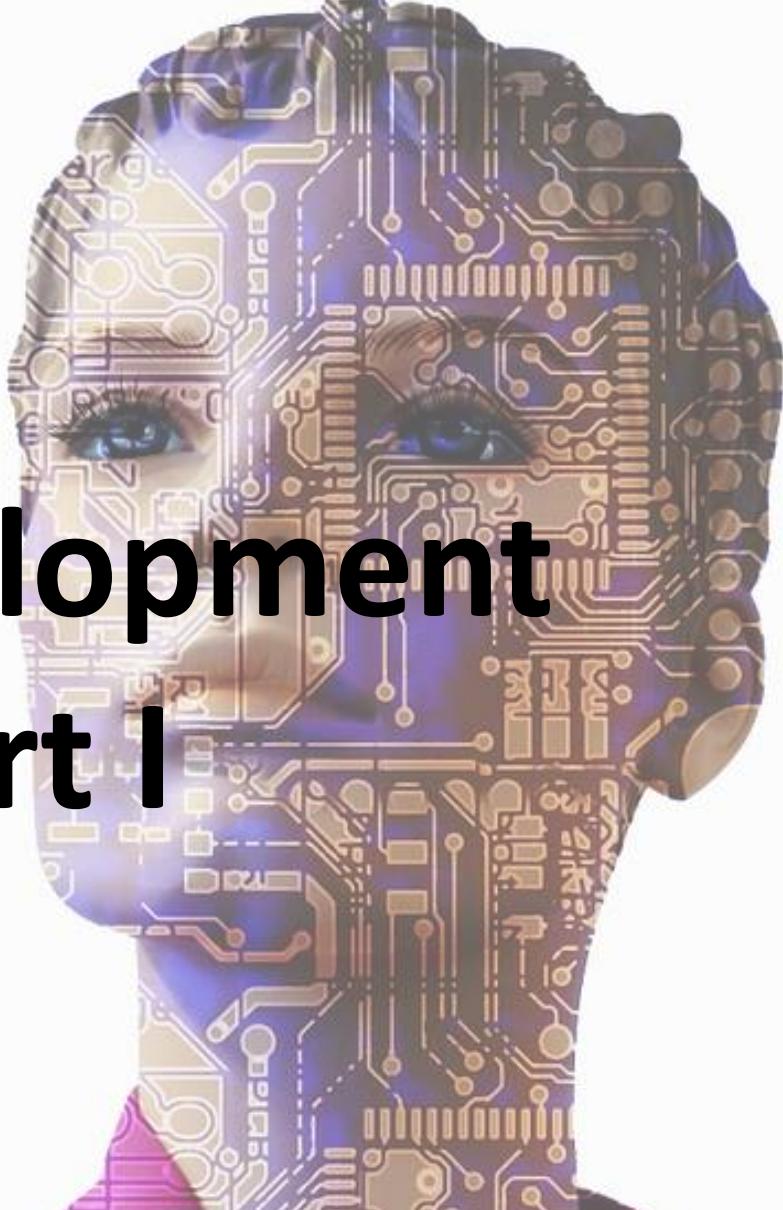
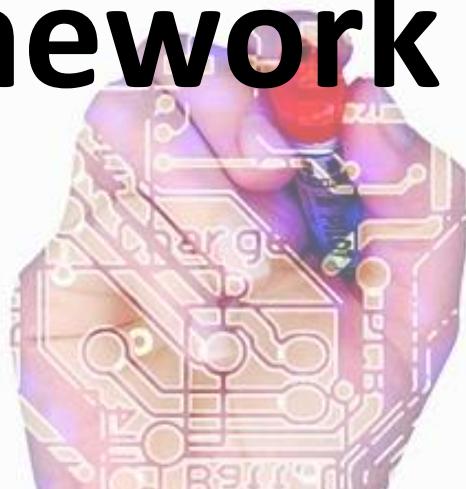
- Paulo Novais – pjon@di.uminho.pt
- Filipe Gonçalves – fgoncalves@algoritmi.uminho.pt

- Departamento de Informática
Escola de Engenharia
Universidade do Minho
- ISLab – (Synthetic Intelligence Lab)
- Centro ALGORITMI
Universidade do Minho

JADE

Java Agent DEvelopment

Framework – Part I



JADE Features

- Agent platform with the purpose of simplifying the development of agents
 - Agent lifetime management
 - Message transport management
 - Message structure
 - Protocol Interaction
 - Ontologies
 - Safety
- API/Interface
 - Execution environment
 - Provides a class library for multi-agent development
 - Graphical administration and management tools for agents
- Agent Structure/Standard

Containers

- Each instance of the execution environment is a Container
- Each Container can have zero or more agents running simultaneously
- Main-Container is required to support the platform
- All other Containers have to register on it when they are created
- Containers can be remote

Containers

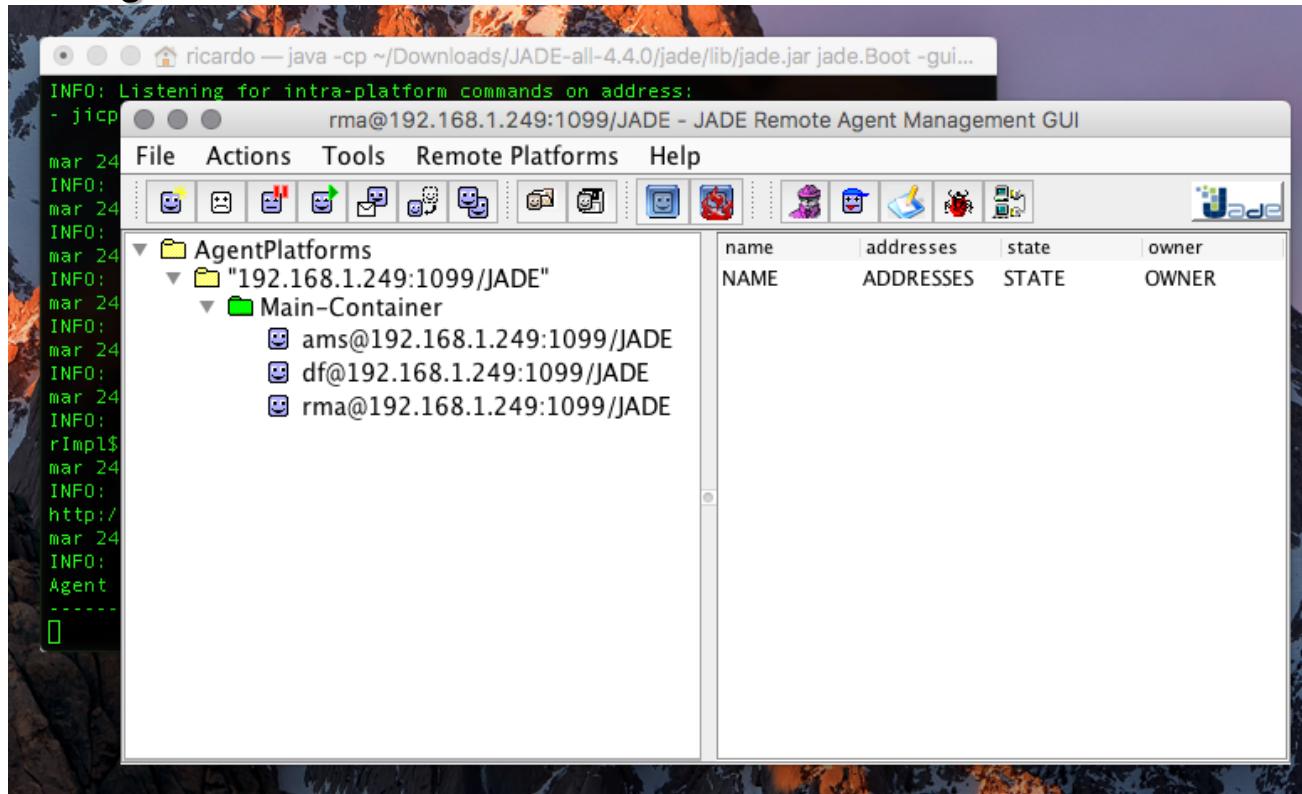
- Main-Container:

- Agent Management System (AMS): make sure there are no name collisions and lets you start and end agents.
- Directory Facilitator (DF): provides a yellow pages services through which an agent can search for other agents.

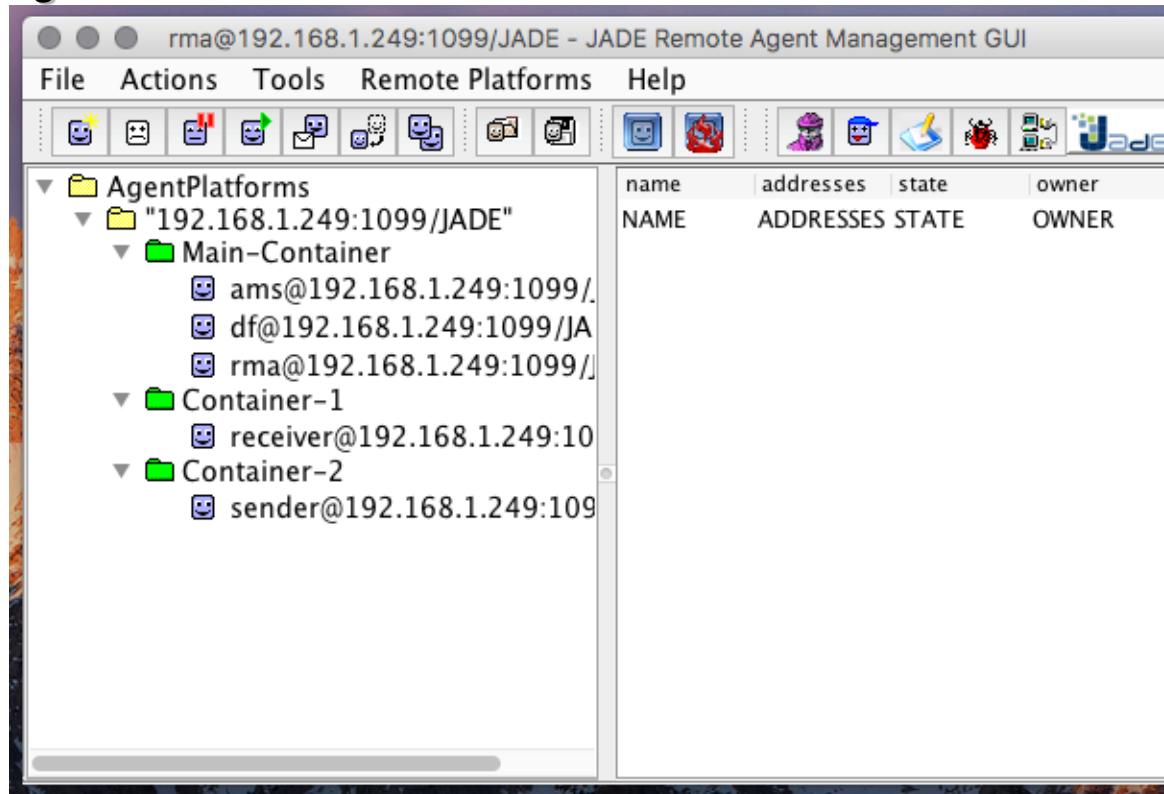
Optionally:

- Remote Monitoring Agent (RMA): allows to manage the platform through a GUI.

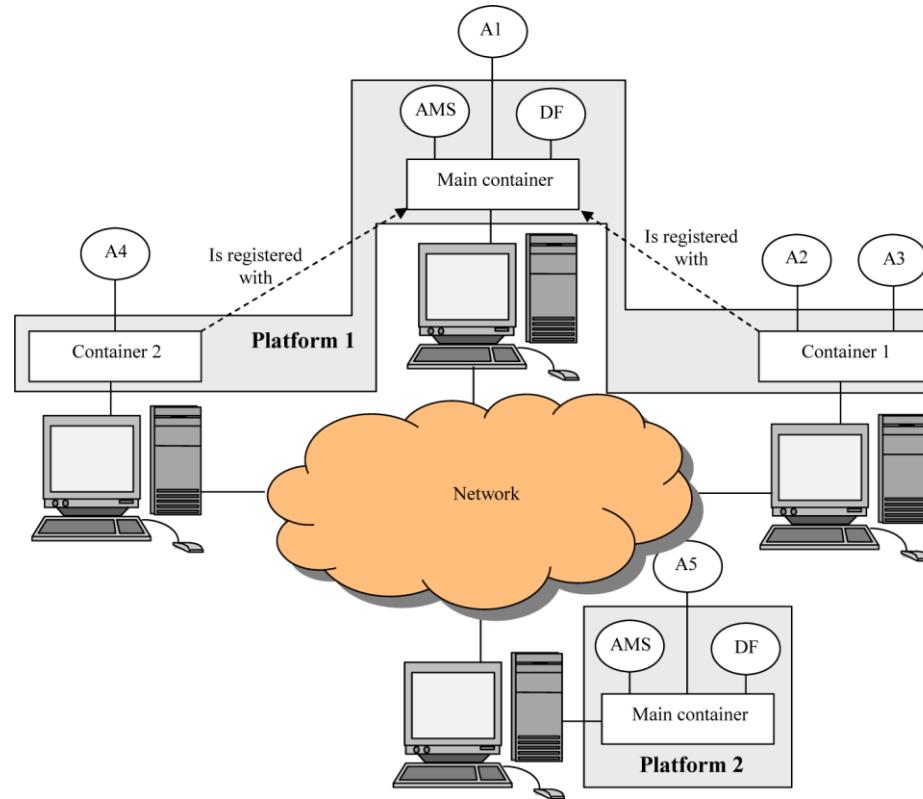
GUI without agents



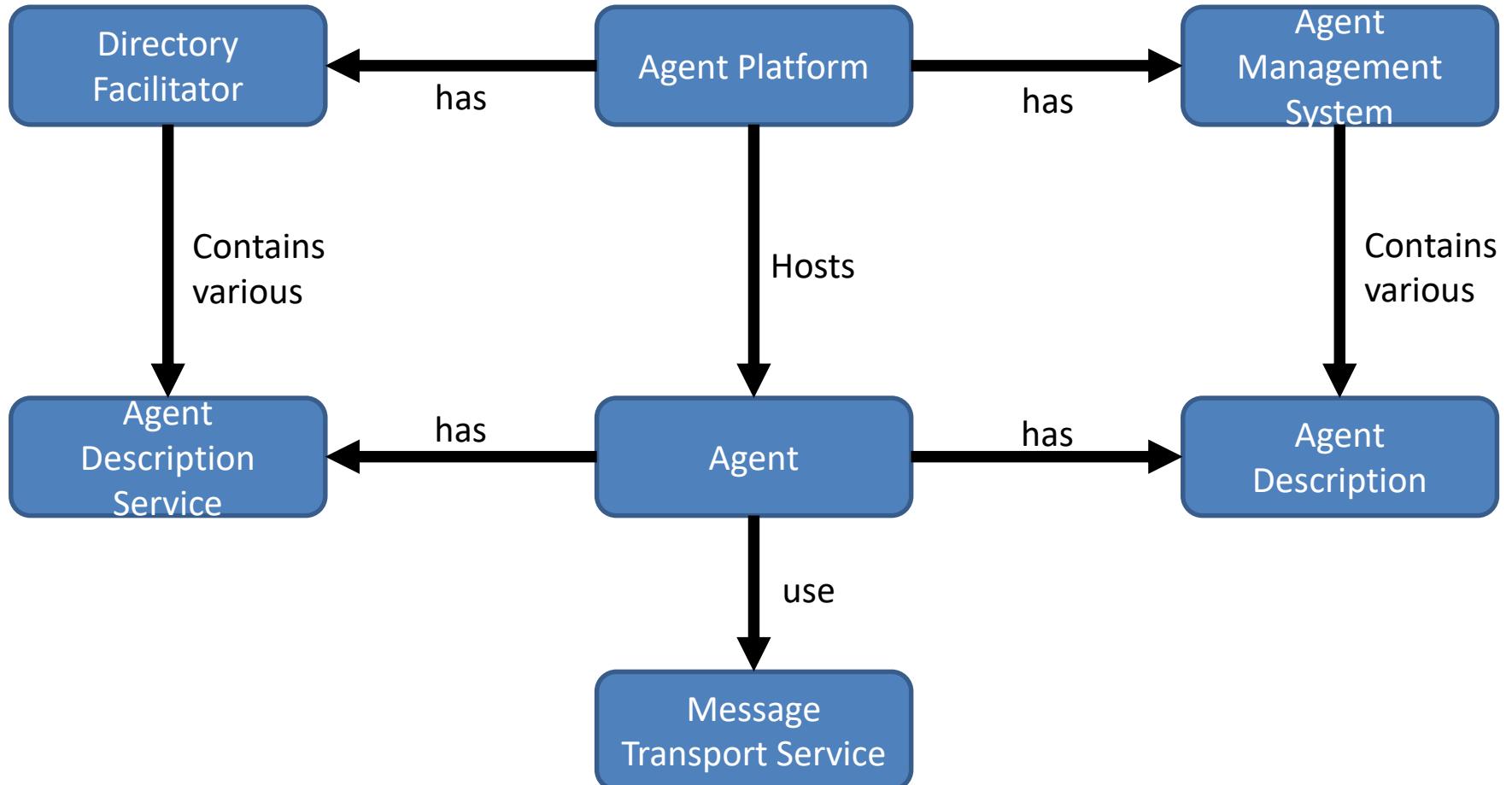
GUI with 2 agents



Platform - Set of distributed containers



Agent UML Diagram



Performative-FIPA compliant

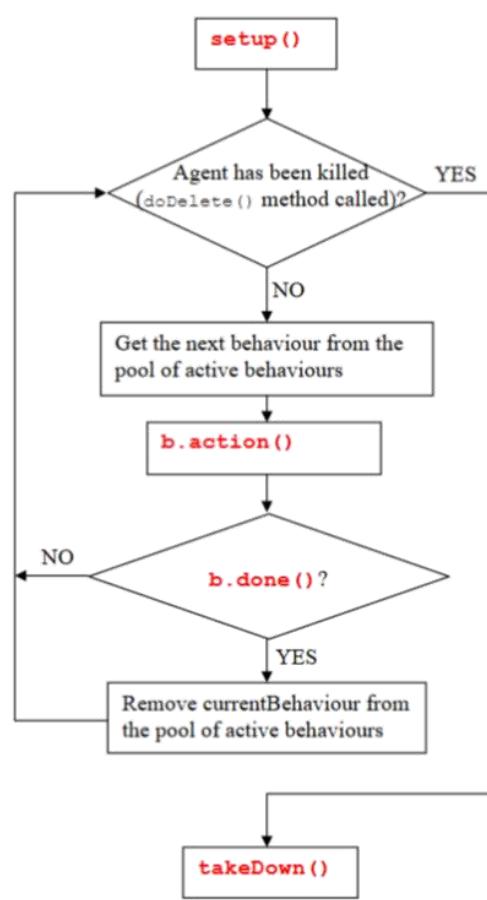
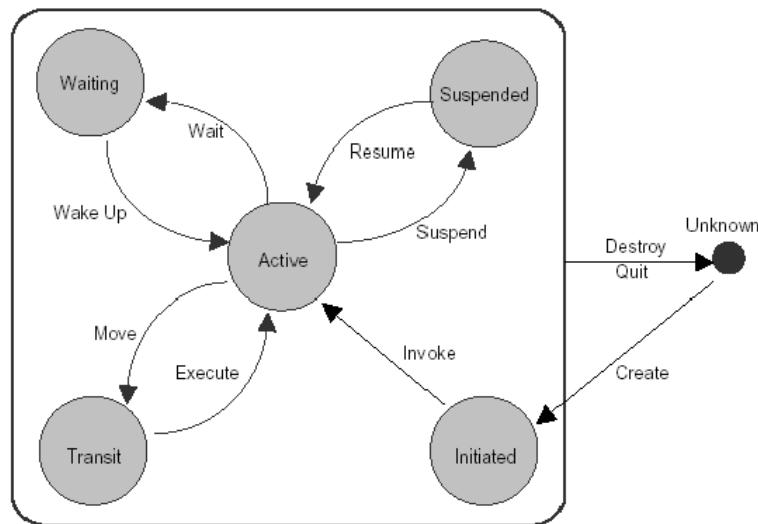
- FIPA - Foundation for Intelligent Physical Agents.
- Promotes standards for agent development.
- Delivers Messages Representation, Agents Structure, Messages Structure, etc.
- FIPA Message Types: ASK, INFORM, REQUEST, PROPOSE, REFUSE, etc.

Agent Behaviours

- Primitive:
 - SimpleBehaviour
 - CyclicBehaviour
 - TickerBehaviour
 - OneShotBehaviour
 - WakerBehaviour
 - SimpleAchieveREInitiator
 - SimpleAchieveREResponder
 - FSMBehaviour
- Complex:
 - ParallelBehaviour
 - SequentialBehaviour

Agentes Inteligentes @ 2019/2020

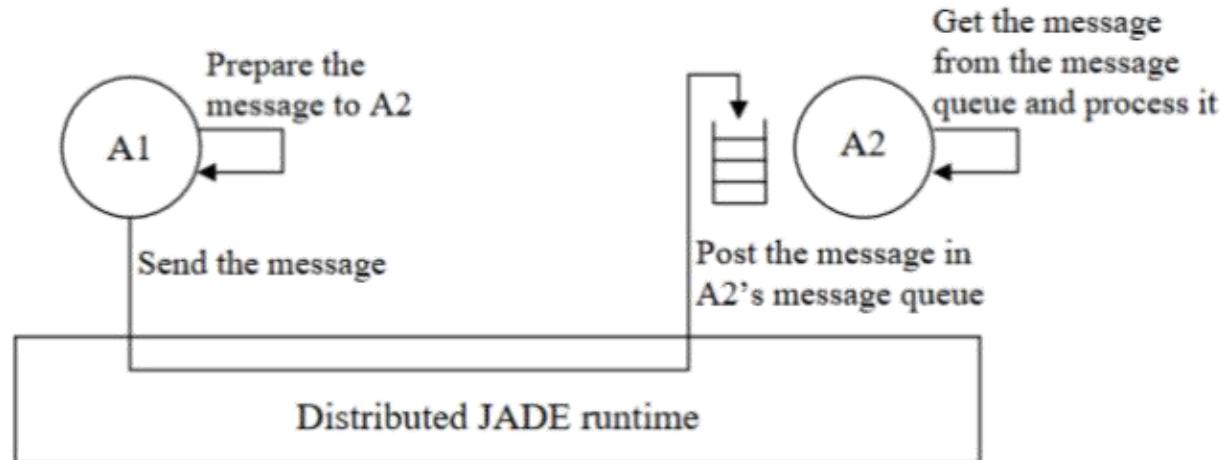
Agent Life Cycle



- Initializations
- Addition of initial behaviours
- Highlighted in red the methods that programmers have to implement*
- Agent “life” (execution of behaviours)
- Clean-up operations

Message Transfer Procedure

- Asynchronous message transfer
- May have more than one receptor

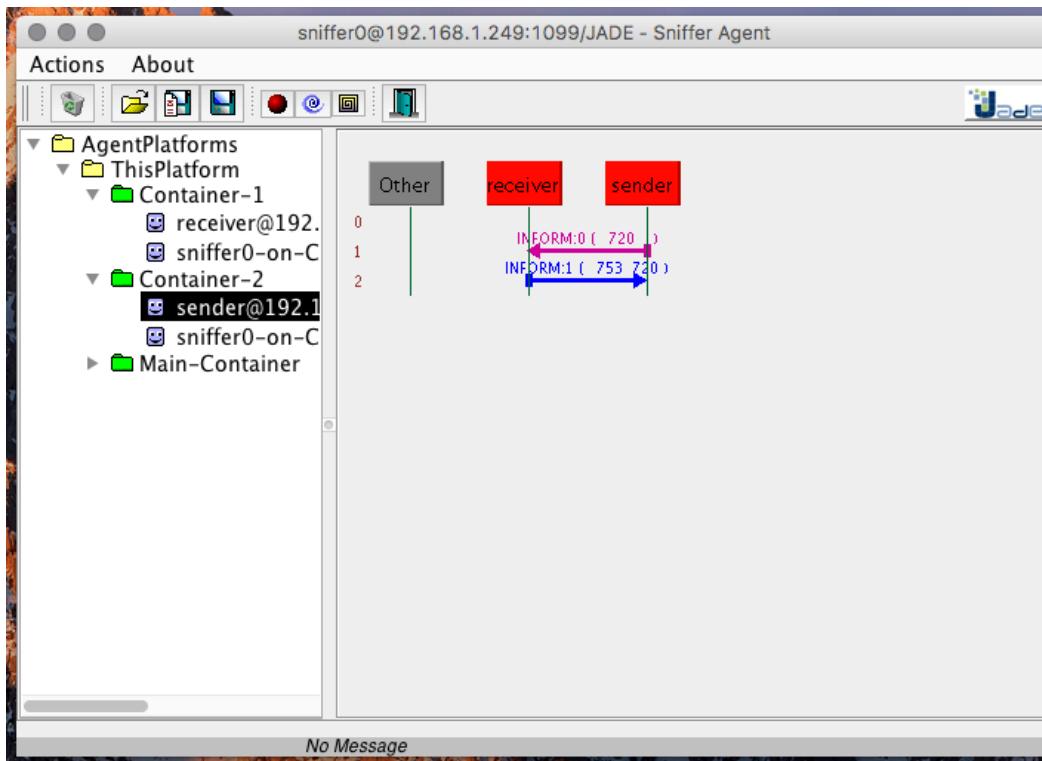


Message Example

```
(inform
  :sender      Admin@AdmM
  :receiver    AM0001@AmM
  :reply-with  1203302
  :language    XML
  :ontology   new_event
  :content     <new-event>
                <id> 0001           </id>
                <prio> 3            </prio>
                <sum> "Visit dentist" </sum>
                <dateb>20090514     </dateb>
                <timeb>110000         </timeb>
                <datee>20090514     </datee>
                <timee>123000         </timee>
              </new-event>
)
```

Agentes Inteligentes @ 2019/2020

Sniffer



ACL Message

ACLMessages Envelope

Sender:92.168.1.249:1099/JADE

Receivers: sender@192.168.1.249:1099/JADE

Reply-to:

Communicative ... inform

Content:
Agente Reciever.

Language:

Encoding:

Ontology:

Protocol: Null

Conversation-id:

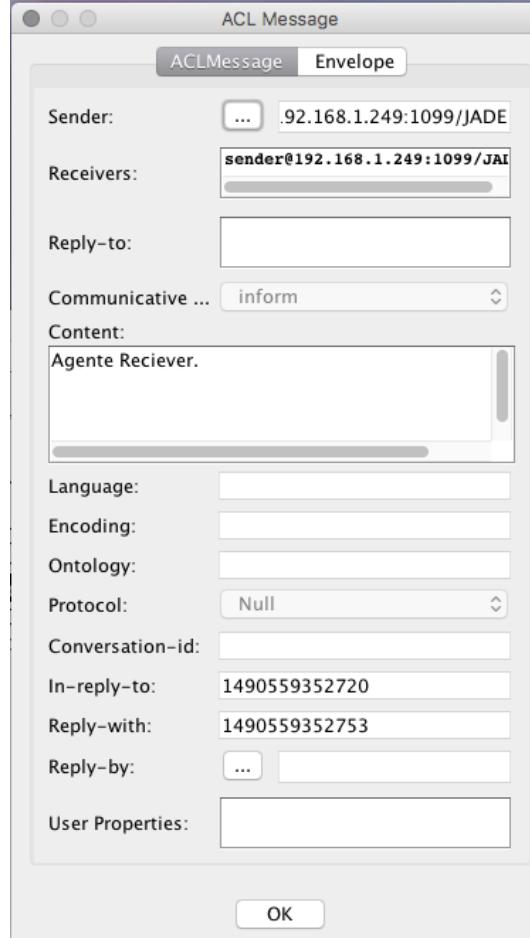
In-reply-to: 1490559352720

Reply-with: 1490559352753

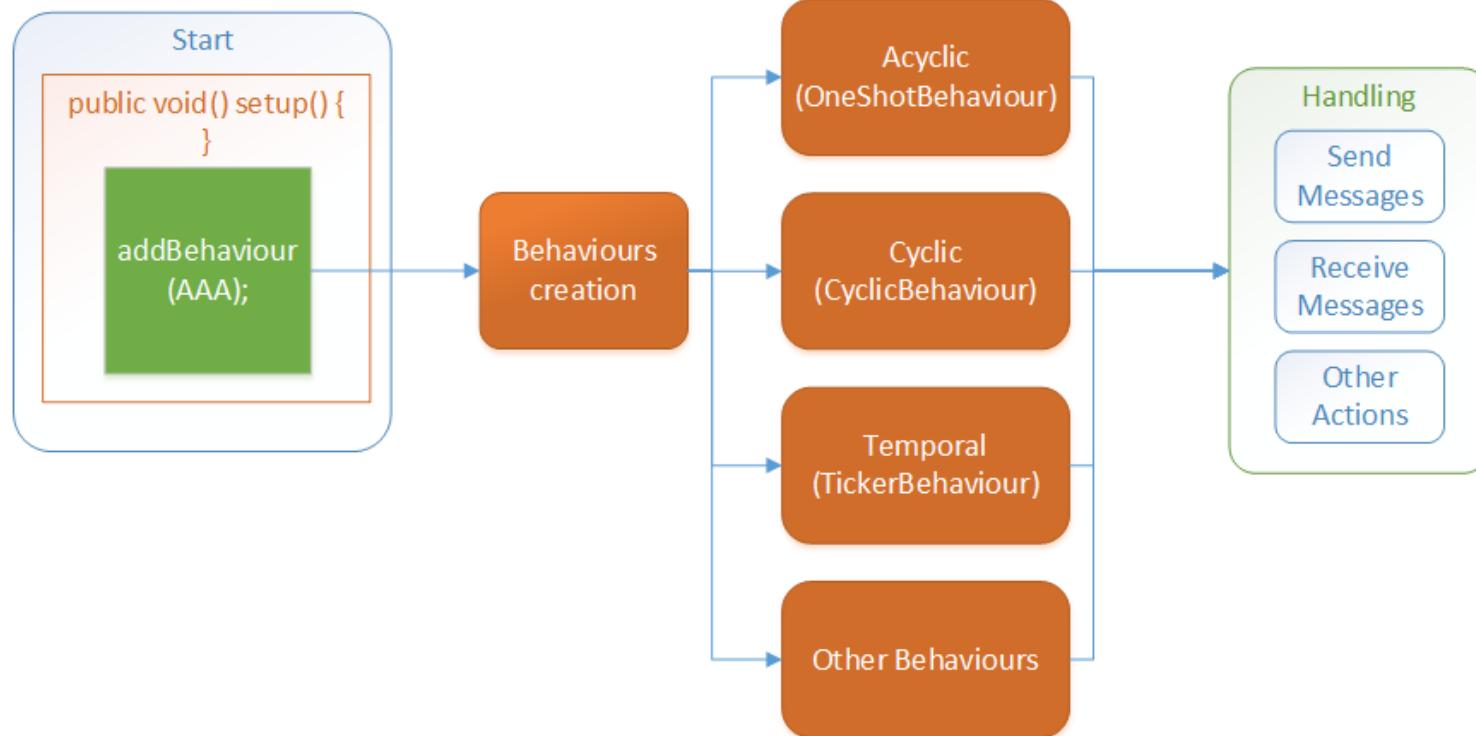
Reply-by:

User Properties:

OK



Development Process



Agent Skeleton

```
public class HelloAgent extends  
Agent {  
    public void setup() {  
        ...  
    }  
}
```



```
public class Hello extends Agent  
{  
    public void setup()  
    {  
        this.addBehaviour(do_something);  
    }  
}  
  
private class do_something extends  
CyclicBehaviour {  
    public void action()  
    {  
        // Actions that the behaviour will do  
        // Such as actions to send/receive messages  
    }  
}
```

Create & Send Messages

```
AID receiver = new AID();
receiver.setLocalName("receiver");
ACLMessage msg = new ACLMessage(ACLMessage.INFORM);
msg.setContent("HelloWorld");
msg.setConversationId(""+System.currentTimeMillis());
msg.addReceiver(receiver);
myAgent.send(msg);
```

Receive Messages

```
private ACLMessage msg = null;  
if ((msg = myAgent.receive()) != null)  
{  
    System.out.println("New message received");  
    System.out.println("Sender: " + msg.getSender().getLocalName());  
    System.out.println("Name:" + msg.getSender().getName());  
    System.out.println("Content: " + msg.getContent());  
}
```

Start Platform

- Run the following command to start the Main-Container (console):
 - `java jade.Boot [Options] [Agents]`
- [Options]:
 - `-container`
 - `-host "HostName"` (default: localhost)
 - `-port "PortNumber"` (default: 1099)
 - `-gui` (RMA GUI)
- [Agents]:
 - Identifiers separated by spaces
 - Identifier is of type: **AgentName:JavaPackage.ClassName**

Start Platform - Multiple Containers

- In the same machine:

- **Start JADE:** java -cp jade.Boot
 - **Add a new container with an agent:** java -cp jade.Boot -container

- In different machines:

- **Start JADE in machine PCAna:** java -cp jade.Boot
 - **Add new container in the machine PCJohn with an agent:**
 - java -cp jade.Boot -host PCAna -container a:ClasseA

➤ Agent **a** lives in the machine PCJohn, but belongs to the platform running in the machine PCAna; its GUID (Globally Unique ID) is a@PCAna:1099/JADE

Start Platform

- Tools that simplify platform administration and Application development:
 - Remote Monitoring Agent;
 - Dummy Agent;
 - Sniffer Agent;
 - Introspector Agent;
 - Directory Facilitator GUI;

Basic Agent

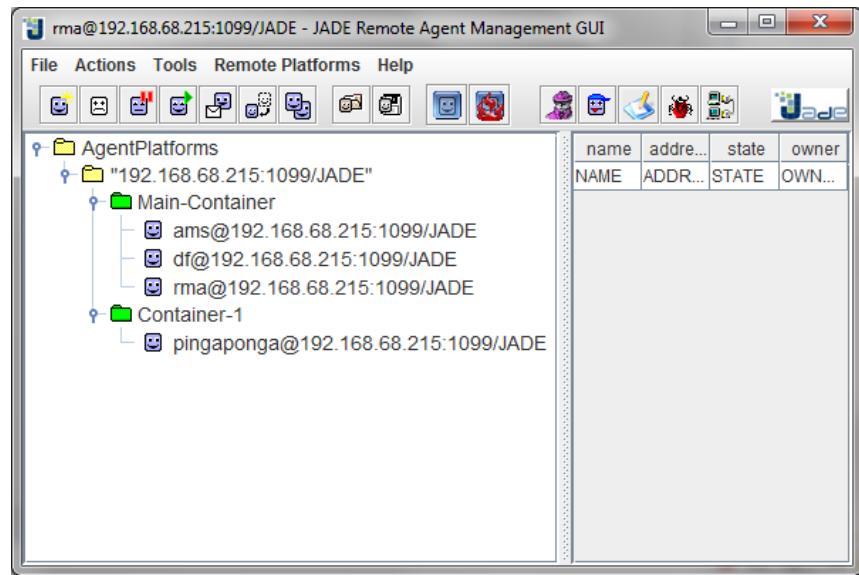
```
public class PingPong extends Agent
{
    @Override
    protected void setup()
    {
        super.setup();

        System.out.println(this.getLocalName()+" a começar!");
    }

    @Override
    protected void takeDown()
    {
        super.takeDown();

        System.out.println(this.getLocalName()+" a morrer...");
    }
}
```

```
set 15, 2015 10:17:58 AM jade.core.BaseService init
INFO: Service jade.core.management.AgentManagement initialized
set 15, 2015 10:17:58 AM jade.core.BaseService init
INFO: Service jade.core.messaging.Messaging initialized
set 15, 2015 10:17:58 AM jade.core.BaseService init
INFO: Service jade.core.resource.ResourceManagement initialized
set 15, 2015 10:17:58 AM jade.core.BaseService init
INFO: Service jade.core.mobility.AgentMobility initialized
set 15, 2015 10:17:58 AM jade.core.event.Notification initialized
set 15, 2015 10:17:58 AM jade.core.AgentContainerImpl joinPlatform
INFO: -----
Agent container Container-1@192.168.68.215 is ready.
-----
pingaponga a começar!
```



In order to execute `takeDown()` method, the agent must invoke the `doDelete()` method inside the `setup()` method.

Message Receive

```
public class PingPong extends Agent
{
    @Override
    protected void setup()
    {
        super.setup();
        System.out.println(this.getLocalName()+" a começar!");

        this.addBehaviour(new ReceiveBehavior());
    }

    @Override
    protected void takeDown()
    {
        super.takeDown();

        System.out.println(this.getLocalName()+" a morrer...");
    }

    private class ReceiveBehavior extends CyclicBehaviour
    {
        @Override
        public void action()
        {
            ACLMessage msg = receive();
            if (msg != null)
            {
                System.out.println("Recebi uma mensagem de "+msg.getSender()+" . Conteúdo: "+msg.getContent());
            }
            block();
        }
    };
}
```

Message Receive

Sniffer Agent

Actions About

AgentPlatforms

- ThisPlatform
 - Container-1
 - pingaponga@192.168.68.215:1099/JADE
 - sniffer0-on-Container
 - sniffer0@192.168.68.215:1099/JADE
- Main-Container
 - ams@192.168.68.215:1099/JADE
 - df@192.168.68.215:1099/JADE
 - rma@192.168.68.215:1099/JADE

Other pingaponga

0 NOT-UNDERSTOOD-1 1

JADE Remote Agent Management GUI

File Actions Tools Remote Platforms Help

AgentPlatforms "192.168.68.215:1099/JADE"

- Main-Container
 - ams@192.168.68.215:1099/JADE
 - df@192.168.68.215:1099/JADE
 - rma@192.168.68.215:1099/JADE
- Container-1
 - pingaponga@192.168.68.215:1099/JADE

Suspend Resume Kill Migrate Agent Clone Agent Save Agent Freeze Agent Send Message

ACL Message

ACLMessages Envelope

Sender: Set pingaponga@192.168.68.215:1099/JADE

Receivers: rma@192.168.68.215:1099/JADE

Reply-to:

Communicative act: not-understood

Content: ola

Language:

Encoding:

Ontology:

Protocol: Null

Conversation-id:

In-reply-to:

Reply-with:

Reply-by: Set

User Properties:

OK Cancel

```

set 15, 2015 10:23:28 AM jade.core.BaseService init
INFO: Service jade.core.event.Notification initialized
pingaponga a começar!
set 15, 2015 10:23:28 AM jade.core.AgentContainerImpl joinPlatform
INFO: -----
Agent container Container-2@192.168.68.215 is ready.
-----
Recebi uma mensagem de ( agent-identifier :name rma@192.168.68.215:1099/JADE :addresses (sequence http://192.168.68.215:7778/acc )). Conteúdo: ola
  
```

Message Reply

```
private class AnswerBehavior extends CyclicBehaviour
{
    @Override
    public void action()
    {
        ACLMessage msg = receive();
        if (msg != null)
        {
            System.out.println("Recebi uma mensagem de "+msg.getSender()+" Conteúdo: "+msg.getContent());
            ACLMessage resp = msg.createReply();

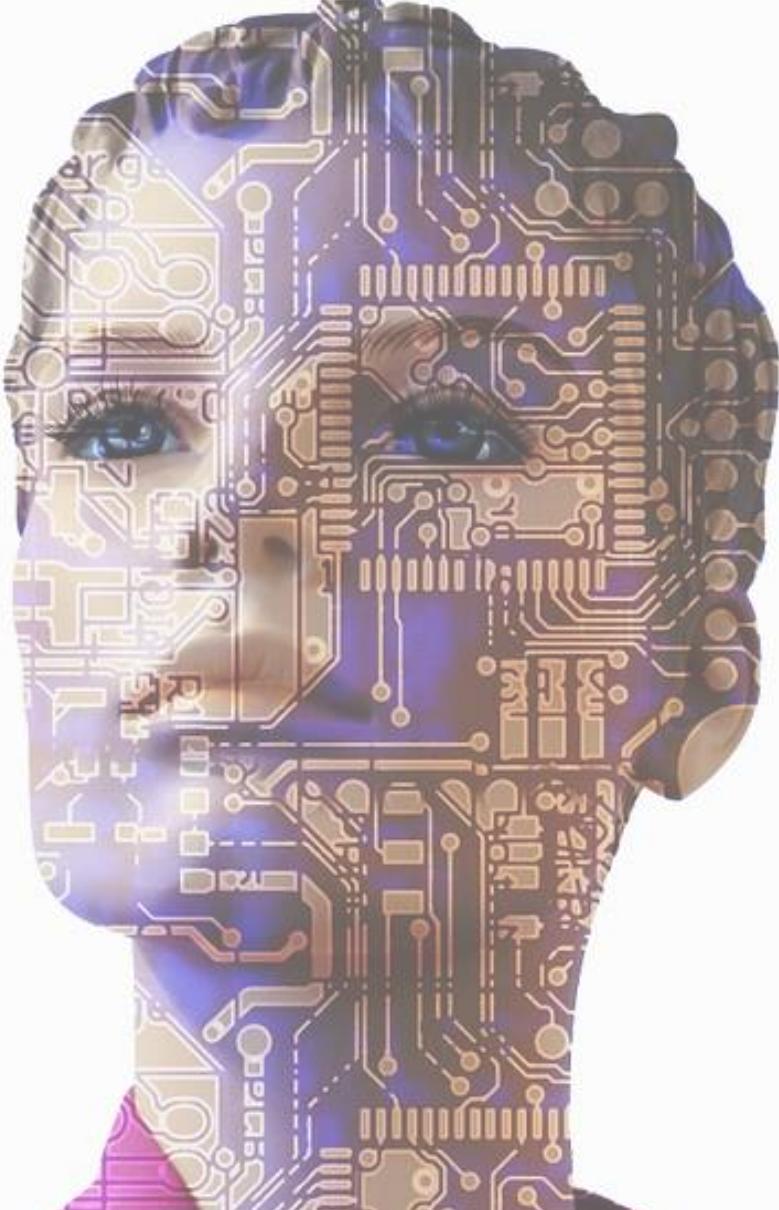
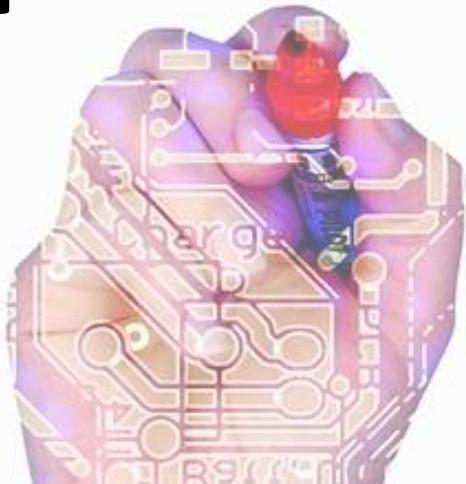
            if (msg.getContent().equals("ping"))
            {
                resp.setContent("pong");
                resp.setPerformative(ACLMessage.INFORM);
            }
            else
            {
                resp.setContent("Não percebi...");
                resp.setPerformative(ACLMessage.NOT_UNDERSTOOD);
            }

            send(resp);
        }
        block();
    }

    @Override
    protected void setup()
    {
        super.setup();
        System.out.println(this.getLocalName()+" a começar!");

        //this.addBehaviour(new ReceiveBehavior());
        this.addBehaviour(new AnswerBehavior());
    }
}
```

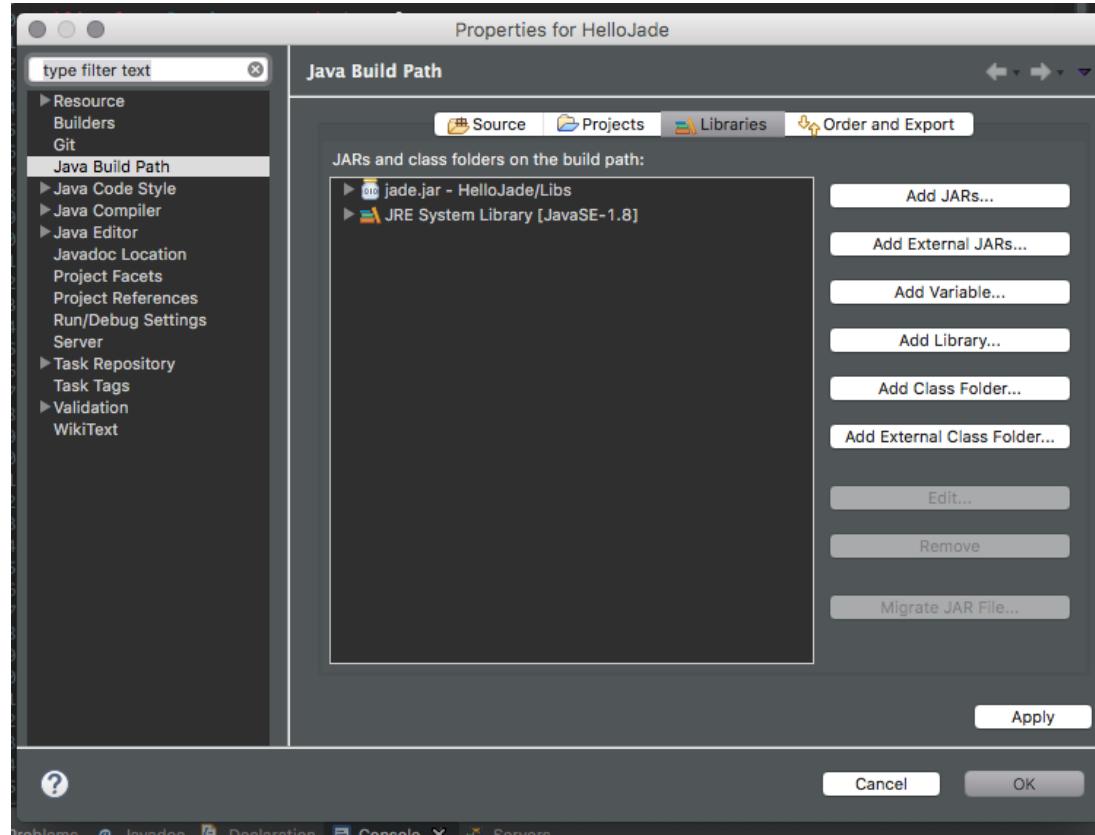
Environment Setup



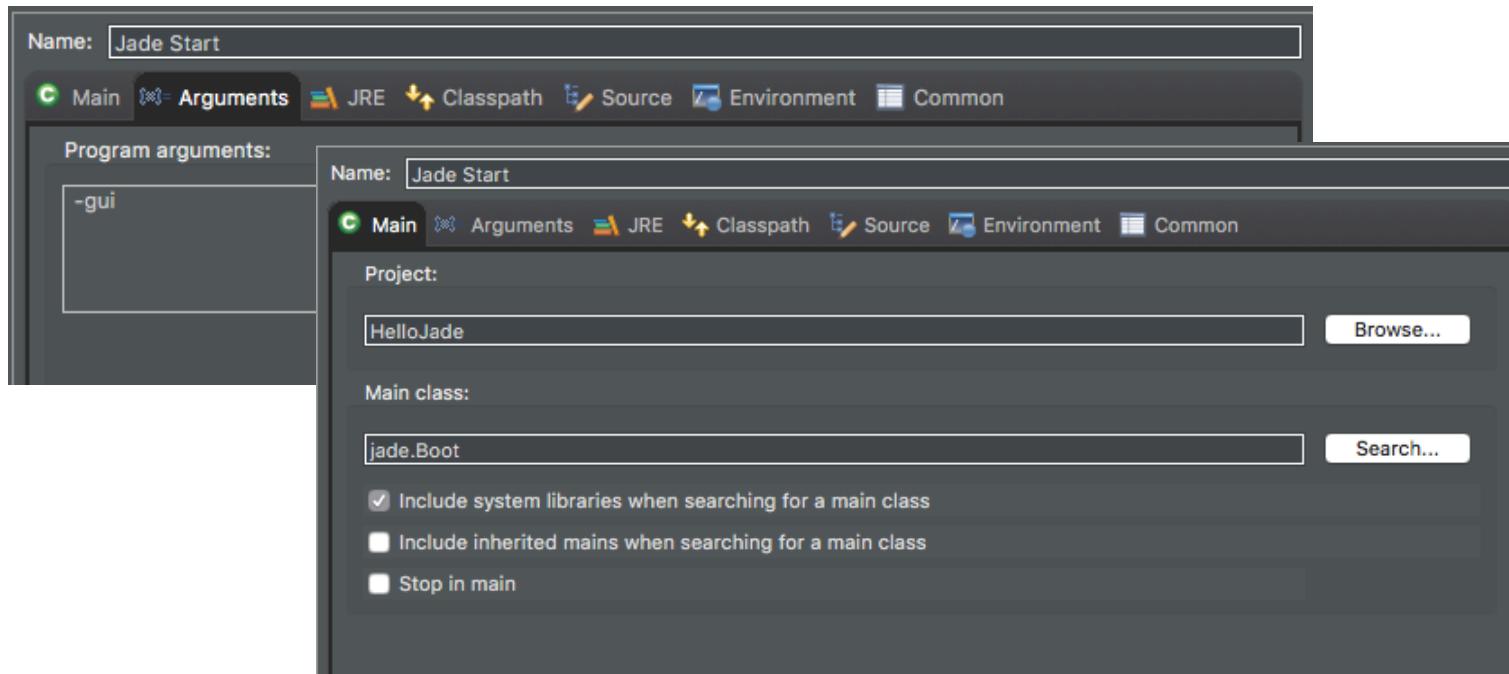
Software Required:

- Downloads:
 - Eclipse IDE for Java Developers (Mars version – 4.5)
 - <https://www.eclipse.org/downloads/packages/release/mars/r/eclipse-ide-java-developers>
 - JADE API (jadeBin is enough!)
 - <http://jade.tilab.com/dl.php?file=JADE-bin-4.5.0.zip>
- After downloading & Installing Eclipse:
 - Open Eclipse
 - Create new Java Project
 - Right Click Project -> Java Build Path -> Add External JAR (select JAR file found in JADE zip)
 - Time to develop your own Agent (<http://jade.tilab.com/doc/tutorials/JADEProgramming-Tutorial-for-beginners.pdf>)

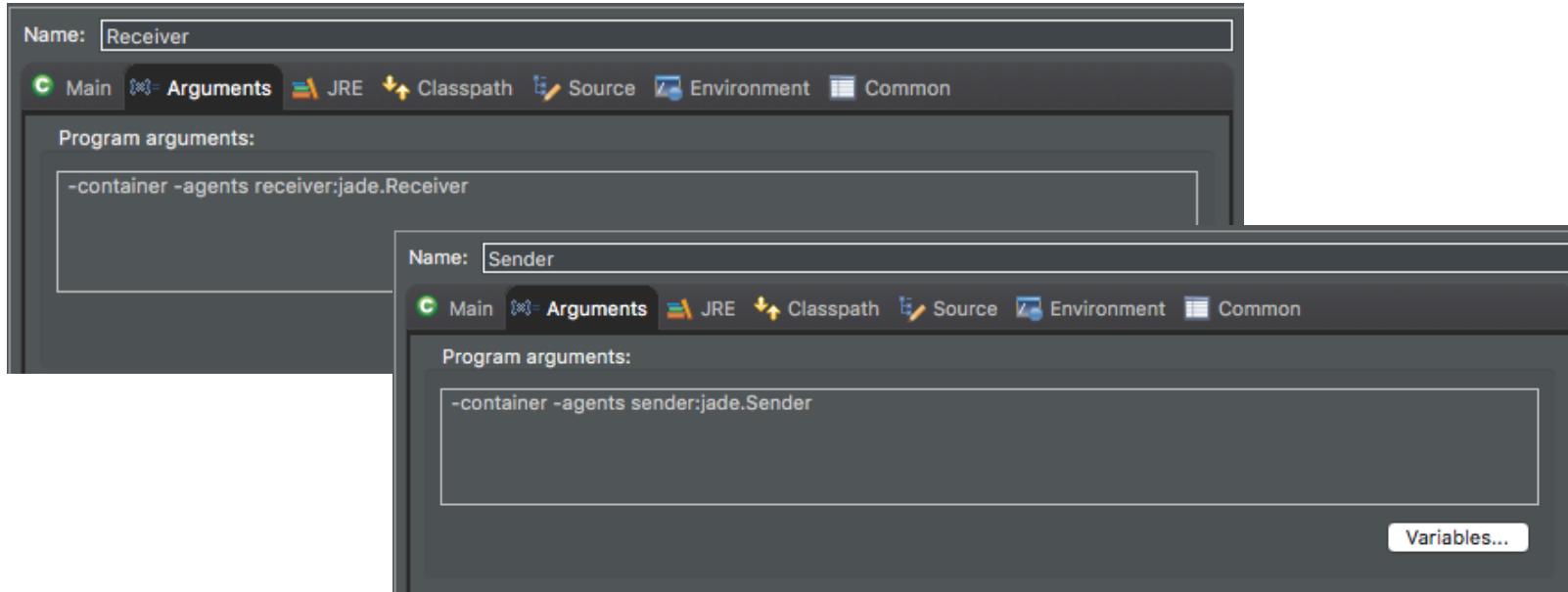
Eclipse Setup



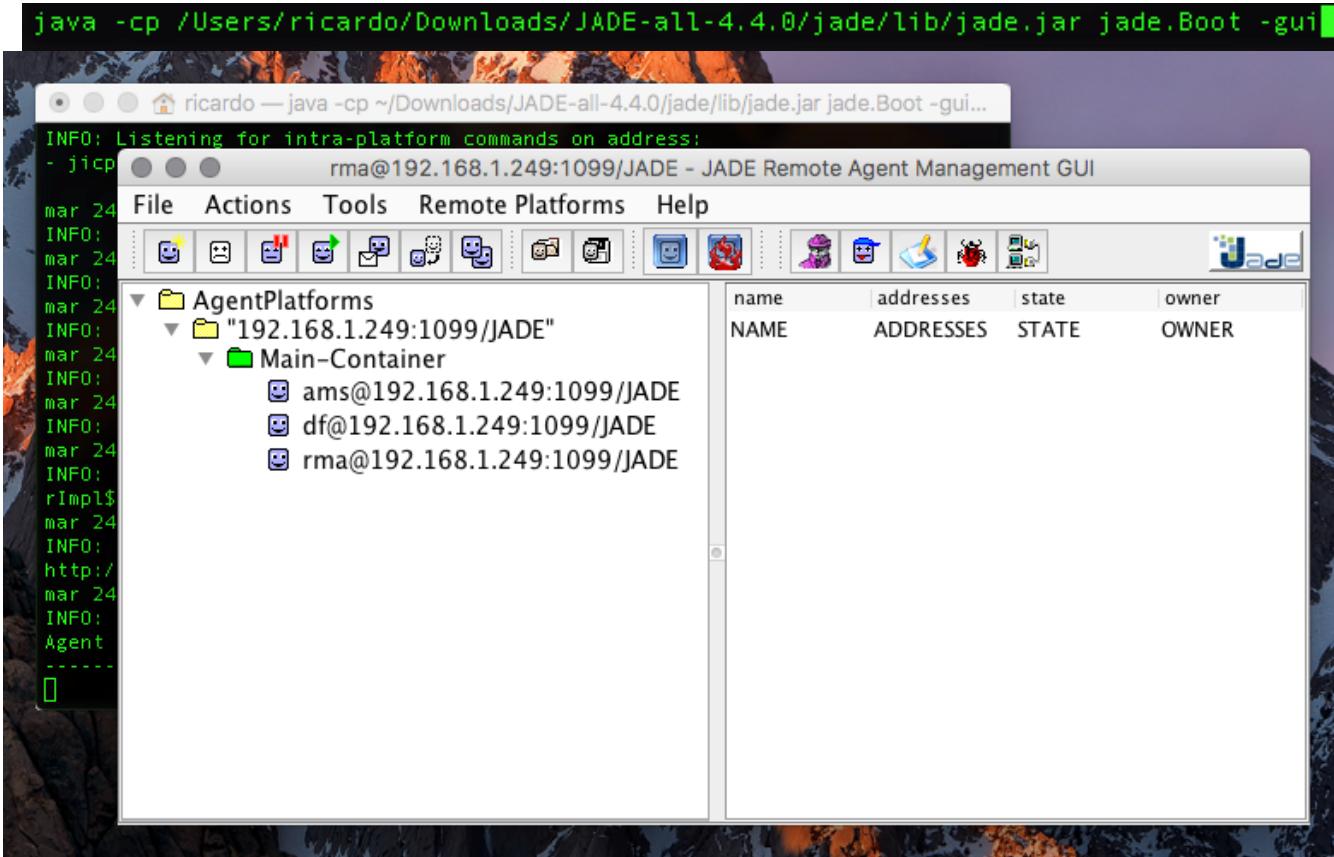
Run Configuration – Launch Main Container



Run Configuration – Execute Agent in Container



Jade Start



Jade Start

```
[iMac-de-Ricardo:~ ricardo$ java -cp /Users/ricardo/Downloads/JADE-all-4.4.0/jade]
/lib/jade.jar jade.Boot -gui
mar 26, 2017 9:25:42 PM jade.core.Runtime beginContainer
INFO: -----
      This is JADE 4.4.0 - revision 6778 of 21-12-2015 12:24:43
      downloaded in Open Source, under LGPL restrictions,
      at http://jade.tilab.com/
-----
mar 26, 2017 9:25:42 PM jade.imtp.leap.LEAPIMTPManager initialize
INFO: Listening for intra-platform commands on address:
- jicp://192.168.1.249:1099

mar 26, 2017 9:25:43 PM jade.core.BaseService init
INFO: Service jade.core.management.AgentManagement initialized
mar 26, 2017 9:25:43 PM jade.core.BaseService init
INFO: Service jade.core.messaging.Messaging initialized
mar 26, 2017 9:25:43 PM jade.core.BaseService init
INFO: Service jade.core.resource.ResourceManagement initialized
mar 26, 2017 9:25:43 PM jade.core.BaseService init
INFO: Service jade.core.mobility.AgentMobility initialized
mar 26, 2017 9:25:43 PM jade.core.BaseService init
INFO: Service jade.core.event.Notification initialized
mar 26, 2017 9:25:43 PM jade.mtp.http.HTTPServer <init>
INFO: HTTP-MTP Using XML parser com.sun.org.apache.xerces.internal.jaxp.SAXParse
rImpl$JAXPSAXParser
mar 26, 2017 9:25:43 PM jade.core.messaging.MessagingService boot
INFO: MTP addresses:
http://192.168.1.249:7778/acc
mar 26, 2017 9:25:43 PM jade.core.AgentContainerImpl joinPlatform
INFO: -----
Agent container Main-Container@192.168.1.249 is ready.
-----
```



Universidade do Minho

Escola de Engenharia

Departamento de Informática

**Mestrado Integrado em Engenharia Informática
Mestrado em Engenharia Informática
Agentes Inteligentes
2019/2020**

Paulo Novais, Filipe Gonçalves