

# JADE

# Java Agent DEvelopment Framework

Integrated Master's in Informatics Engineering

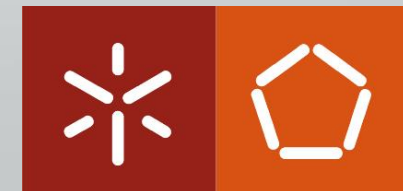
Intelligent Agents

2018/2019

Synthetic Intelligence Lab

Filipe Gonçalves

César Analide



## Useful Links

- <http://jade.tilab.com/>
- <http://jade.tilab.com/doc/tutorials/JADEProgramming-Tutorial-for-beginners.pdf>
- <http://jade.tilab.com/doc/api/jade/core/Agent.html>
- <http://jade.tilab.com/doc/api/jade/core/behaviours/Behaviour.html>
- <http://jade.tilab.com/doc/api/jade/lang/acl/ACLMessage.html>

# Performatives

```

        ACLMessage msg1 = new ACLMessage(ACLMessage.);
        msg1.setContent("ping");
        msg1.setConversationId(""+time);
        msg1.addReceiver(receiver);
        myAgent.send(msg1);
    }else{
        ACLMessage msg2 = new ACLMessage(ACLMessage.INFORM);
        msg2.setContent("not ping");
        msg2.setConversationId(""+time);
        msg2.addReceiver(receiver);
        myAgent.send(msg2);
        ACLMessage msg = new ACLMessage(ACLMessage.INFORM);
    }

    block(1000);
}
}

```

ACCEPT\_PROPOSAL : int - ACLMessage  
 AGREE : int - ACLMessage  
 CANCEL : int - ACLMessage  
 CFP : int - ACLMessage  
 CONFIRM : int - ACLMessage  
 DISCONFIRM : int - ACLMessage  
 FAILURE : int - ACLMessage  
 INFORM : int - ACLMessage  
 INFORM\_IF : int - ACLMessage  
 INFORM\_REF : int - ACLMessage  
 NOT\_UNDERSTOOD : int - ACLMessage  
 PROPAGATE : int - ACLMessage

Press 'Ctrl+Space' to show Template Proposals

INT	Performative	INT	Performative
0	ACCEPT_PROPOSAL	11	PROPOSE
1	AGREE	12	QUERY-IF
2	CANCEL	13	QUERY-REF
3	CFP	14	REFUSE
4	CONFIRM	15	REJECT_PROPOSAL
5	DISCONFIRM	16	REQUEST
6	FAILURE	17	REQUEST_WHEN
7	INFORM	18	REQUEST_WHENEVER
8	INFORM-IF	19	SUBSCRIBE
9	INFORM-REF	20	PROXY
10	NOT_UNDERSTOOD	21	PROPAGATE

# Performatives

```
public class SenderAsk extends Agent{

    @Override
    protected void setup(){

        super.setup();

        this.addBehaviour(new ReceiveBehaviour());
        this.addBehaviour(new SendMessage(this,2000));

    }

    public class SendMessage extends TickerBehaviour {

        public SendMessage(Agent a, long timeout){

            super(a,timeout);

        }

        @Override
        public void onTick(){

            AID receiver=new AID();
            receiver.setLocalName("pingaponga");
            long time =System.currentTimeMillis();

            if(time % 2 == 0){
                ACLMessage msg1 =new ACLMessage(ACLMessage.PROPOSE);
                msg1.setContent("Let's play ping pong?");
                msg1.setConversationId(""+time);
                msg1.addReceiver(receiver);
                myAgent.send(msg1);
            }else{
                ACLMessage msg2 =new ACLMessage(ACLMessage.INFORM);
                msg2.setContent("Let's play ping pong.");
                msg2.setConversationId(""+time);
                msg2.addReceiver(receiver);
                myAgent.send(msg2);
            }

            block(1000);

        }

    }

}
```

```
public class ReceiveBehaviour extends CyclicBehaviour {

    @Override
    public void action(){
        ACLMessage msg =receive();
        if(msg != null){

            if(msg.getPerformative()==0){

                System.out.println("Recebi uma mensagem de "+msg.getSender()+".Conteúdo: Ok! Let's play ping pong.");

            }else System.out.println("Recebi uma mensagem de "+msg.getSender()+".Conteúdo: The offer to play ping pong was not accepted.");

        }

    }

    block();

}

}
```

0: ACCEPT\_PROPOSAL

# Performatives

- New Behaviour in pingaponga agent.

```
public class ReceiveBehaviourProposal extends CyclicBehaviour {

    @Override
    public void action(){
        ACLMessage msg =receive();
        if(msg != null){
            ACLMessage resp=msg.createReply();
            if(msg.getPerformative()==11){

                System.out.println("Recebi uma mensagem de "+msg.getSender()+".Conteúdo: "+msg.getContent());

                resp.setContent("Yes");
                resp.setPerformative(ACLMessage.ACCEPT_PROPOSAL);

            }else{
                System.out.println("Recebi uma mensagem de "+msg.getSender()+".Conteúdo: "+msg.getContent());

                resp.setContent("No");
                resp.setPerformative(ACLMessage.NOT_UNDERSTOOD);

            }

            send(resp);

        }

        block();
    }
}
```

11: PROPOSE

# Performatives

- Configurations:
  - -container -agents pingaponga:agents.PingPong;sender:agents.SenderAsk

```
pingaponga a começar!
Recebi uma mensagem de ( agent-identifier :name sender@192.168.1.76:1099/JADE :addresses (sequence http://Laptop.lan:7778/acc )).Conteúdo: Let's play ping pong.
Recebi uma mensagem de ( agent-identifier :name pingaponga@192.168.1.76:1099/JADE :addresses (sequence http://Laptop.lan:7778/acc )).Conteúdo: The offer to play ping pong was
Recebi uma mensagem de ( agent-identifier :name sender@192.168.1.76:1099/JADE :addresses (sequence http://Laptop.lan:7778/acc )).Conteúdo: Let's play ping pong.
Recebi uma mensagem de ( agent-identifier :name pingaponga@192.168.1.76:1099/JADE :addresses (sequence http://Laptop.lan:7778/acc )).Conteúdo: The offer to play ping pong was
Recebi uma mensagem de ( agent-identifier :name sender@192.168.1.76:1099/JADE :addresses (sequence http://Laptop.lan:7778/acc )).Conteúdo: Let's play ping pong?
Recebi uma mensagem de ( agent-identifier :name pingaponga@192.168.1.76:1099/JADE :addresses (sequence http://Laptop.lan:7778/acc )).Conteúdo: Ok! Let's play ping pong.
Recebi uma mensagem de ( agent-identifier :name sender@192.168.1.76:1099/JADE :addresses (sequence http://Laptop.lan:7778/acc )).Conteúdo: Let's play ping pong?
Recebi uma mensagem de ( agent-identifier :name pingaponga@192.168.1.76:1099/JADE :addresses (sequence http://Laptop.lan:7778/acc )).Conteúdo: Ok! Let's play ping pong.
Recebi uma mensagem de ( agent-identifier :name sender@192.168.1.76:1099/JADE :addresses (sequence http://Laptop.lan:7778/acc )).Conteúdo: Let's play ping pong.
Recebi uma mensagem de ( agent-identifier :name pingaponga@192.168.1.76:1099/JADE :addresses (sequence http://Laptop.lan:7778/acc )).Conteúdo: The offer to play ping pong was
Recebi uma mensagem de ( agent-identifier :name sender@192.168.1.76:1099/JADE :addresses (sequence http://Laptop.lan:7778/acc )).Conteúdo: Let's play ping pong?
Recebi uma mensagem de ( agent-identifier :name pingaponga@192.168.1.76:1099/JADE :addresses (sequence http://Laptop.lan:7778/acc )).Conteúdo: Ok! Let's play ping pong.
Recebi uma mensagem de ( agent-identifier :name sender@192.168.1.76:1099/JADE :addresses (sequence http://Laptop.lan:7778/acc )).Conteúdo: Let's play ping pong?
Recebi uma mensagem de ( agent-identifier :name pingaponga@192.168.1.76:1099/JADE :addresses (sequence http://Laptop.lan:7778/acc )).Conteúdo: Ok! Let's play ping pong.
Recebi uma mensagem de ( agent-identifier :name sender@192.168.1.76:1099/JADE :addresses (sequence http://Laptop.lan:7778/acc )).Conteúdo: Let's play ping pong?
Recebi uma mensagem de ( agent-identifier :name pingaponga@192.168.1.76:1099/JADE :addresses (sequence http://Laptop.lan:7778/acc )).Conteúdo: Ok! Let's play ping pong.
```

# Performatives

```
public class SenderTestFilter extends Agent {

    @Override
    protected void setup(){

        super.setup();

        this.addBehaviour(new SendMessage(this,2000));
    }

    public class SendMessage extends TickerBehaviour {

        public SendMessage(Agent a, long timeout){

            super(a,timeout);
        }
    }
}
```

```
public class SendMessage extends TickerBehaviour {

    public SendMessage(Agent a, long timeout){

        super(a,timeout);
    }

    @Override
    public void onTick(){
        AID receiver=new AID();
        receiver.setLocalName("pingaponga");
        long time =System.currentTimeMillis();

        if(time % 2 == 0){
            ACLMessage msg1 =new ACLMessage(ACLMessage.PROPOSE);
            msg1.setOntology("event");
            msg1.setContent("Let's play ping pong?");
            msg1.setConversationId(""+time);
            msg1.addReceiver(receiver);
            myAgent.send(msg1);
        }else{
            ACLMessage msg2 =new ACLMessage(ACLMessage.INFORM);
            msg2.setContent("Let's play ping pong.");
            msg2.setConversationId(""+time);
            msg2.addReceiver(receiver);
            myAgent.send(msg2);
        }

        block(1000);
    }
}
```

# Filter Messages from Message Pool

```
public class SenderTestFilter extends Agent {

    @Override
    protected void setup(){

        super.setup();

        this.addBehaviour(new SendMessage(this,2000));

    }

    public class SendMessage extends TickerBehaviour {

        public SendMessage(Agent a, long timeout){

            super(a,timeout);

        }

    }

}
```

```
public class SendMessage extends TickerBehaviour {

    public SendMessage(Agent a, long timeout){

        super(a,timeout);

    }

    @Override
    public void onTick(){
        AID receiver=new AID();
        receiver.setLocalName("pingaponga");
        long time =System.currentTimeMillis();

        if(time % 2 == 0){
            ACLMessage msg1 =new ACLMessage(ACLMessage.PROPOSE);
            msg1.setOntology("event");
            msg1.setContent("Let's play ping pong?");
            msg1.setConversationId(""+time);
            msg1.addReceiver(receiver);
            myAgent.send(msg1);
        }else{
            ACLMessage msg2 =new ACLMessage(ACLMessage.INFORM);
            msg2.setContent("Let's play ping pong.");
            msg2.setConversationId(""+time);
            msg2.addReceiver(receiver);
            myAgent.send(msg2);
        }

        block(1000);

    }

}
```



# Filter Messages from Message Pool

- New Behaviour in pingaponga agent.

The pingaponga agent will only process messages with performative PROPOSE.

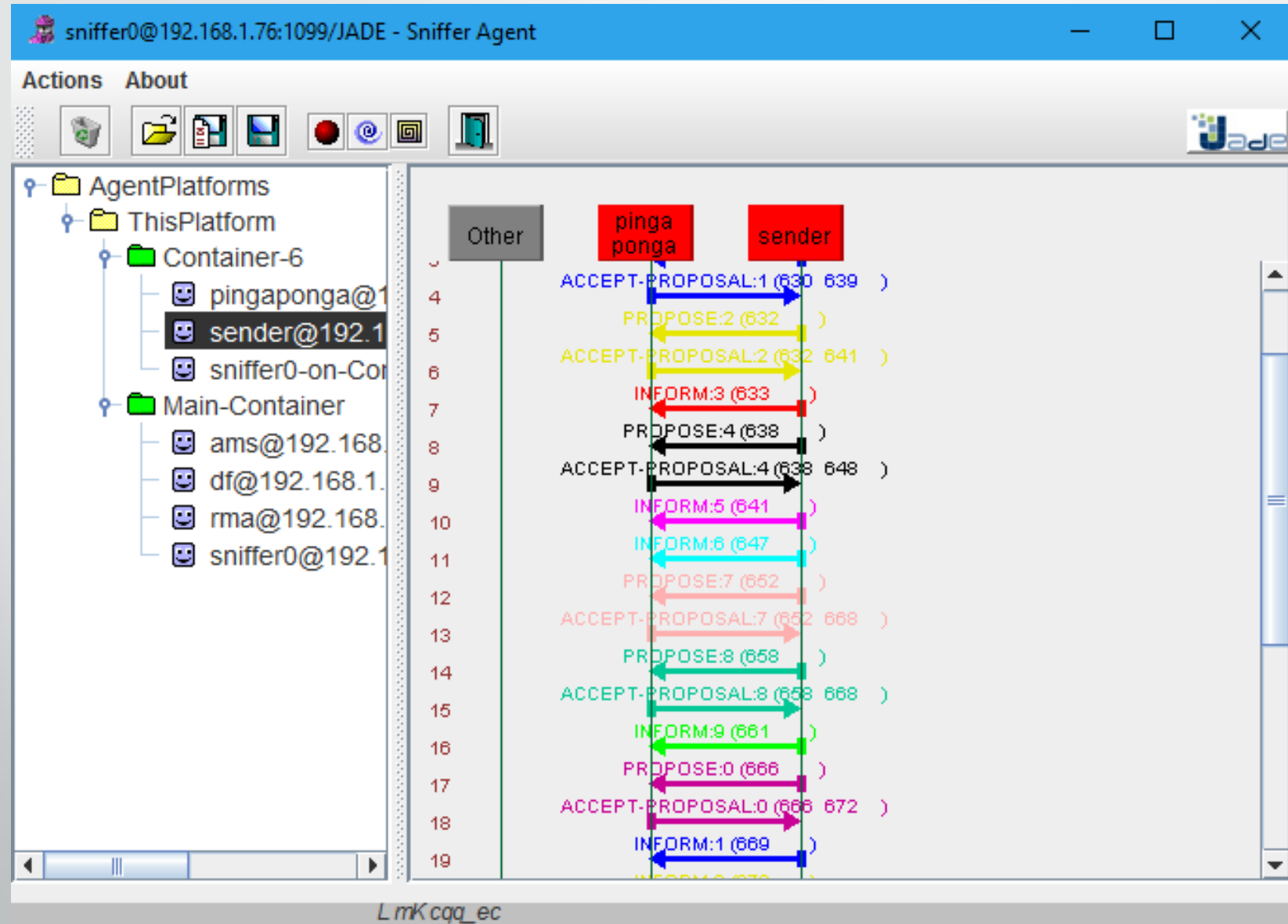
```
public class ReceiveBehaviourFilter extends CyclicBehaviour {  
  
    @Override  
    public void action(){  
        MessageTemplate mt1= MessageTemplate.MatchPerformative(ACLMessage.PROPOSE);  
        MessageTemplate mt2= MessageTemplate.MatchOntology("event");  
        MessageTemplate mt3= MessageTemplate.and(mt1, mt2);  
        ACLMessage msg =receive(mt1);  
        if(msg != null){  
            ACLMessage resp=msg.createReply();  
            System.out.println("Recebi uma mensagem de "+msg.getSender()+".Conteúdo: "+msg.getContent());  
            resp.setContent("Yes");  
            resp.setPerformative(ACLMessage.ACCEPT_PROPOSAL);  
            send(resp);  
        }  
        block();  
    }  
}
```

# Filter Messages from Message Pool

- Configuration:
  - -container -agents pingaponga:agents.PingPong;sender:agents.SenderTestFilter

```
pingaponga a começar!  
Recebi uma mensagem de ( agent-identifier :name sender@192.168.1.76:1099/JADE :addresses (sequence http://Laptop.lan:7778/acc )) .Conteúdo: Let's play ping pong?  
Recebi uma mensagem de ( agent-identifier :name sender@192.168.1.76:1099/JADE :addresses (sequence http://Laptop.lan:7778/acc )) .Conteúdo: Let's play ping pong?  
Recebi uma mensagem de ( agent-identifier :name sender@192.168.1.76:1099/JADE :addresses (sequence http://Laptop.lan:7778/acc )) .Conteúdo: Let's play ping pong?  
Recebi uma mensagem de ( agent-identifier :name sender@192.168.1.76:1099/JADE :addresses (sequence http://Laptop.lan:7778/acc )) .Conteúdo: Let's play ping pong?  
Recebi uma mensagem de ( agent-identifier :name sender@192.168.1.76:1099/JADE :addresses (sequence http://Laptop.lan:7778/acc )) .Conteúdo: Let's play ping pong?  
Recebi uma mensagem de ( agent-identifier :name sender@192.168.1.76:1099/JADE :addresses (sequence http://Laptop.lan:7778/acc )) .Conteúdo: Let's play ping pong?  
Recebi uma mensagem de ( agent-identifier :name sender@192.168.1.76:1099/JADE :addresses (sequence http://Laptop.lan:7778/acc )) .Conteúdo: Let's play ping pong?
```

# Filter Messages from Message Pool



# Agents with Graphical Interface

- A GUI already has a mechanism for dealing with events, implemented in the `actionPerformed()` method.
- `actionPerformed()` method is associated with each of the components registered with an `ActionListener` object.
- When a call to the GUI is made, an `ActionEvent` is generated by the component, invoking the `actionPerformed()` method.
- When an agent interacts with a GUI, it is actually invoking the code provided in this mechanism to handle events.

# Agents with Graphical Interface

- JADE provides the GuiAgent abstract class that extends the Agent class.
- This class contains the postGuiEvent() and onGuiEvent() methods.
- onGuiEvent() contains the methods required for the agent to receive and process the events provided by the GUI through the postGuiEvent() method.
- The GUI provides an event to the agent through the GuiEvent object.

# Agents with Graphical Interface

- Class that implements the interface.

```
public class SenderGui {

    public JFrame frame;
    private JTextField textField;
    private SenderWithGui myAgent;

    /**
     * Launch the application.
     */

    public SenderGui (SenderWithGui a) {
        initialize();
        myAgent = a;
    }

    public JTextField getTextField() {
        return textField;
    }

    public static void main(String[] args) {
        EventQueue.invokeLater(new Runnable() {
            public void run() {
                try {
                    SenderGui window = new SenderGui();
                    window.frame.setVisible(true);
                } catch (Exception e) {
                    e.printStackTrace();
                }
            }
        });
    }
}
```

Reference to the agent that implements the interface

Provide the value for class SenderWithGui

```
public SenderGui() {
    initialize();
}

private void initialize() {
    frame = new JFrame();
    frame.setBounds(100, 100, 450, 300);
    frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);

    JPanel panel = new JPanel();
    frame.getContentPane().add(panel, BorderLayout.NORTH);

    JLabel lblTypeAMessage = new JLabel("Type a message to send to pingaponga:");
    panel.add(lblTypeAMessage);

    textField = new JTextField();
    frame.getContentPane().add(textField, BorderLayout.CENTER);
    textField.setColumns(10);

    JButton btnNewButton = new JButton("Send");
    frame.getContentPane().add(btnNewButton, BorderLayout.SOUTH);
    btnNewButton.addActionListener(new ActionListener() {
        public void actionPerformed(ActionEvent arg0) {
            textField.getText();
            GuiEvent ge = new GuiEvent(textField.getText(),1);
            myAgent.postGuiEvent(ge);
        }
    });
}
```

Transition from event to agent

# Agents with Graphical Interface

- Agent that implements the interface.

```
public class SenderWithGui extends GuiAgent {

    protected SenderGui myGui;

    protected void setup() {
        myGui = new SenderGui(this);
        myGui.frame.setVisible(true);
    }

    @Override
    protected void onGuiEvent(GuiEvent ev) {
        int command=ev.getType();

        if(command==1){

            String content = (String)ev.getSource();
            AID receiver=new AID();
            receiver.setLocalName("pingaponga");
            long time =System.currentTimeMillis();
            ACLMessage msg =new ACLMessage(ACLMessage.INFORM);
            msg.setContent(content);
            msg.setConversationId(""+time);
            msg.addReceiver(receiver);
            send(msg);

        }
    }
}
```

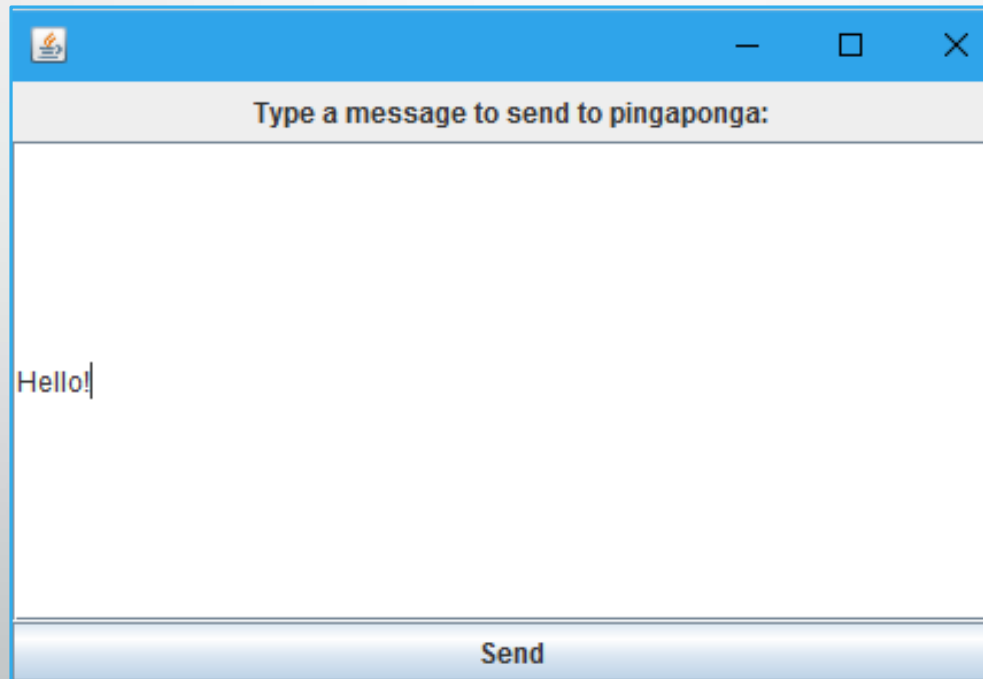
Interface instantiation

Processing of the event according to its type.

# Agents with Graphical Interface

- Configuration:
  - -container -agents pingaponga:agents.PingPong;sender:agents.SenderWithGui

```
pingaponga a começar!  
Recebi uma mensagem de ( agent-identifier :name sender@192.168.1.76:1099/JADE :addresses (sequence http://Laptop.lan:7778/acc )).Conteúdo: Hello!
```



The screenshot shows a window titled "Type a message to send to pingaponga:". Inside the window, there is a text input field containing the text "Hello!". Below the input field is a "Send" button. The window has a standard Windows-style title bar with minimize, maximize, and close buttons.



# Agents with Graphical Interface

- Dynamically add a new behaviour.
  - Add a new button to SenderGui.

```
btnNewButton_1.addActionListener(new ActionListener() {  
    public void actionPerformed(ActionEvent e) {  
        textField.getText();  
        GuiEvent ge = new GuiEvent(textField.getText(),2);  
        myAgent.postGuiEvent(ge);  
    }  
});
```

Transition from event to agent

# Agents with Graphical Interface

- Dynamically add a new behaviour.
  - Add the behaviour SendMessage and the processing of a new event to SenderWithGui.

```
public class SendMessage extends CyclicBehaviour {

    String message;

    public SendMessage(String content){

        message=content;
    }

    @Override
    public void action(){

        AID receiver=new AID();
        receiver.setLocalName("pingaponga");

        ACLMessage msg =new ACLMessage(ACLMessage.INFORM);
        long time =System.currentTimeMillis();
        msg.setConversationId(""+time);
        msg.addReceiver(receiver);
        msg.setContent(message);
        myAgent.send(msg);
        block(1000);

    }

}
```

```
@Override
protected void onGuiEvent(GuiEvent ev) {
    int command=ev.getType();

    if(command==1){

        String content = (String)ev.getSource();
        AID receiver=new AID();
        receiver.setLocalName("pingaponga");
        long time =System.currentTimeMillis();
        ACLMessage msg =new ACLMessage(ACLMessage.INFORM);
        msg.setContent(content);
        msg.setConversationId(""+time);
        msg.addReceiver(receiver);
        send(msg);

    }else {

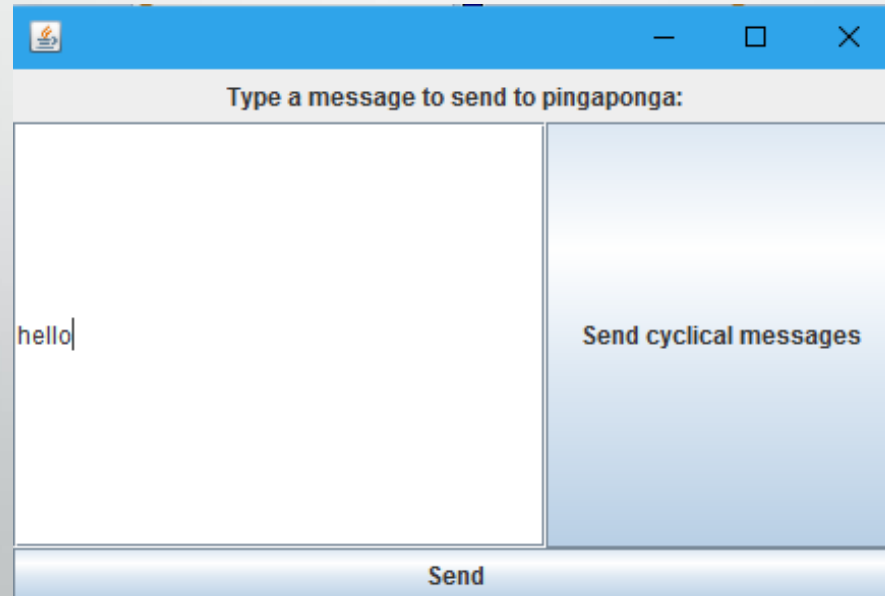
        this.addBehaviour(new SendMessage((String)ev.getSource()));

    }
}
```

# Agents with Graphical Interface

- Configuration:
  - -container -agents pingaponga:agents.PingPong;sender:agents.SenderWithGui

```
pingaponga a começar!
Recebi uma mensagem de ( agent-identifier :name sender@192.168.1.76:1099/JADE :addresses (sequence http://Laptop.lan:7778/acc )).Conteúdo: hello
Recebi uma mensagem de ( agent-identifier :name sender@192.168.1.76:1099/JADE :addresses (sequence http://Laptop.lan:7778/acc )).Conteúdo: hello
Recebi uma mensagem de ( agent-identifier :name sender@192.168.1.76:1099/JADE :addresses (sequence http://Laptop.lan:7778/acc )).Conteúdo: hello
Recebi uma mensagem de ( agent-identifier :name sender@192.168.1.76:1099/JADE :addresses (sequence http://Laptop.lan:7778/acc )).Conteúdo: hello
Recebi uma mensagem de ( agent-identifier :name sender@192.168.1.76:1099/JADE :addresses (sequence http://Laptop.lan:7778/acc )).Conteúdo: hello
Recebi uma mensagem de ( agent-identifier :name sender@192.168.1.76:1099/JADE :addresses (sequence http://Laptop.lan:7778/acc )).Conteúdo: hello
Recebi uma mensagem de ( agent-identifier :name sender@192.168.1.76:1099/JADE :addresses (sequence http://Laptop.lan:7778/acc )).Conteúdo: hello
Recebi uma mensagem de ( agent-identifier :name sender@192.168.1.76:1099/JADE :addresses (sequence http://Laptop.lan:7778/acc )).Conteúdo: hello
```



# References

- JADE: Java Agent DEvelopment framework:
  - <http://jade.tilab.com/>
- Bellifemine F., Caire G., Greenwood D., Developing Multi-Agent Systems with JADE, John Wiley & Sons, ISBN: 978-0470057476, 2007.
- Giovanni, Caire; TUTORIAL, J. A. D. E. JADE programming for beginners., 2003.

# JADE

# Java Agent DEvelopment Framework

Integrated Master's in Informatics Engineering

Intelligent Agents

2018/2019

Synthetic Intelligence Lab

Filipe Gonçalves

César Analide

