PanTAIA y Pintores

MAS of arts

KIT – University of the State of Baden-Württemberg and
National Large-scale Research Center of the Helmholtz Association

www.kit.edu
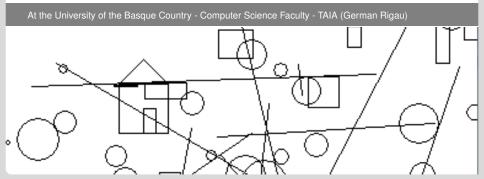
Introduction

Scenario

Some artists are working together to paint a picture.

Two different types of agents:

- PantallaAgent (only one)
- PintorAgent (more than one)
- Different types of PintorAgents can be created by commandline argument

Communication

Sending drawable objects to the PantallaAgent

- PantallaAgent is registered as a service, so it can be found by the artists
- To send objects they need to implement serializable
- Use msg.setContentObject(drawableObject); to add the object to the message
- Use DrawableObject d = (DrawableObject) msg.getContentObject(); to get the object from the message
- The objects themselves "know" how to be drawn on a board
- Easy to extend with more different objects without changing the PantallaAgent

Behaviours

Behaviours used:

- TickerBehaviour: to paint random objects, depending on the commandline arguments
- WakerBehaviour: to have a break and to schedule painting a house
- ThreeStepsBehaviour: to actually paint the house
- CyclicBehaviour: to wait for objects (messages) to paint
- SequentialBehaviour: to create and register the PantallaAgent
- OneShotBehaviour: as subbehaviours of the SequentialBehaviour

Other goals:

- The CyclicBehaviour waiting for messages is blocked when there are no messages
- Used setup() to add behaviours, takeDown() to say goodbye
- Used OnStart() and OnEnd() to say what the PintorAgent is going to paint