

# **Project Demo: Tutorial**

**Group: Gamygdala-Integration**

B.L.L. Kreynen

M. Spanoghe

R.A.N. Starre

Yannick Verhoog

Joost Wooning

June 12, 2015

# 1 Overview

In this section you can find the main goal of this tutorial. This tutorial will mainly be a practice exercise for both the developing group and the group that will use our software. In this exercise you will be making a DEMO of the BW4T environment by learning how to use the integration of Gamygdala in GOAL. Please keep in mind that reading the paper on Gamygdala is recommended. Also, you will need install the project on your computer. That will not be explained in this tutorial. For this, you will have to take a look at the Project Documentation guide.

This guide however, will be used as a feedback report on our documentation and implementations. If something is unclear or not as expected, please make sure you let us know. We will interview you with the "think-out-loud" method. This involves you making the implementations talking about what you see and what goes wrong if the occasion arises.

## 1.1 working environment

Split up in groups of 2 or 3. Each group should then try to get the project working locally. When all this is set up, try to follow the instructions and complete the tutorial.

## 1.2 given prerequisites

- A guide explaining how to get the project running locally.
- We have a GOAL agent file that is already programmed to take the blocks in the environment. The implementation resembles the bot that is made in the first year course on MAS.
- An empty configuration file.
- A MAS2G file that is being prepared to run a single bot.
- Server and client of the environment.
- Documentation for the EmotionConfig.

# 2 Learning objectives

Here you can find the main objectives this tutorial will focus on.

- Running a goal agent with the default values of the EmotionConfig.
- Setting a whitelist of goals that will alter emotions. All the other will be ignored (not important enough to cast emotions)
- Creating a subgoal.
- Adding second agent. This agent will then perform actions based on its emotion.
- Learning the difference between a common and individual goal.
- Creating a relation between the two agents.

NOTE: The objectives will be implemented in this specific order. We think that this will build up the complexity in a way that is comfortable.

## 3 Tutorial

Finally we arrive at the interesting part. The implementation itself. Follow the next steps.

### 3.1 Agent with basic emotions

First of all we will try something very easy. Just try to run the agent with no changes to its .goal file. Also no settings are present in the EmotionConfig text file. First click on the server.jar and then run the mas2g file as an application. See what happens and reflect. What do you see and does the emotions make any sense?

### 3.2 Whitelist

Now, the basic values could work in some cases but are not good overall. Next, we are going to change the EmotionConfig. Look for the EmotionConfig text file and open it. It is empty. First take a look in the Documentation file for this EmotionConfig. Then, try to think about the fact that some goals are important for emotions, but some are not at all. Right now, as you probably found out, achieving goals gives you positive emotions. But what about goals that are not that important. The in/1 goal for example. You might want to ignore some goals regarding emotions. This can be done using the whitelist command. Try setting it on. As you can see in the documentation setting whitelist on is not enough. You will have to define the goals that are whitelisted and thus will cast emotions when achieved or dropped. Look for some goals in the .goal file that are important. Add them to the EmotionConfig. Keep in mind that defining a goal there has to be created with its arity. For example "at" needs to be defined as "at/1". This is so that goals with different parameter counts but with the same name can still be addressed individually. When you are ready, run the agent again. Reflect.

### 3.3 subgoals

When creating an agent subgoals can be used to create structure. For example when you want to pick up a sequence of blocks. Your main goal is to get all the blocks in the sequence. Fixing this problem as an undivided whole is rather impossible. The agent will have to achieve sub-problems. In this case that will be getting the next block. When that is done, the next one, and so on. Getting these individual blocks can thus be considered subgoals of the main goal of getting the complete sequence. Our implementation supports this structure. You can set some goals as a subgoal of some other goals. This can be very interesting since the agent is able to get some specific feelings when using this structure. Keep in mind that you need to define these goals and subgoals in the .goal first. build them in logically. Then try to use these definitions in the EmotionConfig.

### 3.4 Second agent

Now you can copy the .goal file and give it an appropriate name. Also, make sure this second agent runs together with the first agent in BW4T. Now change the code of the second agent so that it performs actions based on his emotions. The emotion are given as percepts to the agent. The code for inserting and deleting these percepts is already working. Try to use the gam("emotion",intensity) to query an emotion. Use this predicate to take or refuse an action. (Think about preconditions of the action specifications)

### 3.5 Common goals

When two agents are grabbing blocks for the sequence they have a common goal to fulfill this task. They both want to complete the task. Some goals on the other hand (think about being at a block, ...) are just individual. Try to setup your goals correctly. It may already be set up perfectly. If not, complete this.

### 3.6 Relation

Think about two agents. They may or may not be friends at all. Try to set a symmetric relation between the two agents by adding some settings in the EmotionConfig. Does this give other emotions? What emotions do you get?

## 4 Summary

If you completed all the tasks, reflect on what you did. See if you have any question and ask them right away.