



Prácticas AIN

Jason-JGOMAS

JADE Game Oriented MultiAgent System

Sesión 1



Índice

- ❖ Introducción
 - ❖ Especificación
 - ❖ Uso
 - ❖ Trabajo a realizar
-



Índice

- ❖ ► **Introducción**
 - ❖ Especificación
 - ❖ Uso
 - ❖ Trabajo a realizar
-



Introducción (I)

- ❖ Desarrollo de un SMA en un entorno cooperativo y competitivo
 - ❖ Objetivo: Mejorar el comportamiento individual y colectivo
 - ❖ Uso de una plataforma de agentes sobre entornos 3D para simulaciones y videojuegos
 - ❖ Básicamente...
 - ❖ Un puñado de agentes repartidos en dos bandos...
 - ❖ ... con unos objetivos que cumplir...
 - ❖ ... integrados en un entorno virtual
-



Introducción (II)

- ❖ Taxonomía de Agentes → Diferentes tipos
 - ❖ Componente Estratégico
 - ❖ Integración en Entorno
 - ❖ Entre Agentes → cooperación
 - ❖ Con el Terreno → dificultad de movimiento
 - ❖ Comunicación → paso de mensajes
-



Índice

- ✧ Introducción
 - ✧ ► **Especificación**
 - ✧ Uso
 - ✧ Trabajo a realizar
-
-



Especificación

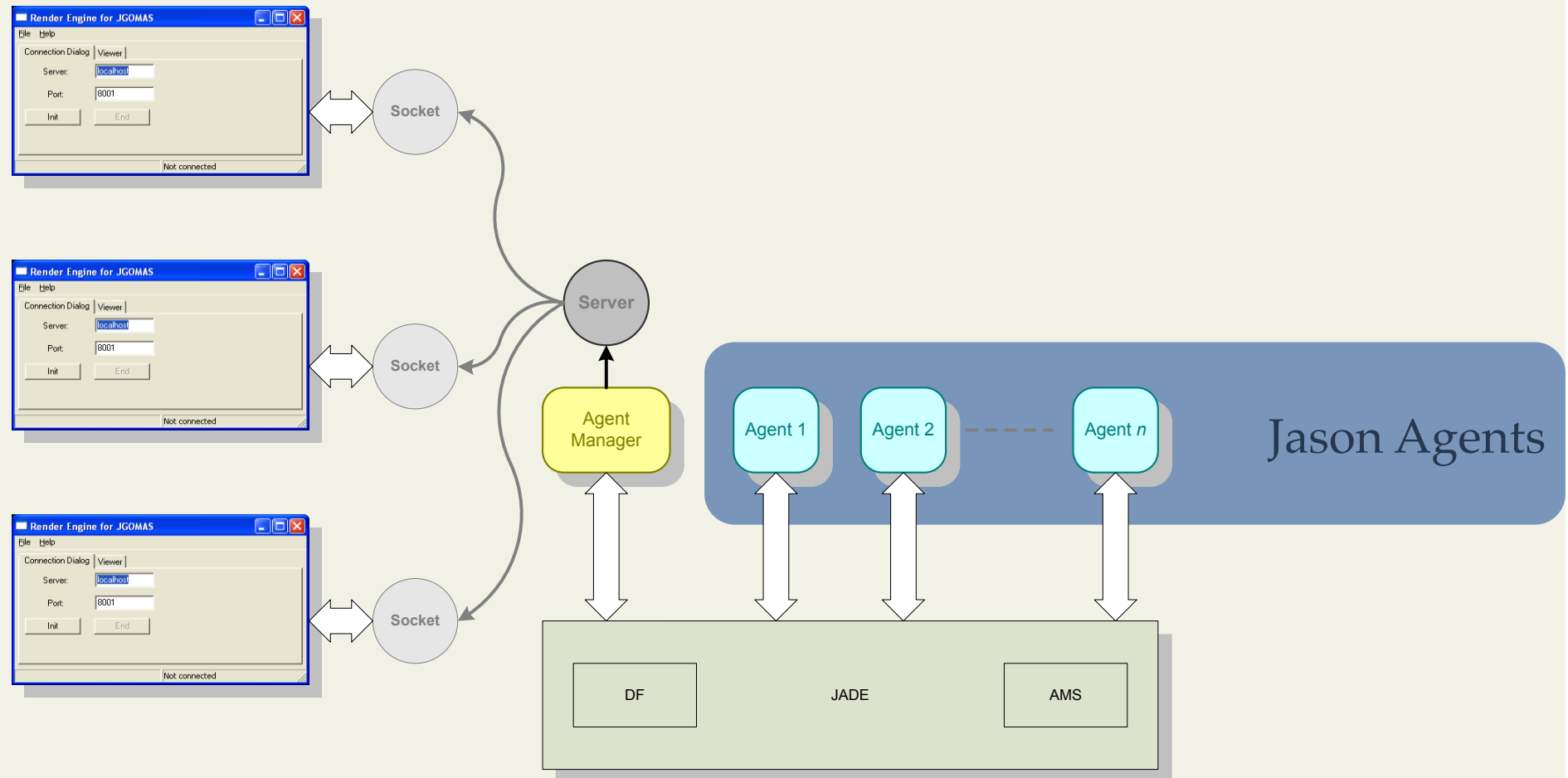
- ❖ Arquitectura
 - ❖ Tareas
 - ❖ Bucle de Ejecución
 - ❖ Interfaz (API)
-
-



Especificación

- ❖ ► **Arquitectura**
 - ❖ Tareas
 - ❖ Bucle de Ejecución
 - ❖ Interfaz (API)
-

Arquitectura (I)



Arquitectura (II)

The image displays a multi-agent system architecture, likely for a simulation environment like JGomas. It consists of several overlapping windows:

- Terminal Window (jgomas.bat):** Shows the execution log of the system. It includes messages from various managers (e.g., Manager[12], Manager[13], Manager[14], Manager[15], Manager[16]) and the Registry, indicating the activation of agents and the registration of services. The log shows the activation of agents like 'Ammo_Allied', 'Backup_Axis', and 'Medic_Axis'.
- JGomas Viewer (Top Left):** A small window showing a 3D view of the simulation environment. It displays a yellow ground with a red flag and several small, stylized robot-like agents.
- JADE Interface (Center):** A window titled 'RMA@pandemonium:1099/JADE - JADE'. It features a menu bar (File, Actions, Tools, Remote Platforms, H) and a tree view of the system's structure. The tree view shows a hierarchy starting with 'AgentPlatforms', followed by '"pandemonium:1099/JADE"', and then 'Main-Container'. Below this, a list of agents is displayed, including A3@pandemonium, E9@pandemonium, RMA@pandemonium, ams@pandemonium, A6@pandemonium, E6@pandemonium, E5@pandemonium, ObjectivePack, A5@pandemonium, A1@pandemonium, A4@pandemonium, A2@pandemonium, df@pandemonium, Manager@pandemonium, MedicPack_51, and MedicPack_71.
- JGomas Viewer (Bottom Right):** A larger window showing a 3D view of the simulation environment. It displays a yellow ground with a red flag and several small, stylized robot-like agents. The environment appears to be a maze-like structure with yellow walls and a red flag.

Arquitectura (III) Mapas

- ❖ Una partida se desarrolla en un entorno virtual (mundo virtual) de 256×256

- ❖ Posición de los agentes:

$x [0..255], z [0..255]$

- ❖ Cada agente tiene acceso parcial al mapa donde se desarrolla la partida:

- ❖ Sabe su posición (puede no ser la real)

- ❖ Ve objetos cuando está a cierta distancia
-



Especificación

- ❖ Arquitectura
 - ❖ ► **Tareas**
 - ❖ Bucle de Ejecución
 - ❖ Interfaz (API)
-

Especificación Tareas (I)

- ❖ Definición

task(Priority, TaskType, Agent, Position, Content)

- ❖ Priority: Prioridad de la tarea
 - ❖ TaskType: Tipo de tarea (siguiente transparencia)
 - ❖ Agent: agente asociado a la tarea
 - ❖ Position: Posición donde llevar a cabo la tarea
 - ❖ Content: Posible contenido adicional
-

Especificación Tareas (II)

- ❖ Algunos de los tipos de tareas son:
 - ❖ TASK_GIVE_MEDICPAKS
 - ❖ TASK_GIVE_AMMOPAKS
 - ❖ TASK_GIVE_BACKUP
 - ❖ TASK_GET_OBJECTIVE
 - ❖ TASK_GOTO_POSITION
 - ❖ ...
-

Especificación Tareas (III)

- ❖ **Priority** indica la prioridad de la tarea
 - ❖ Se lanza siempre la tarea de prioridad **más alta**
 - ❖ Es posible **redefinir la prioridad** de cada tipo de tarea
 - ❖ Las tareas las pone en ejecución el sistema, no el usuario
-

Especificación Tareas (IV)

- ❖ Declaración:

- ❖ Se utiliza un plan: **add_task**

!add_task(task(TaskPriority, TaskType, Agent, Position, Content))

Ejemplos (dos opciones):

- ❖ **!add_task(task(1000, "TASK_GET_OBJECTIVE", M, pos(ObjectiveX, ObjectiveY, ObjectiveZ), ""));**
- ❖ **!add_task(task("TASK_GET_OBJECTIVE", M, pos(ObjectiveX, ObjectiveY, ObjectiveZ), ""));**

Dichos objetivos disparan el plan que crea la tarea, el segundo le asigna la prioridad definida por el agente



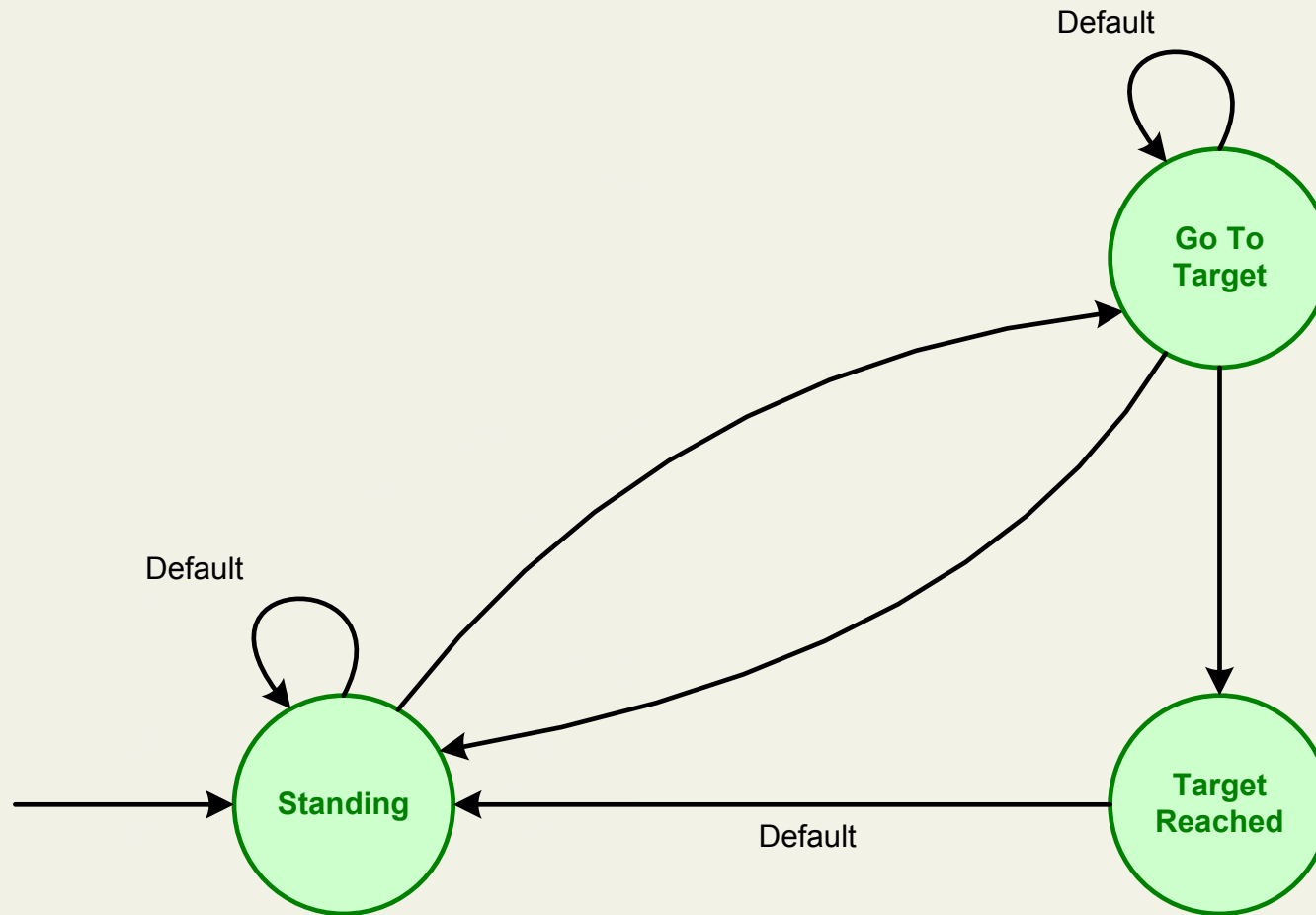
Especificación

- ❖ Arquitectura
 - ❖ Tareas
 - ❖ ► **Bucle de Ejecución**
 - ❖ Interfaz (API)
-

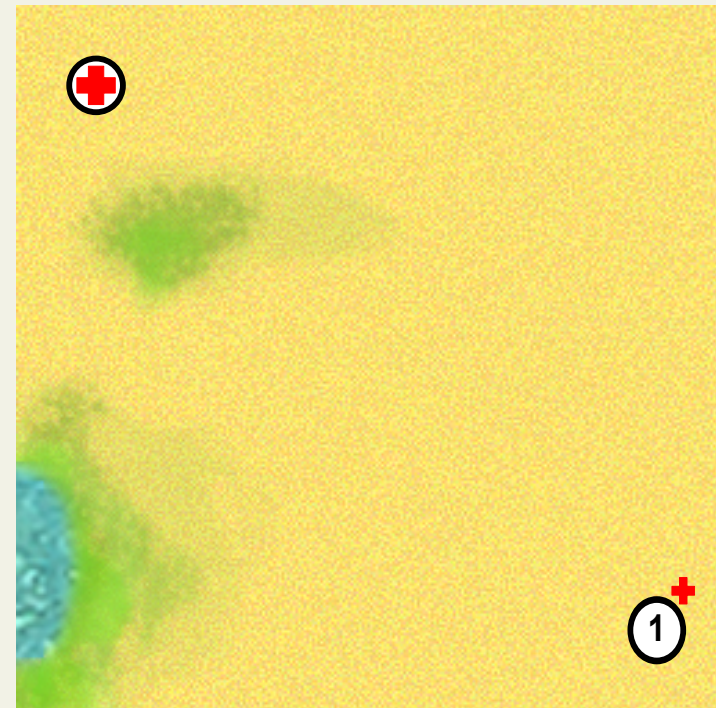
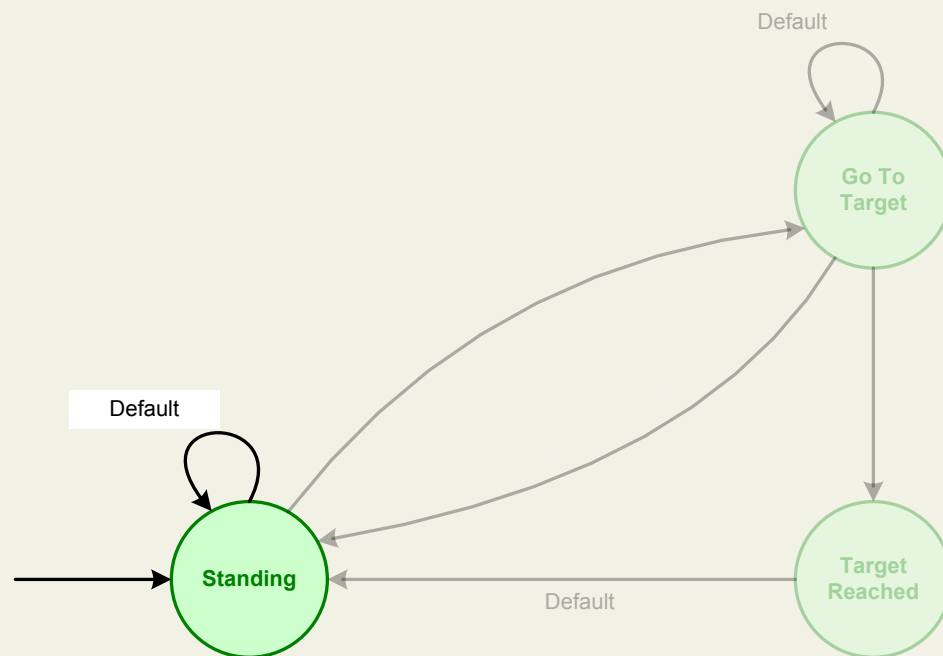
Bucle de Ejecución (I)

- ❖ Cada agente ejecuta una FSM (4 estados):
 - ❖ STANDING
 - ❖ GO_TO_TARGET
 - ❖ TARGET_REACHED
 - ❖ QUIT
 - ❖ FSM se utiliza para realizar tareas:
 - ❖ Inicio (Lanzamiento)
 - ❖ Desarrollo (Ejecución)
 - ❖ Final (Acción y Destrucción)
 - ❖ Se lanza siempre la tarea de prioridad más alta
-

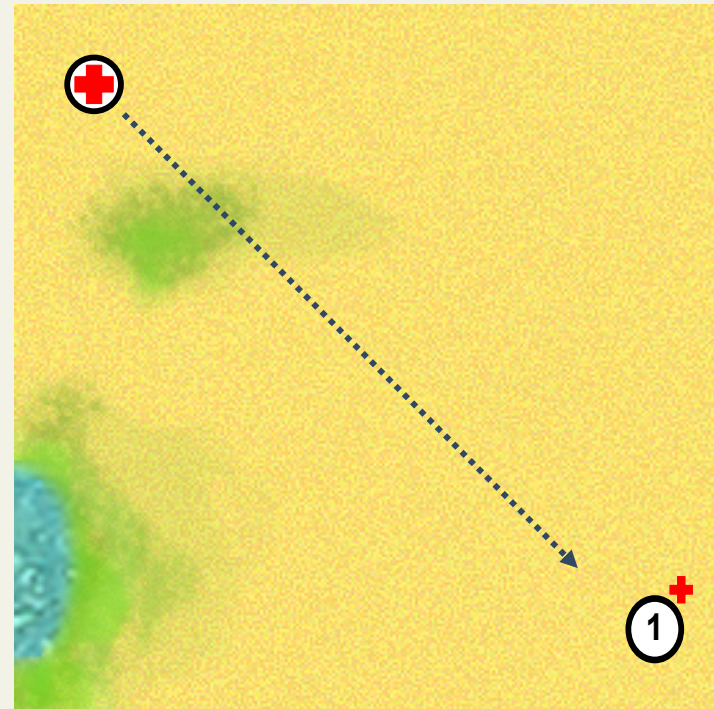
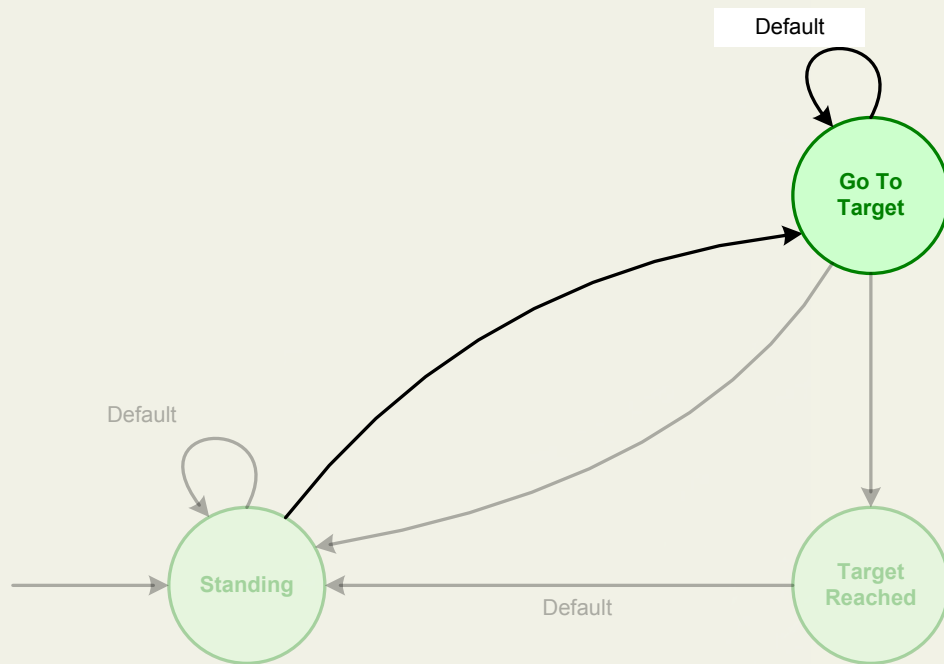
Bucle de Ejecución (II)



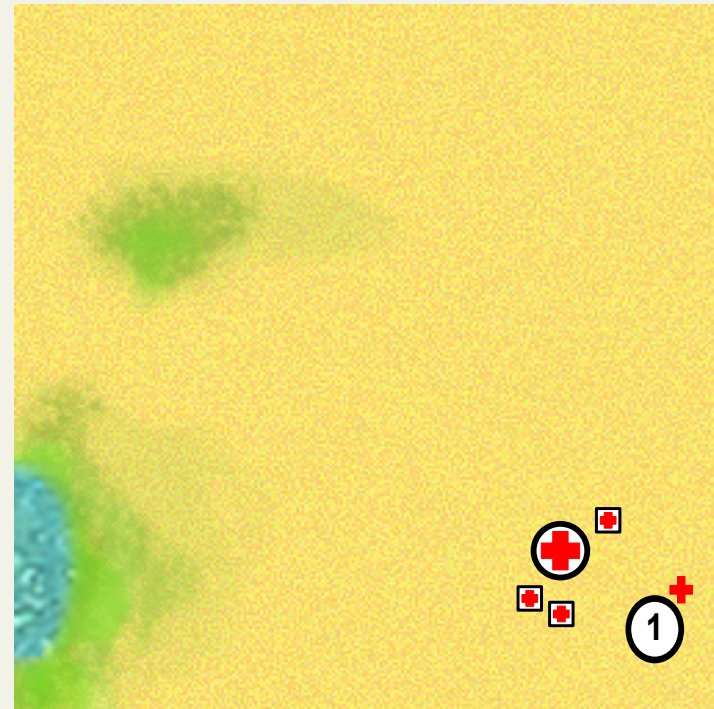
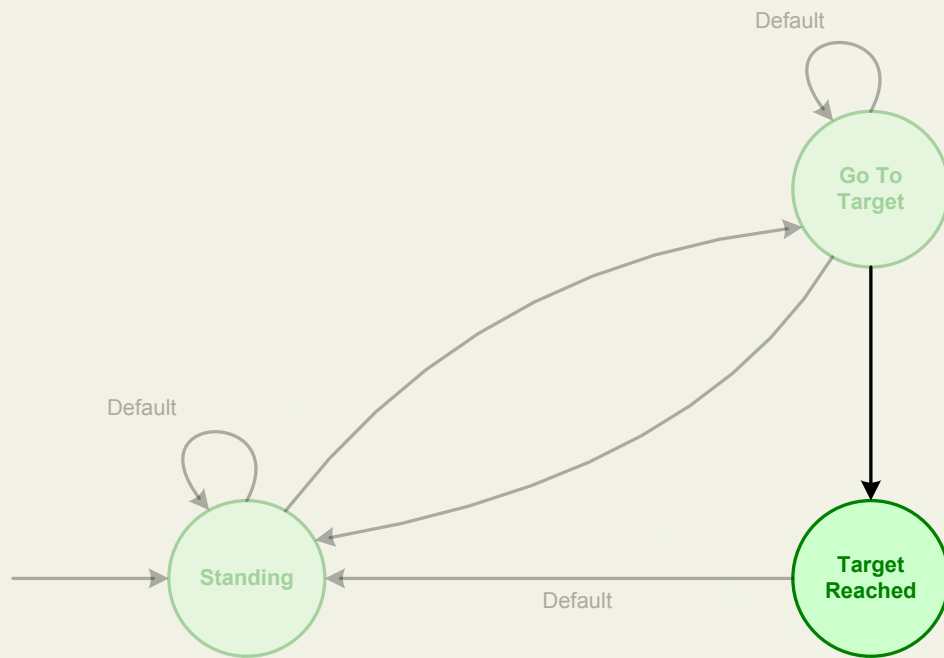
Ejemplo (I)



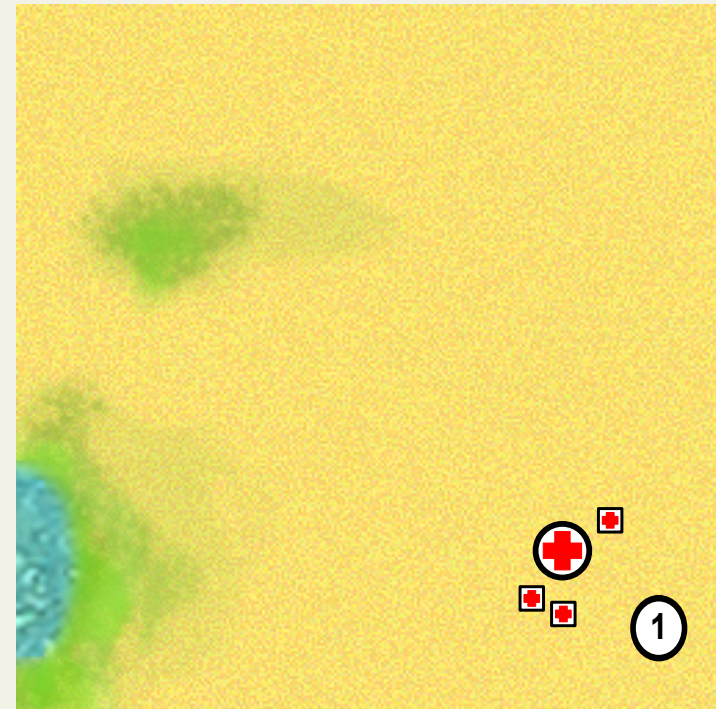
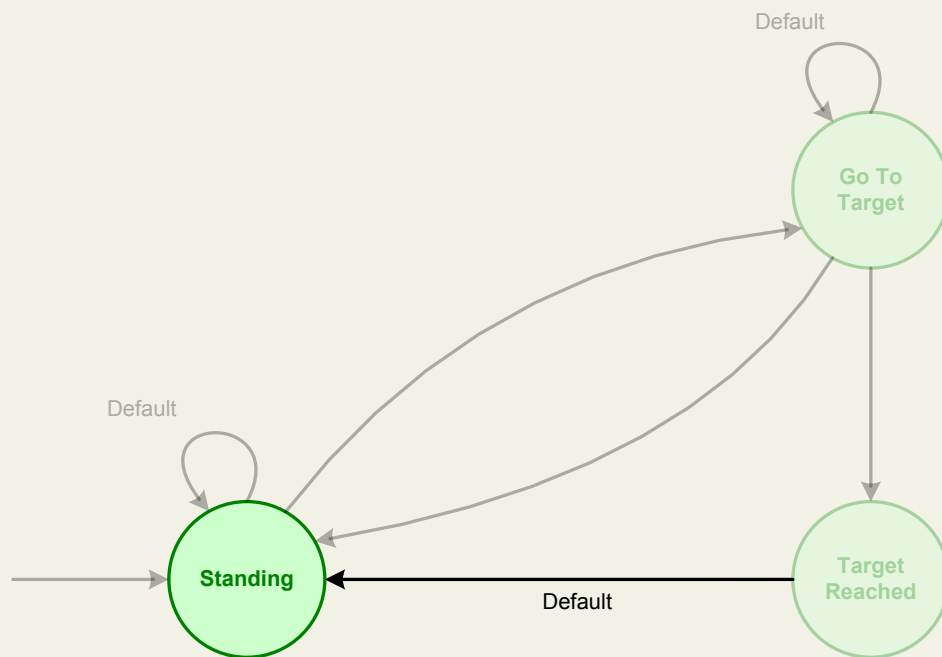
Ejemplo (II)



Ejemplo (III)



Ejemplo (IV)





Especificación

- ❖ Arquitectura
 - ❖ Tareas
 - ❖ Bucle de Ejecución
 - ❖ ► **Interfaz (API)**
-



Interfaz

jgommas.asl → Fichero con el comportamiento no modificable del agente

jasonAgent_TEAM_TYPE.asl → 6 ficheros para distintos agentes

jasonAgent_ALLIED.asl

jasonAgent_ALLIED_MEDIC.asl

jasonAgent_ALLIED_FIELDOPS.asl

jasonAgent_AXIS.asl

jasonAgent_AXIS_MEDIC.asl

jasonAgent_AXIS_FIELDOPS.asl

Creencias

- ❖ Las principales son:
 - ❖ `tasks(task_list)`
 - ❖ `fovObjects(object_list)`
 - ❖ `state(Estado_actual)`
 - ❖ `my_health(X)`
 - ❖ `my_ammo(X)`
 - ❖ `my_position(X,Y,Z)`
-

Planes disponibles sobrecargables

- ✧ `!init`
 - ✧ `!perform_look_action`
 - ✧ `!perform_aim_action`

 - ✧ `!get_agent_to_aim`

 - ✧ `!perform_no_ammo_action`
 - ✧ `!perform_injury_action`
 - ✧ `!performThresholdAction`

 - ✧ `!setup_priorities`
 - ✧ `!update_targets`
-



Índice

- ❖ Introducción
 - ❖ Especificación
 - ❖ ► **Uso**
 - ❖ Trabajo a realizar
-
-

Uso: MAS

- ❖ Ejecución en dos pasos:

1º paso → Lanzar el manager

- ❖ Línea de comandos:

```
java -classpath "lib\jade.jar;lib\jadeTools.jar;lib\n\nBase64.jar;lib\http.jar;lib\iiop.jar;lib\n\nbeangenerator.jar;lib\jgomas.jar;student.jar;lib\n\njason.jar;lib\JasonJGomas.jar;classes;." jade.Boot -gui\n\n"Manager:es.upv.dsic.gti_ia.jgomas.Cmanager(4, map_04, 125,\n10) "
```

Manager:es.up...CManager(4,map_04,125,10)

Nombre
Agente

Clase

Parámetros:

4: nº de agentes
map_04: nombre del mapa
125: refresco en ms.
10: duración (minutos)

Uso: MAS

2º Paso → Lanzar los agentes

✦ Línea de comandos:

```
java -classpath "lib\jade.jar;lib\jadeTools.jar;lib\Base64.jar;lib\http.jar;lib\iiop.jar;lib\beangenerator.jar;lib\jgomas.jar;student.jar;lib\jason.jar;lib\JasonJGomas.jar;classes;."
jade.Boot -container -host localhost
"T1:es.upv.dsic.gti_ia.JasonJGomas.BasicTroopJasonArch(jasonAgent_AXIS.asl);T2:es.upv.dsic.gti_ia.JasonJGomas.BasicTroopJasonArch(jasonAgent_AXIS_MEDIC.asl);A1:es.upv.dsic.gti_ia.JasonJGomas.BasicTroopJasonArch(jasonAgent_ALLIED_FIELDOPS.asl);A2:es.upv.dsic.gti_ia.JasonJGomas.BasicTroopJasonArch(jasonAgent_ALLIED.asl)"
```

T1:es.up... BasicTroopJasonArch(jasonAgent_AXIS.asl)

↑
Nombre
Agente

↑
Clase

↑
Parámetros:
Fichero asl con el código

Uso: Render Engine

- ❖ Línea de comandos:
- ❖ Lanzamos el fichero batch

```
c:\> run_jgomasrender.bat
```

- ❖ Cuyo contenido es

```
set OSG_FILE_PATH=../../../../../data  
JGOMAS_Render.exe --server <hostname> --port <integer>
```



Índice

- ❖ Introducción
 - ❖ Especificación
 - ❖ Uso
 - ❖ ► **Trabajo a realizar**
-
-



Prueba

Realizar distintas ejecuciones con distintos números de agentes, clases y mapas.

- Equipo de 2 contra 2 del tipo soldado
 - Equipos de 8 aliados y 4 del eje: 1 médico y 1 fieldops en cada bando
 - Equipos de 4 aliados y 8 del eje: 1 médico y 1 fieldops en cada bando
 - Probad los mapas map_01, map02, map_03, map_04
 - Modificar la duración de la partida
 - La creencia debug(X) permite cambiar la verbosidad del agente (su valor es entre 1 y 3), probad a cambiar la verbosidad, modificando la creencia en alguno de los ficheros de los agentes
-

Trabajo a Realizar (ahora y en casa)

1. Leerse el documento “JGomas Manual Jason.doc”
2. Mirar con detalle el código de los ficheros “.asl”

Sobretudo:

- ❖ jgomas.asl (no se puede modificar)
 - ❖ jasonAgent_ALLIED.asl
 - ❖ jasonAgent_AXIS.asl
-

Trabajo a Realizar (próximas sesiones)

- ❖ Objetivo (en dos sesiones):
 - ❖ Implementar un agente **atacante** (allied) que recorra las esquinas del mapa.
 - ❖ Implementar un agente **defensor** (axis) que recorra las esquinas del mapa.
 - ❖ Implementar un agente **atacante** (allied) que localice un compañero y lo vaya siguiendo.
 - ❖ Implementar un agente **defensor** (axis) que localice un compañero y lo vaya siguiendo.
-