# Prácticas AIN

# Jason-JGOMAS

## JADE Game Oriented MultiAgent System

### Práctica final: Comunicación y Coordinación

# Índice

- Taxonomía (a modo de recordatorio)
- Registro de Servicios
- Comunicación y Coordinación
- Trabajo a realizar

# Taxonomía (III)

- Agentes Internos (en Jade):
  - ***Manager***: coordina todo el juego. Actúa de interfaz con los visualizadores gráficos.
  - ***Pack***: paquetes de medicina, munición y objetivo.
- Agentes Externos (en Jason):
  - ***Troop***: agentes de usuario (*medic*, *fieldops* y *soldier*). Disponen de un comportamiento básico, que el usuario puede mejorar.

# Taxonomía (I)

- Hay definidos tres tipos de roles en los **agentes externos**:
  - ***Soldier***: soldado de tipo general
    - ALLIED: va a por la bandera y vuelve a la base
    - AXIS: patrulla alrededor de la bandera
  - ***Medic***: acude a curar
  - ***FieldOps***: acude a dar munición
- Un agente asume un único rol durante toda la partida
- Cada rol tiene unas características y ofrece unos determinados **servicios**

# Registro de servicios (I)

- Un rol debe registrar un **servicio** para que el resto de roles puedan solicitarlo:

  **.register( "JGOMAS", "type")**

- Se utiliza para registrar en el DF un servicio del tipo “type” por parte del agente que lo ejecuta.
- Ej:

  *.register( "JGOMAS", "medic_AXIS");*

  registra en el DF el servicio “medic_AXIS”.

# Registro de servicios (II)

Registros que se hacen por defecto en todos los agentes:

- ALLIED

  *.register( "TEAM", "ALLIED");*

  Soldado: *.register( "JGOMAS", "backup_ALLIED");*

  Médico: *.register( "JGOMAS", "medic_ALLIED");*

  Fieldops: *.register( "JGOMAS", "fieldops_ALLIED");*

- AXIS

  *.register( "TEAM", "AXIS");*

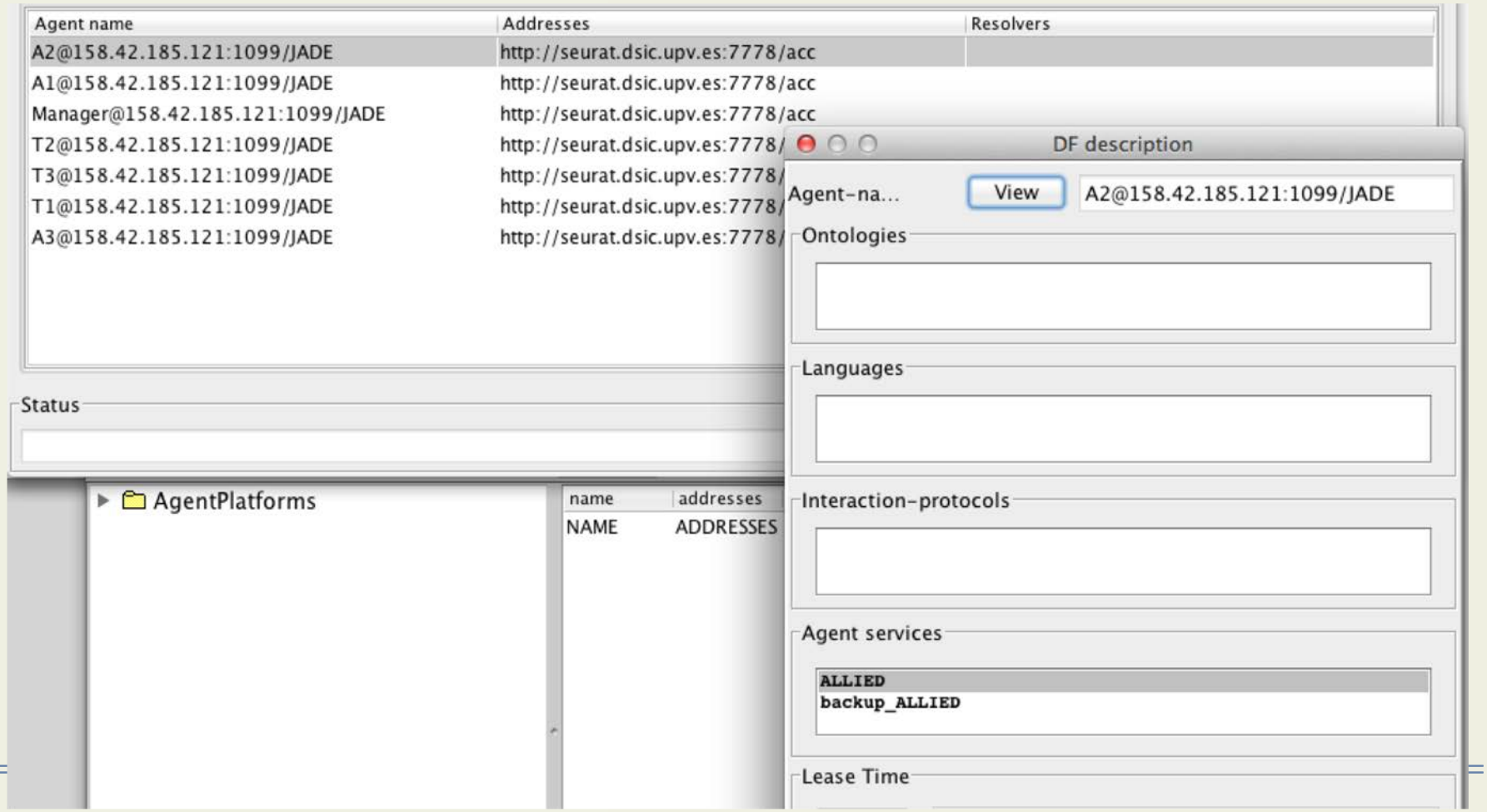
  Soldado: *.register( "JGOMAS", "backup_AXIS");*

  Médico: *.register( "JGOMAS", "medic_AXIS");*

  Fieldops: *.register( "JGOMAS", "fieldops_AXIS");*

# Registro de servicios (III)

Seleccionando la opción “Tools / Show DF GUI” se pueden ver los servicios ofertados en el GUI de JADE



# Registro de servicios (IV)

- ¿Cómo saber que servicios hay disponibles desde un agente?

  Utilizar la acción interna **.my_team(type,list)**

  Esta acción devuelve la lista de agentes de **tu equipo** disponibles a través de las páginas amarillas (DF) que son del tipo "type".

  Ej: *.my_team("medic_AXIS", E)* ejecutado por un agente del equipo "AXIS" devuelve la lista E que contiene los médicos de mi equipo.

  Entonces ¿cómo saber el nombre de los agentes de mi equipo?

  *.my_team("AXIS", E);* ó

  *.my_team("ALLIED", E);*

# Registro de servicios (V)

- Ejemplo de uso

…

*.my_team("AXIS", E1);*

*.my_name(Me);*

*.println("Mi equipo es: ", E1, " y yo soy: ", Me );*

*.length(E1, X);*

*if (X==0){ .println("Me he quedado sólo"); }*

# Coordinación (I)

- JGOMAS dispone de mecanismos que permiten la coordinación entre agentes:
  - Sin comunicación (implícita):
    - Sensorización del entorno (ya visto en la práctica 1)
  - Con comunicación (explicita):
    - Mediante paso de mensajes

# Coordinación (II)

- Con comunicación (**CAMBIA RESPECTO A LO VISTO!!!!!!**)
  - Envío de mensajes mediante la acción interna

    *.send_msg_with_conversation_id (Rec, Perf, Cont, ConvId)*

    Donde:

    Rec → receptor del mensaje (puede ser una lista)

    Perf → performativa (tell, untell, achieve, …)

    Cont → contenido

    ConvId → Id de conversación (se usa en Jade)

# Coordinación (III)

Ej: A1 quiere enviar un mensaje a su equipo diciendo que vayan a su posición (para ayudar, para coordinarse, para reagruparse, …)

…

*?my_position(X,Y,Z);*

*.my_team("AXIS", E1);*

*.concat("goto(",X, ", ", Y, ", ", Z, ")", Content1);*

*.send_msg_with_conversation_id(E1, tell, Content1, "INT");*

NOTA: "INT" es un identificador de conversación inventado, **recomiendo su uso**

# Coordinación (IV)

Ej: el resto de agentes del equipo dispondrían de un plan de la forma:

*+goto(X,Y,Z)[source(A)]*

*<-*

*.println("Recibido un mensaje de tipo goto de ", A);*

*.*

# Coordinación (V)

Ej: Si queremos que los agentes hagan algo más sofisticado

```
+goto(X,Y,Z)[source(A)]
<-
   .println("Recibido mensaje goto de ", A);
   !add_task(task("TASK_GOTO_POSITION", A, pos(X, Y, Z), ""));
   -+state(standing);
   -goto(_,_,_).
```

Mejoras:

- Comprobar si A tiene autoridad sobre mi
- Hacer caso sólo si tengo salud, armamento o las dos cosas
- Revisar antes mis tareas pendientes

# Coordinación (VI)

- Estrategias vistas (o por ver) en clase:
  - Organización jerárquica: El jefe manda !!!
  - Contract Net: Delegación de tareas
  - Social Choice: Votamos !!!
  - Subastas: quien me ofrece algo mejor !!!
  - …

# Trabajo a Realizar (próximas sesiones)

- Objetivo:
  - Diseñar e implementar un equipo de 8 agentes con la distribución de tipos que deseéis (médicos, soldados y fieldops) para jugar a **capturar la bandera** en un mapa cualquiera como **atacante o como defensor**, de manera que **ganen en cualquier situación a los equipos suministrados**.
  - Es **necesario** emplear alguna técnica de **coordinación vía paso de mensajes** entre agentes del mismo equipo.
  - Se debe incluir al menos un **servicio nuevo por parte de un agente** y el uso del mismo por parte de otros agentes
  - Se deben realizar mejoras de **comportamientos** existentes (por ej. tratar de evitar el fuego amigo)

# Trabajo a Realizar (próximas sesiones)

Posibles estrategias

- ALLIED
  - Elegir un capitán que coordine el ataque del resto
  - Dividir el equipo en dos y atacar por oleadas
  - Coordinar la retirada cuando se tiene la bandera
- AXIS
  - Elegir un capitán que coordine la defensa
  - Coordinar a los agentes para patrullar con distintos radios
  - Añadir algún agente vigía
  - Identificar que la bandera ha sido capturada y buscarla

# Trabajo a Realizar (Normas)

- **Reglas Básicas**:
  - No se puede consultar/solicitar información del sistema sobre el bando contrario que no sea suministrada por el entorno.
  - No puede existir comunicación entre agentes que no sea usando la acción interna *.send_with_conversation_id* y de acuerdo a la especificación proporcionada.

# Trabajo a Realizar (Entrega)

- Entrega:
  - Ficheros *.asl desarrollados. El código, comentado y documentado debe seguir unas mínimas normas de estilo:
    - Tabulado y comentado.
  - Comprimir todo el directorio en un fichero <nombre_equipo>.zip
  - Pequeña memoria, indicando las principales ideas de mejora aplicadas al equipo, así como unas breves conclusiones sobre los resultados obtenidos.