

Práctica 7:

Agentes

JADE

**Fernando Candelario
Herrero**

Gonzalo Sanz Rodríguez

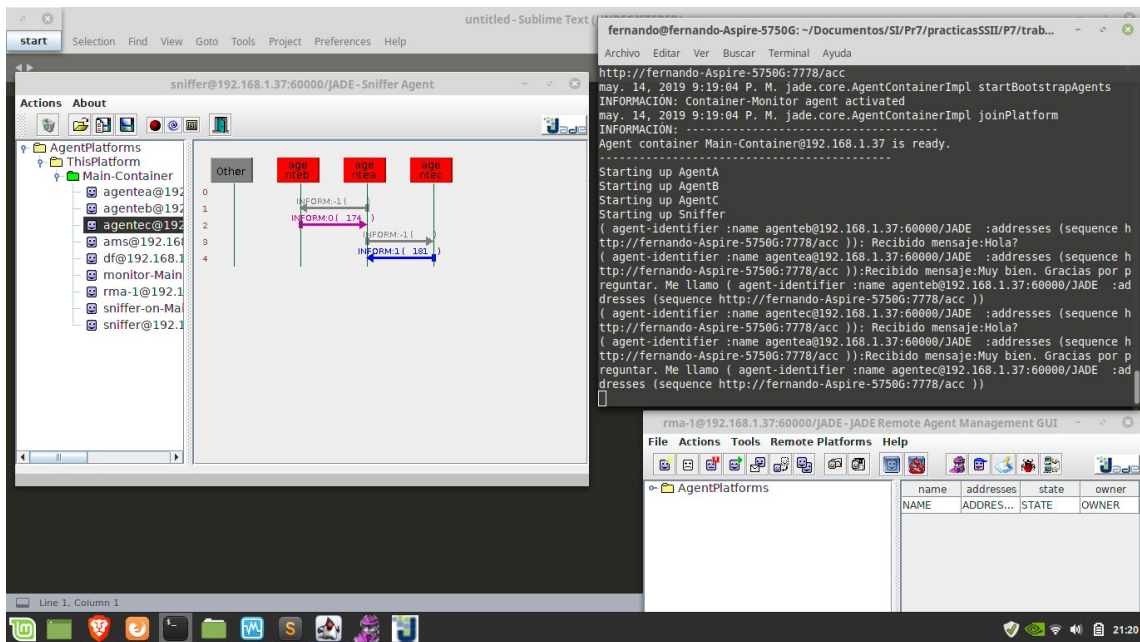
Introducción:

La práctica está basada en 4 ejemplos que hay que ejecutar y aquí adjuntamos los resultados obtenidos:

- **Ej. 1 Hello Word!:** muestra un bucle infinito mostrándonos el mensaje Hello Word, hasta que procedamos a detenerlo.
- **Ej. 2 Migración de los agentes:** En este ejemplo podremos migrar los agentes de una consola a otra, a través de una GUI
- **Ej. 3 Comunicación entre agentes:** En este ejemplo podremos realizar un diálogo entre agentes arrancando la GUI siguiendo una secuencia de acciones que nos permitirán realizar la comunicación.
- **Ej. 4 Implantación de un dialogo a través de Sniffer:**
Creando 2 agentes arrancamos la GUI y estableceremos una conversación entre agentes, los mensajes enviados se plasmarán a través del Sniffer lo que nos mostrará el siguiente diagrama.

Preparación del Trabajo:

- **Versión 1:** Crear los 3 agentes necesarios (A, B y C), sin utilizar todavía la movilidad y hacer que dialoguen entre ellos, primero A con B y por último B con C.



- Versión 2:** Modificando el código realizado hasta ahora incluir la movilidad y poder arrancar el Agente B y C en otra consola. Para la implementación de esta versión hemos desarrollado un `secuencialBehaviour` que se como de la siguiente secuencia de acciones:
 - One Shot Behaviour, migrar al contenedor 1 con el uso de `"doMove()"`.
 - Send Message Behaviour, enviar mensaje al agente B.
 - Recive Message Behaviour, recibir un mensaje del agente B.
 - One Shot Behaviour, migrar al contenedor 2 con el uso de `"doMove()"`.
 - Send Message Behaviour, enviar un mensaje al agente C.
 - Recive Message Behaviour, recibir mensaje del agente C.
 - One Shot Behaviour, migrar Main Container con el uso de `"doMove()"`.

Esta secuencia de acciones se realiza mediante el uso de boton.

Preguntas

1. ¿Qué utilidad le ves a la movilidad y qué inconvenientes?

La movilidad nos permite establecer comunicaciones entre agentes de una manera rápida y eficaz, estableciendo nuestros agentes en puertos a través de los cuales realizar el diálogo. Los principales inconvenientes son la organización en Containers y que la comunicación debe de realizarse desde el mismo puerto lo que dificultará la comunicación entre varios usuarios.

2. ¿Los agentes necesitan estar en la misma máquina para interactuar?

No de hecho hemos podido comprobar que es posible la comunicación de agentes en varias máquinas, pero es necesario realizar el mismo procedimiento en cada una de estas máquinas y establecer los mismos parámetros para que la comunicación sea posible.

3. Dado un agente que se encuentre en una plataforma y con el que se quiera interactuar, ¿qué hace falta para conseguirlo?

Establecer el mismo puerto, conocer la dirección IP del agente objetivo para que de esta manera la capa de transporte esté bien configurada y desde este punto migrar el agente.

4. Un agente mío, ¿tiene que interactuar siempre con los mismos agentes?

No tiene porque, ya que siendo nosotros quien definimos su receptor podremos, pasándole su identificador, establecer cualquier otro agente con el que se comuniquen. Siempre y cuando sepamos identificador del agente receptor.

Lo aprendido:

- **Gonzalo:** Me ha resultado muy interesante el proceso de migración y comunicación entre agentes. También cómo funciona la comunicación como diagramas de secuencia y como después de haber solicitado información se bloquean hasta que reciben la información. Asignando a los agentes comportamientos para que realicen cuantas acciones como necesitemos .
- **Fernando:** He aprendido que el diseño de IA está más automatizado de lo que pensaba. Creía que eso del desarrollo de algoritmos de inteligencia artificial era algo que desarrollaban un grupo de ingenieros y que cada equipo tenía que empezar de “cero”. Con el uso del framework JADE permite tener un esqueleto común para el desarrollo, ejemplos son la infraestructura de comunicación entre agentes, la posibilidad de migración, son acciones muy recurridas en entornos de IA. El desarrollo con Ingenias también me ha sorprendido, pensar en las entidades abstractas del problema y representar cómo se relacionan en forma de diagrama y de ahí poder generar gran parte del código para implementarlo.