

MaDKit

The Multiagent Development Kit

Kevin García Santos

MaDKit

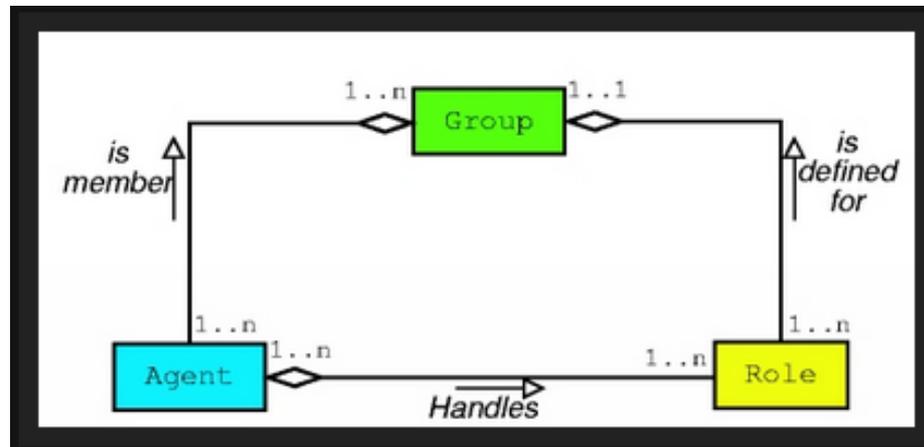
- * MaDKit es un plataforma para la creación de sistemas multiagente implementada en Java.
- * Desarrollada por el Laboratoire d'Informatique de Robotique et de Microélectronique de Montpellier (LIRMM).
- * Página oficial: <http://www.madkit.org/>

Características

- * Soporta varios lenguajes de programación: Java, Scheme (Kawa), Jess y BeanShell.
- * Licencia de software GPL/LGPL (Software Libre).
- * Se ejecuta sobre la Máquina Virtual de JAVA (JVM), los agentes se pueden ejecutar y comunicar entre sí.
- * Mecanismo de comunicación P2P (Punto a Punto).
- * No sigue el modelo FIPA.

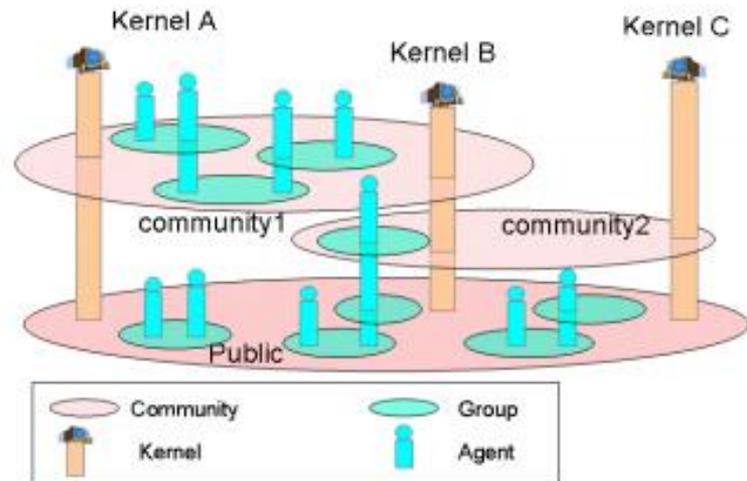
Características

- * Modelo de organización: Agente, Grupo y Rol (AGR), los agentes juegan un papel en grupos y el registro de servicios es diferente, un servicio se especifica como un rol.



Arquitectura

- * Se basa en dividir su estructura en pequeños núcleos llamados Kernels o Micro-kernels. Cada uno puede actuar de forma individual, ofreciendo la mayoría de servicios a los diferentes agentes registrados, sin embargo para obtener una plataforma distribuida se necesitan mas de un Kernel.

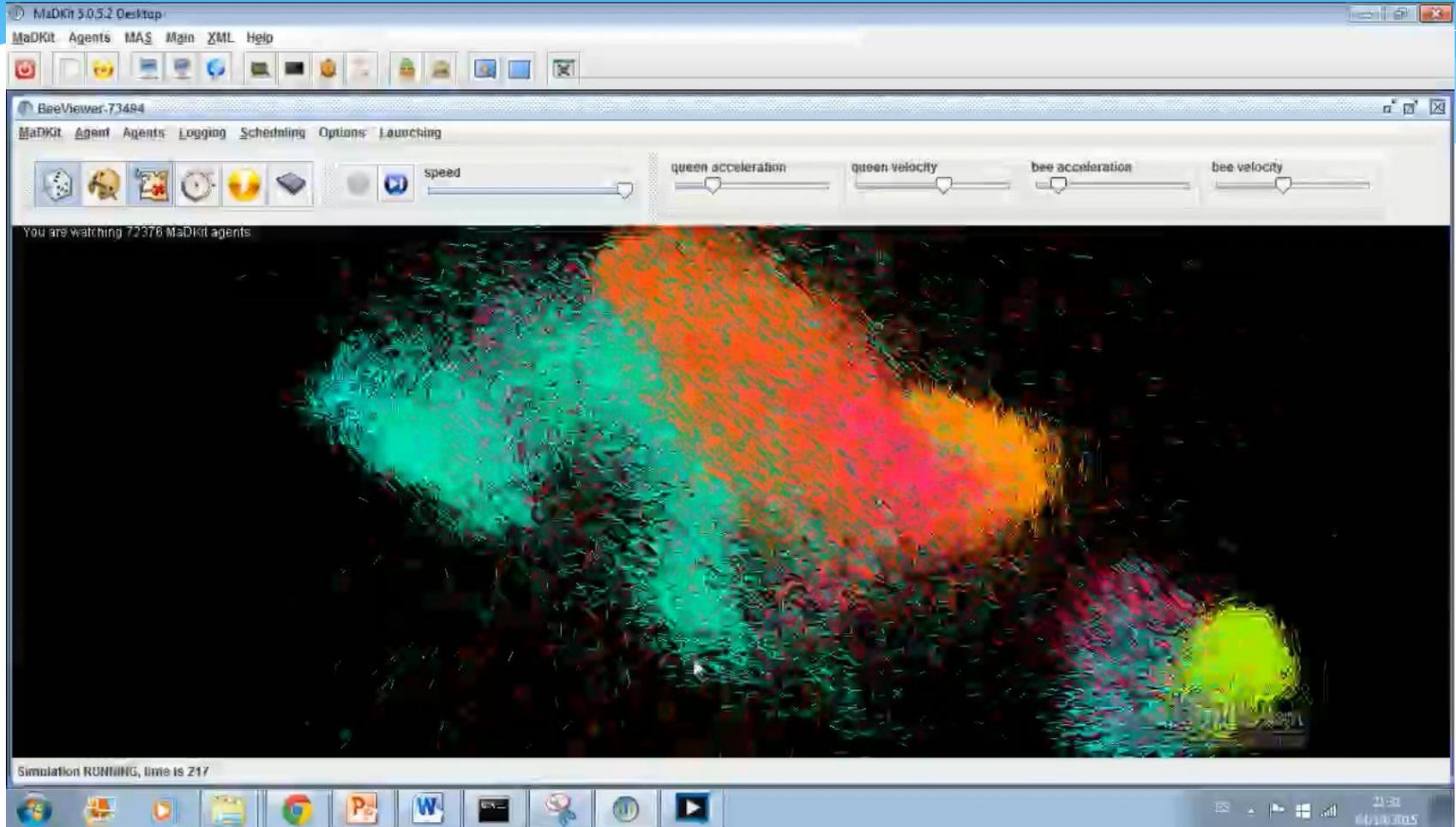


Instalación

- * Sencilla.
- * Tutoriales online disponibles.
- * Se necesita tener instalado Java y tener actualizadas las variables de entorno.
- * Se puede añadir a eclipse fácilmente.
- * Tutorial:
<https://www.youtube.com/watch?v=g3RhSHcEfi8>

	MaDKit	JADE
Arquitectura	AGR	JVM RMI
Lenguajes	Java Scheme (Kawa) Jess BeanShell	Java Jess

Demo



Demo Eclipse

The screenshot displays the Eclipse IDE interface. The main editor shows the source code for `HelloWorldAgent.java`. The code includes a `main` method that calls `Madkit.main` with specific arguments. A small dialog window titled "HelloWorldAgent-2" is open, showing "MaDKit Agent Logging Help" and the output "INFORMACIÓN: Hello World !!". Below the dialog, the text "I am activating..." is visible. The console window at the bottom shows the execution path and the output of the agent: "[HelloWorldAgent-2] INFORMACIÓN : Hello World !!" and "I am activating...". The Package Explorer on the left shows the project structure, and the Outline view on the right shows the class hierarchy. The system tray at the bottom indicates the date and time as 22:27 on 04/10/2015.

```
Java - Prueba/src/helloworld/ex01/HelloWorldAgent.java - Eclipse
File Edit Source Refactor Navigate Search Project Run Window Help

Package Explorer
Prueba
├── helloworld.ex01
│   ├── HelloWorldAgent.j
│   └── HelloWorldAge
│       ├── main(String
│       ├── activate(): \
│       ├── end(): void
│       └── live(): void
├── JRE System Library [JavaSE
└── Referenced Libraries

HelloWorldAgent.java
}
/**
 * Now we have to start this agent one way or another.
 * Implementing a main method is not required at all but it could be used
 * to simulate a command line call to MaDKit with the desired options inside
 * an IDE. Such a command line
 * > java madkit.kernel.MaDK
 * @param args unused here
 */
public static void main(String
String[] args2 = { "--laur
"helloworld.ex01.H
Madkit.main(args2);
}
/**
 * What we used here is exactl
 * String[] argss = {Madkit.Opt
 * Madkit.main(argss);
 * which in turn is exactly th

HelloWorldAgent-2
MaDKit Agent Logging Help
INFORMACIÓN: Hello World !!
I am activating...
clear

Problems @ Javadoc Declaration Console
HelloWorldAgent [Java Application] C:\Program Files\Java\jre1.8.0_60\bin\javaw.exe (4 de oct. de 2015 22:27:13)
[HelloWorldAgent-2] INFORMACIÓN : Hello World !!
I am activating...

www.XSplit.com
POWERING BROADCASTING

Writable Smart Insert 107:32
ES 22:27 04/10/2015
```

Fin

¿Preguntas?

