

Modelado de un sistema multi-agente para el monitoreo de residuos peligrosos en la industria manufacturera

Carlos A. Soto¹, Juan P. Soto², Adrián Vázquez², Elvira Rolón³, Julio C. Rolón³

¹ Instituto Tecnológico de Sonora Unidad Guaymas, Guaymas, Sonora, México

² Universidad de Sonora, Hermosillo, Sonora, México

³ Universidad Autónoma de Tamaulipas Centro Universitario Tampico-Madero, Facultad de Ingeniería "Arturo Narro Siller", Tampico, Tamaulipas, México

csoto@iswug.net, {juanpablo.soto, adrian.vazquez}@unison.mx, {erolon, jrolon}@docentes.uat.edu.mx

Resumen. En la actualidad, las tecnologías de la información y la comunicación se han convertido en una parte inevitable de la planificación, diseño y la gestión de procesos en el sector industrial. El uso de sistemas multi-agente para gestionar el manejo de residuos peligrosos facilita a las compañías de manufactura una mejor planificación, monitoreo y recolección de dichos residuos. Los agentes software han demostrado ser una solución eficiente, ya que pueden desarrollar tareas en nombre de los usuarios. Además, los agentes pueden comunicarse entre ellos y utilizar técnicas de razonamiento para la solución de conflictos. Por esta razón, en este trabajo se propone el uso de agentes software para el monitoreo de residuos peligrosos en la industria manufacturera.

Palabras clave: sistema multi-agente, residuos peligrosos, control de residuos.

Modeling a Hazardous Waste Monitoring System in the Manufacturing Industry

Abstract. Nowadays, information and communication technology has become a necessary component in the planning, design and management of the different processes in the industry sector. To manufacturing companies, the use of multiagent systems aiming to develop hazardous waste monitoring systems facilitates the planning, monitoring, collection, and management of hazardous waste. Intelligent agents have proven to be an efficient solution, since they can do tasks on behalf of the users. Moreover, these agents can use different intelligent techniques and communicate among themselves. For this reason, this work proposes the use of software agents for hazardous waste monitoring in manufacturing companies. This article will describe the analysis and design of our proposal using the INGENIAS methodology.

Keywords: multi-agent system, hazardous waste, waste control.

1. Introducción

Los residuos son materiales o productos cuyo propietario o poseedor desecha y que se encuentra en estado sólido o semisólido, o es un líquido o gas contenido en recipientes o depósitos, y que puede ser susceptible de ser valorizado o requiere sujetarse a un tratamiento o disposición final de acuerdo a lo dispuesto en la Ley y demás ordenamientos que de ella se deriven [1].

La generación y manejo de residuos no es exclusiva del sector industrial, sin embargo, derivado de la naturaleza de las mismas, son las que generan mayor cantidad de residuos comparadas con otros sectores. Es por esto, que requieren un mayor control con el fin de paliar su impacto en el medio ambiente.

Hoy en día las tecnologías de información son herramientas de gran ayuda para la solución de situaciones que conlleva un impacto ambiental, ya que permiten a los usuarios facilitar la toma de decisiones relacionadas con la logística y el manejo de residuos [2] [3]. Al utilizar las tecnologías de información se simplifica el manejo de los residuos y con ello el cumplimiento de las normatividades aplicables para el sector industrial.

Además, el paradigma de agentes constituye una metáfora de los sistemas con agentes de interacción con propósito, y esta abstracción es similar a la forma en que los humanos pensamos y actuamos respecto a nuestras propias actividades [4]. Además, los agentes pueden mejorar el rendimiento de los individuos, así como el del sistema general en el que se encuentran [5,6].

Los agentes tienen entre otras, las siguientes propiedades útiles [7]:

- Autonomía: los agentes operan sin la intervención directa de humanos u otros y tienen algún tipo de control sobre sus acciones y estados internos.
- Habilidad social: los agentes interactúan con otros agentes (posiblemente humanos) a través de algún tipo de lenguaje de comunicación.
- Proactividad: los agentes toman la iniciativa para lograr sus propios objetivos. Es decir, pueden exhibir comportamientos flexibles, proporcionando conocimiento tanto "reactivo" a solicitud del usuario, o "proactivamente", anticipándose a las necesidades de conocimiento del usuario.

Un sistema basado en agentes o sistema multi-agente es aquel en el que el nivel de abstracción utilizado es el agente. En principio, un sistema basado en agentes podría especificarse en términos de agentes, pero no implementarse con un entorno específico de desarrollo software orientado a agentes. Sin embargo, lo ideal es realizar tanto el diseño como la implementación en términos de agentes.

Un sistema basado en agentes puede estar constituido por un único agente (single-agent system) o por múltiples agentes (SMA) [8]. La mayor diferencia entre estos sistemas se basa en los patrones de comunicación. Un SMA se comunica con la aplicación y el usuario, así como con otros agentes del sistema. Sin embargo, en los sistemas basados en un único agente los canales de comunicación están abiertos solamente entre el agente y el usuario.

Las principales características de un SMA son [9]:

- Proveen la infraestructura adecuada para la comunicación entre agentes.

- Son normalmente diseñados para ser sistemas abiertos sin ningún diseño centralizado.
- Los agentes que componen un SMA son autónomos y pueden ser cooperativos o de naturaleza competitiva.

Es por esto, que en el presente trabajo se propone un modelo multi-agente que permita el monitoreo constante de los residuos generados hasta su disposición final. Además, dicho modelo permitirá alertar y tener al tanto a las personas indicadas para la toma de decisiones en caso de presentarse alguna situación atípica que ponga en riesgo al personal o a la población.

El resto de este documento está organizado de la siguiente manera: la sección dos describe la metodología INGENIAS, la cual es la utilizada para modelar nuestra propuesta; la sección tres describe la arquitectura y el modelo utilizando la metodología descrita en la sección anterior; finalmente, la conclusión y el trabajo futuro son presentados.

2. Metodología INGENIAS

Saber diseñar agentes y encajarlos en un sistema no es suficiente ya que [10]:

- Un sistema debe satisfacer las necesidades del cliente que lo solicitó.
- Hay que tomar decisiones fundamentales como elegir las cualidades que se quieren presentes en los agentes (utilizando las diferentes arquitecturas), decidir qué entidades del sistema van a ser agentes o no y organizarlo todo según la plataforma de desarrollo.

Debido a que este proceso no es trivial, se han estudiado las metodologías más importantes para decidir cuál se adapta mejor a las necesidades del proyecto. Como conclusión consideramos a INGENIAS la más adecuada para este proyecto debido a que es una de las metodologías más actualizadas y completas que existen, propone un lenguaje visual para generar los diferentes modelos o vistas [11] y además estos modelos están soportados por sus correspondientes meta-modelos y esto facilita la comprobación automática de inconsistencias en el diseño.

A continuación, se describe brevemente dicha metodología. INGENIAS [10] es una metodología de ingeniería del software orientada a agentes (ISOA) para el desarrollo de sistemas multi-agente que parte de ideas propuestas en MESSAGE y UML [10].

Por otro lado INGENIAS concibe el desarrollo de un SMA como la representación computacional de un conjunto de modelos y, a su vez, cada modelo trata de mostrar una visión parcial del conjunto del SMA [12].

Así, esta metodología dispone de cinco modelos especificados a través de cinco meta-modelos que giran en torno a dos entidades: el agente y la organización. Estos meta-modelos son:

- Meta-Modelo de Agente: Describe a los propios agentes representando las responsabilidades de cada uno y el comportamiento de los mismos.
- Meta-Modelo de Interacción: Trata la coordinación y comunicación entre agentes.

- Meta-Modelo de Tareas y Objetivos: Relaciona el estado mental del agente con las tareas que debe ejecutar.
- Meta-Modelo de Organización: Define cómo se agrupan los agentes, la funcionalidad del sistema, así como las restricciones que hay que imponer sobre el comportamiento de los agentes.
- Meta-Modelo de Entorno: Define los elementos que se encuentran alrededor del SMA.

En la figura 1 se puede ver la relación existente entre cada meta-modelo y cómo las entidades agente y organización son la parte central de la metodología.

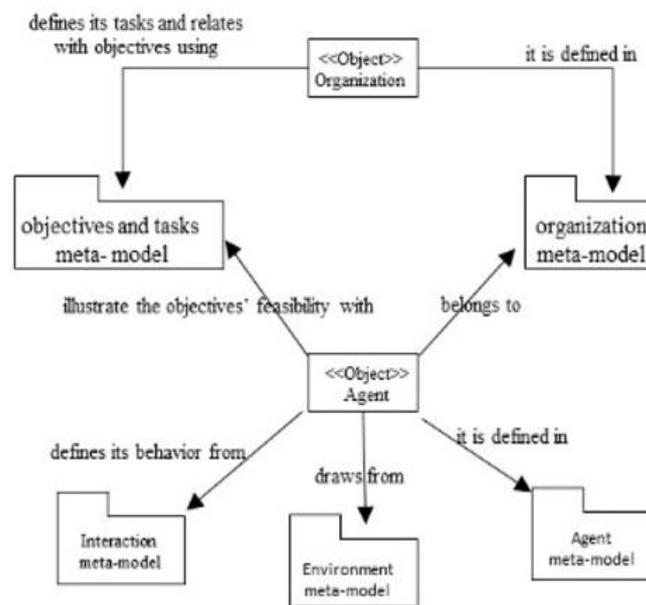


Fig. 1. Relación entre meta-modelos [10].

3. Arquitectura multi-agente

Las principales funcionalidades de un sistema de control y supervisión son: monitorear las variables de operación, procesar dichas variables para generar el control, transmitir las instrucciones de control, permitir las instrucciones de reconfiguración, detectar y diagnosticar condiciones anormales de operación. Estas funcionalidades y tareas pueden ser distribuidas y expresadas a través de una estructura de interacción concurrente y colaborativa.

Con la implementación de agentes inteligentes se puede definir una arquitectura de funcionamiento como se muestra en la figura 2, en la que se define la descripción particular de los elementos que conforman a un agente en concreto y cómo estos elementos interactúan entre sí para lograr su correcto funcionamiento. La arquitectura

determina los mecanismos que utiliza un agente para reaccionar a los estímulos, actuar, comunicarse, etc.

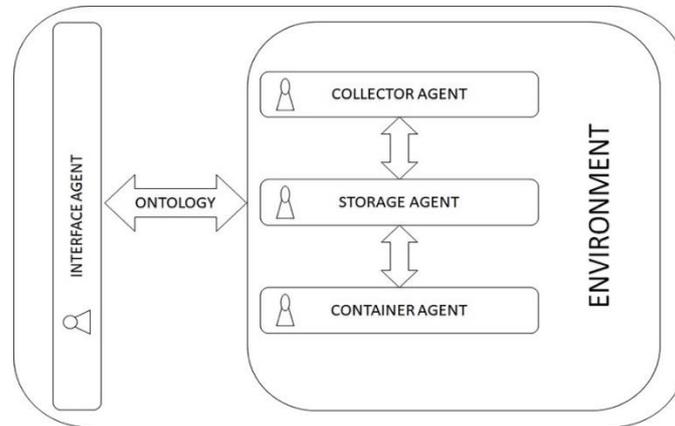


Fig. 2. Arquitectura Multi-agent.

La interacción de agentes (Fig. 2) para el manejo de residuos está definida desde la generación del residuo por el Agente Generador (collector agent), la cual inicia con el depósito en contenedores de los residuos. Una vez que están los residuos en contenedor se dispone al Agente Almacén, el cual gestionará el tiempo para el manejo del recurso definido legalmente para que posterior a esto el Agente Recolector pueda cumplir su actividad hasta la disposición final del residuo.

Para el lograr una comunicación efectiva entre los agentes se utiliza una ontología compartida definida para la comunicación entre ellos.

3.1. Modelo de la arquitectura del sistema

En esta sección se ilustran los meta-modelos de los agentes que describen los roles y las tareas de cada uno de ellos con INGENIAS.

Agente Contenedor. Es el agente a cargo de la gestión de residuos dentro de la industria. Dicha gestión estará definida por las políticas o regulaciones del Agente de Contenedor (Fig. 3). La función principal de este agente es supervisar el nivel de capacidad y tener la información disponible cuando se solicite.

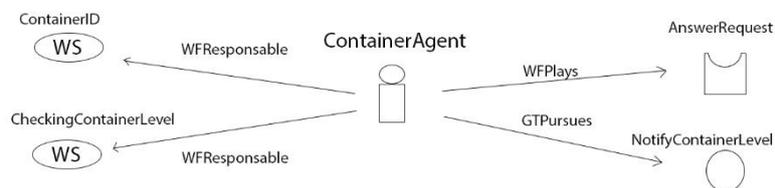


Fig. 3. Agente Contenedor.

La información de monitoreo se utiliza para responder a las solicitudes recibidas (función AnswerRequests). Para responder satisfactoriamente a estas solicitudes, debe asignar un identificador al contenedor (Tarea ContainerID) y monitorearlo (tarea CheckingContainerLevel) con el objetivo de notificar el nivel cuando sea necesario (objetivo NotifyContainerLevel).

Agente Almacén. La responsabilidad de este agente (Fig. 4) es el mayor soporte del sistema, ya que se coordinará con el agente contenedor para la recepción de residuos. El agente recibe el producto para poder realizar sus operaciones internas y cumplir sus objetivos en tiempo de almacenamiento, capacidad de almacenamiento y disposición, la que se realizará al coordinarse con el agente recolector en las actividades necesarias para la entrega y recolección de los contenedores.

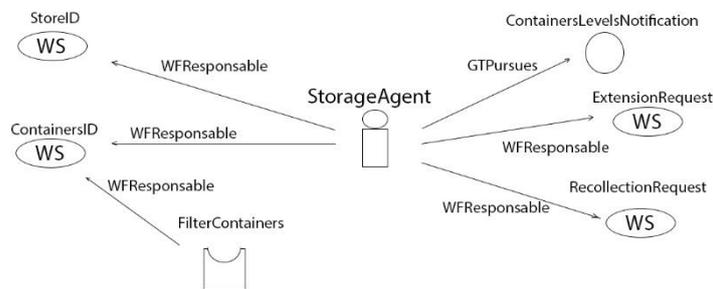


Fig. 4. Agente Almacén.

El objetivo principal del Agente de Almacenamiento es la sincronización con el agente contenedor para llevar a cabo las actividades solicitadas y lograr la recolección de residuos generados (NotifyContainerLevels). Para lograr este objetivo, debe identificar la ubicación de los contenedores en su área (FilteringContainers) a fin de obtener la administración de los diferentes contenedores que se manejan en el almacén (StoreID and ContainerID).

Existen diferentes situaciones en las que se puede solicitar la recolección de cualquiera de los residuos (CollectionRequest) Las razones principales se deben a que el tiempo de gestión de los residuos en el almacenamiento expira, o porque el contenedor está en su capacidad máxima permitida.

Inicialmente, el tiempo que se define para el almacenamiento es de 180 días calendario desde que se coloca el primer residuo dentro del contenedor. Sin embargo, las regulaciones permiten una extensión de tiempo de almacenamiento de residuos.

Las tareas del sector industrial son diversas, y el flujo de residuos puede organizarse dependiendo de la naturaleza de cada uno de ellos. En las empresas de residuos de bajo flujo, se considera una actividad para la recolección de residuos; si hay poca producción de residuos, el agente podrá solicitar una extensión (ExtensionRequest) no mayor de 150 días al comienzo de la recolección.

Agente Recolector. El Agente Recolector se encarga de sincronizar al agente almacén con la logística de la recolección de residuos de la industria, para el manejo correspondiente de los mismos.

El objetivo primario de este agente (Fig. 5) es la recolección de residuos (PickitupResidue). Para esto, el agente debe sincronizarse con el agente de almacenamiento (AcceptRequests). Las tareas que finalizarán la recopilación se definirán mediante la identificación del recolector y la supervisión de almacenamiento (CollectorID and CheckingStore), ya que el Agente recolector debe sincronizar la recopilación cuando sea necesario.

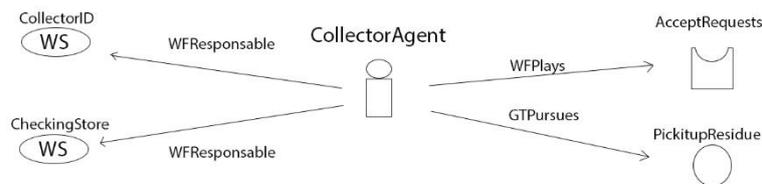


Fig. 5. Agente Recolector.

Agente Interfaz. El agente de interfaz es responsable de organizar la información de los diferentes agentes y presentarla a los usuarios. En otras palabras, permite a los usuarios ver el cumplimiento de los objetivos y metas de los agentes en tiempo real.

La figura 6 muestra el plan para el agente interfaz, cuyo objetivo es recopilar la información (ObtainInfo) de los usuarios del sistema para permitir la interacción, para conseguir este objetivo lleva a cabo las tareas de identificar a los usuarios que interactúan con el sistema (UserID), Gestionar la información capturada (CaptureInfo) y consultar y mostrar las actividades que desea el usuario (RequestInfo). El rol de este agente es filtrar la información (FilterInfo) para tener consistencia entre los resultados obtenidos del agente y las solicitudes de los diferentes usuarios. Para finalizar las solicitudes de los usuarios, el agente debe considerar las tareas de identificación de los usuarios y solicitar la información recopilada por el usuario.

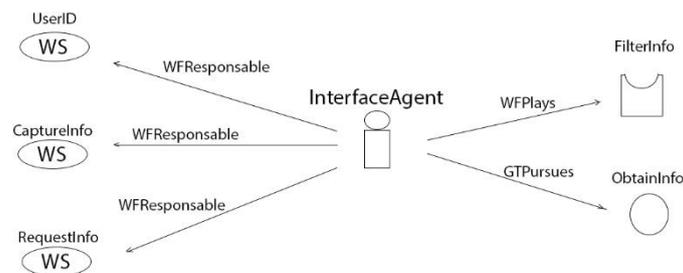


Fig. 6. Agente Interfaz.

Modelo de Organización. La organización que existe en el sistema multi-agente (SMA) está determinada por el objetivo principal, que es la eliminación adecuada de residuos. Para lograr este objetivo, el SMA utiliza la interacción entre los agentes pertenecientes a la misma organización, comenzando con la reunión de los recursos ambientales en el grupo de agentes de supervisión (MonitoringGroup), que se encuentra en la aplicación del Agente de interfaz, ya que se muestra en la figura 7.

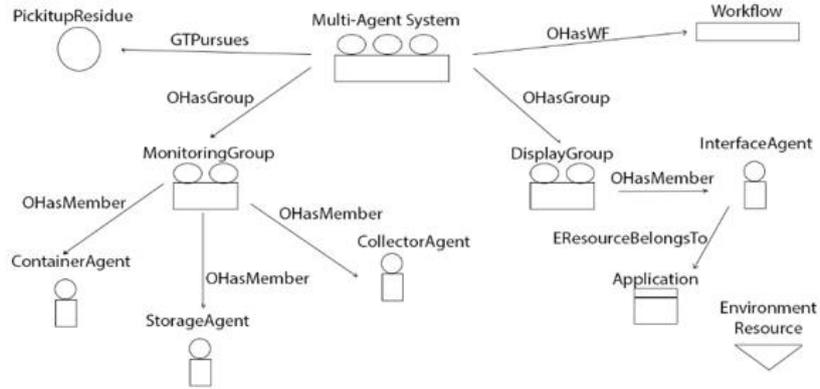


Fig. 7. Modelo de Organización.

La interfaz con los diferentes usuarios del sistema se realiza en la Organización del grupo de visualización (DisplayGroup), que se ha creado para satisfacer las necesidades de información de los usuarios.

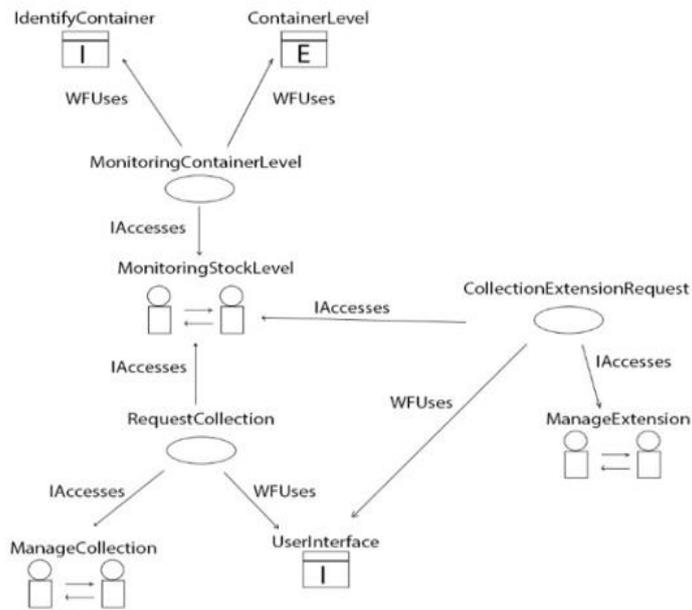


Fig. 8. Modelo de Metas y Tareas.

Modelo de Metas y Tareas. El objetivo de este modelo es mostrar el cumplimiento de los objetivos y tareas en el sistema multi-agente (SMA). En este modelo (Fig. 8) desde el momento en que comienza la recolección de cualquier residuo, comienza el proceso de monitoreo de los contenedores dentro del almacenamiento de la industria (MonitoringContainerLevel); en consecuencia, el almacenamiento se sincroniza con los

otros agentes para finalizar el período de almacenamiento de residuos (CollectionExtensionRequest) o la recolección de residuos (CollectionRequest). Todos los procesos, desde la creación hasta la eliminación final de residuos, estarán disponibles para ser monitoreados y consultados por el usuario.

Modelo de Entorno. El entorno del sistema multi-agentes propuesto (Fig. 9) está definido por los componentes internos del sistema sugerido. Los usuarios interactúan con el sistema y las aplicaciones indican los niveles de los diferentes contenedores que integran el sistema.

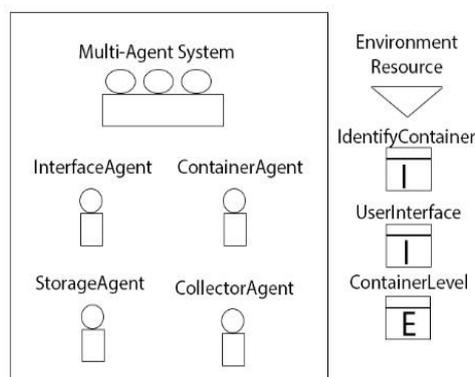


Fig. 9. Modelo de Entorno.

4. Conclusión y trabajos futuros

Este artículo propone una arquitectura multi-agente para el control de residuos peligrosos en empresas de manufactura con el objetivo de aumentar la eficiencia del uso de residuos y evitar al máximo los errores humanos en la gestión de los mismos.

La principal contribución de este trabajo es describir cómo se utilizó la metodología de INGENIAS para desarrollar nuestra arquitectura. Este modelo se encuentra actualmente en el proceso de validación y prueba piloto, y hay propuestas para desarrollarlo como parte del trabajo futuro. Una de esas propuestas es adaptar el modelo al proceso de fabricación en Maquilas Tetakawi's, una compañía de manufactura ubicada en Guaymas, México.

Por último, se contempla el desarrollo de un prototipo. Para esto, se tiene contemplado el uso de la plataforma JADE [13].

Referencias

1. Ley General para la Prevención y Gestión Integral de los Residuos (2003)
2. Anagnostopoulos, T., Kolomvatsos, K., Anagnostopoulos, C., Zaslavsky, A., Hadjiefthymiades, S.: Assessing dynamic models for high priority waste collection in smart cities. *Journal of Systems and Software*, pp. 178–192 (2015)

3. Popa, C., Carutasu, G., Cotet, C., Carutasu, N., Dobrescu, T.: Smart City Platform Development for an Automated Waste Collection System, In *Sustainability*, vol. 9, p. 11 (2017)
4. Greenberg, S., Mercer, S.: A multi-agent architecture for knowledge sharing. In: *Sixteenth European Meeting on Cybernetic and Systems Research*, Viena (2001)
5. Sen, S.: Reciprocity: A foundational principle for promoting cooperative behavior among self-interested agents. In: *Second Int. Conf. Multi-Agent Systems* (1996)
6. Kanj, H., Flaus, J.: Agent-based framework for mitigating hazardous materials transport risk. In: *IEEE International Conference on Evolving and Adaptive Intelligent Systems*, pp. 1–8 (2015)
7. Wooldridge, M., Jennings, N.: *Intelligent Agents: Theory and Practice* (1995)
8. Wooldridge, M., Ciancarini, P.: *Agent-Oriented Software Engineering: The State of the Art.*, vol. 1957, Berlin: Springer (2001)
9. Weiss, G.: *Multiagent Systems: A Modern Approach to Distributed Artificial Intelligence* (1999)
10. Pavón, J., Gómez-Sanz, J., Fuentes, R.: The INGENIAS Methodology and Tools. *Henderson-Sellers*, pp. 236–276 (2005)
11. Gómez-Sanz, J., Fuentes, R., Pavón, J., García-Magariño, I.: INGENIAS development kit: a visual multi-agent system development environment. In: *Proceedings of the 7th Conference on Autonomous Agents and Multi-agent Systems*, Berlin (2008)
12. Gascuena, J., Navarro, E., Fernandez-Sotos, P., Fernandez-Caballero, A., Pavon., J.: IDK and ICARO to develop multi-agent systems in support of Ambient Intelligence. *Journal of Intelligent & Fuzzy Systems*, vol. 28, p. 1 (2015)
13. Java Site: JAVA Agent DEvelopment Framework, <https://jade.tilab.com/>. Último acceso: 2019/04/20