

# Caracterización de las metodologías INGENIAS y GAIA en las etapas de análisis y diseño

## Characterization of the INGENIAS and GAIA methodologies in the stages of analysis and design

Iván Fernando LEAL Ramírez [1](#); Johanna Marcela SUÁREZ Pedraza [2](#); Carlos Andrés GUERRERO Alarcón [3](#); Luz Elena GUTIÉRREZ López [4](#)

Recibido: 12/03/2018 • Aprobado: 29/04/2018

### Contenido

- [1. Introducción](#)
- [2. Metodología](#)
- [3. Resultados](#)
- [4. Discusión](#)
- [5. Aportes](#)
- [6. Conclusiones](#)
- [Referencias bibliográficas](#)

#### RESUMEN:

Este trabajo intenta mostrar la caracterización y el análisis de las diferencias en la implementación de un sistema multiagente con las metodologías INGENIAS y GAIA con base en FIPA. La implementación de estas metodologías y sus diferencias no se han estudiado en detalle, especialmente en las etapas de análisis, diseño y todo el camino hasta la implementación; estas metodologías son bastante diferentes, así como los resultados son tan opuestos en las etapas de análisis y diseño, con respecto a las características y la implementación, los resultados parecen representar dos mundos diferentes; como conclusión general, incluso sin un marco, la implementación de GAIA es muy simple comparándola con INGENIAS y son colaborativas para crear una nueva metodología.

**Palabras-Clave:** Inteligencia Artificial, aplicación informática, tecnología de la información, sistema multi agente.

#### ABSTRACT:

This paper attempts to show the characterization and analyzing the differences in implementation of a multi-agent system with the INGENIAS and GAIA methodologies with FIPA based. The implementation of these methodologies and their differences have not been studied in detail, especially in the stages of analysis, design, all the way to the implementation; these methodologies are quite different, just as the results are so opposite in the analysis and design stages, that regarding features and implementation the results seem to represent two different worlds; as a general conclusion, even without a framework GAIA implementation is very simple comparing it with INGENIAS and are collaborative for create a new methodology.

**Keywords:** Artificial Intelligence, Computer application, Information Technology, Multi Agent System

## 1. Introducción

En esta investigación se caracterizan las metodologías INGENIAS y GAIA para el desarrollo de Sistemas Multiagente-SMA identificando fortalezas, debilidades, carencias y resultados de la ejecución del proceso de construcción de un SMA en las etapas de análisis, diseño e implementación como uso de estas.

Los agentes son entes que realizan algo en beneficio de otros. Los agentes computacionales son un software informático que hace su función para que otros (software,) logren sus metas y también pueda lograr las suyas. Como un sistema es un conjunto de componentes que interactúan entre sí, se puede definir que un SMA es un conjunto de componentes o entes, denominados agentes, que interactúan para beneficiarse unos a otros.

El desarrollo de aplicaciones de SMA puede ser orientado a múltiples metodologías, en esta investigación se trabajan dos: INGENIAS y GAIA. INGENIAS es una metodología deliberativa creada por Jorge Gómez Sanz y GAIA es una metodología reactiva creada por M. Wooldridge y N. R. Jennings, a pesar de que una metodología es deliberativa y la otra es reactiva, ambas tienen modelos que se nombran de manera similar pero no son análogos.

El abordaje del estudio se realizó por medio de una investigación cualitativa cuasi-experimental, la selección de las metodologías para la experimentación fue una muestra no probabilística. Los criterios de selección de las metodologías para el presente estudio fueron: el reconocimiento de los autores del trabajo de SMA para el caso de GAIA y la aceptación mayoritaria de la metodología para INGENIAS.

El presente escrito se estructura en cinco fases, la descripción metodológica de la investigación que se aplicó, los resultados obtenidos de abordar las dos metodologías SMA, luego se abre la discusión en donde se enfrentan los resultados esperados y la experiencia, fruto de eso se presentan los aportes y por último las conclusiones.

## 2. Metodología

El desarrollo de este trabajo es basado en la metodología cualitativa estudiada por (Hernández-Sampieri, Fernández-Collado, & Baptista Lucio, 2014); ésta metodología se fusionó con un diseño cuasi experimental aplicado, lo anterior confluye en que no hay manipulación de variables (Cambel & Stanley, 1963), y la recolección y el análisis de los datos, sirven para descubrir cuáles son las preguntas de investigación más importantes.

Se sugiere iniciar con un contacto directo con el entorno que involucra la indagación y sensibilización para que se logren aportes, implementa un plan de acción para resolver el problema e introducir mejoras. Para el proceso cualitativo se tienen tres fases: a) selección de muestra y recolección de información, b) pensar, analizar e interpretar; c) actuar, resolver problemas e implementar mejoras, las cuales se dan de manera continua, hasta que el problema es resuelto (Hernández-Sampieri et al., 2014), es de acotar que estas fases se realizan de manera simultánea. En consecuencia, la investigación aplicada, permite identificar aportes a través de la experimentación con la muestra objeto de estudio.

### 2.1. Etapas del proceso investigativo

En el proceso investigativo se trabajan tres etapas: diagnóstico, conceptualización e implementación. (Hernández-Sampieri et al., 2014).

**2.1.1. Diagnóstico:** recolección y análisis documental de las metodologías INGENIAS Y GAIA en las etapas de análisis, diseño e implementación, recolectadas desde su documentación fuente.

**2.1.2. Conceptualización:** en esta etapa se realiza una indagación literaria sobre el análisis y diseño en SMA, los modelos necesarios para implementar estas dos fases con cada metodología, por último, la validación de la implementación de cada una de ellas de acuerdo con sus características y principios fundamentales dictados.

**2.1.3. Implementación:** uso de los marcos de trabajo (Guerrero, Londoño, Suárez, & Gutiérrez, 2014) y herramientas que soportan cada una de las metodologías, desarrollando un SMA con base en los modelos realizados en fases previas.

## 3. Resultados

### 3.1. Diagnóstico

La recolección de diferentes documentos base se realiza para cada una de las metodologías y sus temas involucrados los cuales se describen a continuación.

**3.1.1. Sistemas Multi-agente:** Un sistema integra diferentes componentes que interactúan entre sí, de ese mismo modo Wooldridge (2009), expresa que los SMA, son sistemas compuestos por múltiples elementos computacionales que interactúan entre sí, denominados agentes, los cuales tienen la capacidad de coordinar su conocimiento, objetivos, habilidades, toma de decisiones y planes (Piedrahita Ospina, 2012).

**3.1.2. Agente:** Los autores Wooldridge & Jennings (1995) presentan la definición de agente como "un sistema de cómputo, situado en un ambiente cualquiera, el cual es capaz de realizar acciones autónomas que afectan su ambiente de acuerdo a sus objetivos de diseño". Las características más importantes que presenta un agente son: acción (González, Hamilton, Moreno, Marichal, & Mendez, 2006), autonomía (Orrego & Bolívar, 2006), reactividad, proactividad, habilidad social (Odell, 2018), oportunidad, objetivo, entre otras, según lo mencionado por Wooldridge (2009).

**3.1.3. Foundation for Intelligent Physical Agents FIPA:** FIPA (Odell, 2018) Es un estándar bajo el cual se han construido un gran número de plataformas de agentes, el cual provee un estándar de comunicación entre agentes y las diferentes plataformas.

**3.1.4. Java Agent Development JADE:** JADE (Telecom Italia SpA, 2018) es un marco de trabajo desarrollado en JAVA, trabaja bajo el estándar de comunicación FIPA, en su aspecto de implementación usa ontologías como un requerimiento directo para desarrollar software basado en agentes.

**3.1.5. Tipos de Arquitecturas Multi-Agente:** Dentro de las arquitecturas para SMA se encuentran tres tipos, la deliberativa, reactiva y la híbrida. Las metodologías INGENIAS y GAIA hacen uso de la deliberativa y la reactiva respectivamente.

- **Arquitectura Deliberativa:** Basada en la teoría de planificar y ejecutar. Es así que los agentes parten de un inicio, luego realizan una planificación y ejecutan acciones para poder alcanzar sus metas/objetivos (Piedrahita Ospina, 2012), es importante tener en cuenta que las acciones se deben ejecutar de forma oportuna, eso quiere decir que se deben llevar a cabo en el momento adecuado.
- **Arquitectura Reactiva:** Esta arquitectura pondera a la acción primero que al razonamiento. Los agentes no hacen ningún tipo de uso de la experiencia adquirida en el pasado, como consecuencia no tienen conocimiento de su historia (Weiss, 2000). La percepción del ambiente según los estímulos que recibe el agente es conocido como estímulo-respuesta (Nwana, 1996).

El razonamiento de los agentes en la arquitectura reactiva es prácticamente nulo. Esta arquitectura es acogida en el ámbito de la robótica (Berlanga, Sanchis, Isasi, & Molina, 2002), el actuar de la arquitectura reactiva está basado en sensores.

**3.1.6. Metodologías de desarrollo de SMA:** La palabra metodología sugiere unos pasos a ejecutar en cierto orden, las metodologías de desarrollo en el ámbito de la ingeniería de sistemas especifica una serie de pasos, probablemente consecutivos y lógicos que conducen a realizar alguna implementación para darle solución a algún problema; es por esto que existen metodologías como GAIA e INGENIAS que se apoyan en este concepto (Norvig & Russell, 2018).

La metodología de arquitectura deliberativa INGENIAS y la metodología de arquitectura reactiva GAIA, hacen uso de los diferentes conceptos de los SMA como son: acción, oportunidad, objetivos, deseos y metas; que complementan lo aportado por Wooldridge (2009).

**3.1.6.1 GAIA:** Es una metodología para SMA basada en roles, acciones (oportunidad), objetivos y ontologías. Los roles se definen con el fin de saber el quehacer del agente en el SMA. Las acciones son la forma de buscar alcanzar los objetivos o metas de forma oportuna; pero un agente no va a actuar o a responder sobre algo que no tiene conocimiento, para eso se hace uso de las ontologías (Wooldridge, Jennings, & Kinny, 2000). Las dos primeras etapas para el desarrollo de un SMA bajo la metodología GAIA son análisis y diseño, en la metodología GAIA estas etapas contienen cinco modelos que dirigen la construcción de un SMA.

- **Análisis:** La etapa de análisis se compone de dos modelos: roles e interacciones, se inicia con la identificación de roles en el sistema lo que implica conocer su estructura. Las estructuras son jerárquicas y hacen parte de una organización o sistema, luego se encuentran los roles o grupo de estos, la interacción entre ellos conduce a la composición de la sociedad, este esquema es el mismo bajo el cual trabajan los SMA (Wooldridge et al., 2000).
  - **Roles:** Un rol es definido por: responsabilidades, permisos, actividades y protocolos (Wooldridge et al., 2000). Cada rol tiene sus propias responsabilidades, así, un ejemplo del rol es vicepresidente, y su deber es llamar mensualmente a los directores de área a reunión de reporte; de igual forma tiene la responsabilidad de rendir informe al presidente cada dos meses. De acuerdo con Wooldridge et al. (2000), las responsabilidades están sujetas de los roles, y son divididas en dos, liveness properties y safety properties. Los roles tienen un aspecto muy interesante, porque por medio de ellos se esquematizan modelos organizacionales (Thangarajah, Padgham, & Harland, 2002). La identificación de los roles se puede dar de una manera más sencilla al dividirse en: individuos, departamentos y organizaciones.
  - **Interacciones:** Son los protocolos que indican cómo interactúan los diferentes roles para establecer la comunicación, la interacción entre roles está condicionada a seguir un grupo de pasos, esos requisitos que se deben cumplir para interactuar con otro rol (Wooldridge et al., 2000).
- **Diseño:** La etapa de diseño en GAIA consta de tres modelos: agentes, servicios y conocimiento. El primero incluye las instancias de agentes que serán usadas por los tipos, el segundo incluye los servicios principales que ofrece el agente específico, el tercero documenta las líneas de comunicación entre los diferentes agentes (Wooldridge et al., 2000).
  - **Modelo de Agentes:** Es el refinamiento del modelo de roles, la correspondencia entre el modelo de roles y el de tipos de agentes normalmente es "uno a uno" (Wooldridge et al., 2000).
  - **Modelo de Servicios:** Para Wooldridge et al. (2000) el modelo de servicios es la especificación de cada rol de agente y sus responsabilidades, se especifican cada uno de los servicios que va a ser desarrollado por un rol específico.
  - **Modelo de Conocimiento:** Ilustra desde cual tipo de agente y a cual tipo de agente se efectúa la comunicación; no definen los tipos, tampoco definen la oportunidad "momento exacto o cuándo" de los mensajes (Thangarajah et al., 2002).

**3.1.6.2 INGENIAS:** La documentación contenida en este escrito, se fundamentó en la tesis doctoral de Jorge J. Gómez Sanz, en donde se presenta el "Modelado de Sistemas Multi-Agente" con INGENIAS (Gómez Sanz, 2002). La metodología INGENIAS se compone de seis metamodelos para el desarrollo de SMA. El marco de trabajo de INGENIAS usa los metamodelos, estos a su vez se basan en las tareas a realizar por cada agente o rol de este, teniendo en cuenta los objetivos principales a conseguir, y las interacciones que deben existir entre los agentes para satisfacer dichos objetivos.

- **Análisis:** La etapa de análisis se compone de tres modelos: casos de uso, organización y tareas y metas. El modelo de organización plasma la relación que van a tener agentes y tareas, el modelo de tareas y metas especifica las tareas y cuales objetivos satisfacen las mismas (Gómez Sanz, 2002).
  - **Modelo de Casos de Uso:** este modelo se compone de los módulos y actores que integran el SMA (Gómez Sanz, 2002), los casos de uso pueden ser representados con el caso de uso de INGENIAS que es una extensión del caso de uso de UML.
  - **Modelo de Organización:** Representa las interacciones entre las tareas de un grupo de trabajo y organizaciones, incluye los agentes y sus tareas (basado en (Gómez Sanz, 2002)).
  - **Modelo de Tareas y Metas:** Ilustra las tareas a realizar con los objetivos que las satisfacen y que objetivos persigue cada rol de agentes, las interacciones que producen las tareas y las precondiciones y postcondiciones que manejan las tareas (basado en (Gómez Sanz, 2002)).
- **Diseño:** En esta etapa se continua con el modelo de agentes, el cual representa roles y objetivos entre otros. Así mismo se desarrolla el modelo de interacciones y el modelo de entorno (Gómez Sanz, 2002).
  - **Modelo de Agentes:** es integrado por los agentes que existirán en la aplicación, los roles que manejará cada agente, objetivos a seguir, y las tareas que realizará cada uno para lograr sus metas(objetivos). También se dan a conocer el estado mental de los agentes, sus creencias y acciones (Autonomía) (Gómez Sanz, 2002).
  - **Modelo de Interacciones:** representa el intercambio de conocimiento entre los agentes y su flujo de comunicación, este modelo incluye las tareas que se deben desarrollar en cada comunicación con sus respectivos actores (Gómez Sanz, 2002).
  - **Modelo de Entorno:** Se definen las entidades del entorno con las que se vaya a interactuar, se involucran los recursos que va a usar el SMA como: a) interacción con aplicación externa o b) interacción con agente (Gómez Sanz, 2002).

## 3.2. Conceptualización

La indagación literaria condujo a realizar una comparación de las características que debe tener un SMA vs el estándar que las condensa, igualmente se enfoca en las etapas de análisis y diseño en SMA con INGENIAS y GAIA, se crean los modelos necesarios para implementar estas dos etapas con cada metodología. La validación de los modelos se realiza con el uso de cada metodología de acuerdo con sus características y los principios que las rigen y son tomados de su documentación fuente.

**3.2.1. Características SMA Vs FIPA:** Las características de SMA nombradas por autores reconocidos como Wooldridge y Jennings, son fundamentales en la caracterización, también se incluyen otras que se consideran importantes en este contexto. Wooldridge et al. (2000) proponen el grupo de características que debe tener un SMA, estas características son: habilidad social, autonomía, reactividad y proactividad. El estándar FIPA, fija una guía acerca de los elementos que debe integrar un Agent Management System-AMS.

La comprensión del estándar FIPA incluye términos como: Agent Platform-AP, Agent Management System-AMS, Message Transport Service-MTS y el Directory Facilitator-DF. La AP involucra "machine, operative system, agent support software" también incluye los componentes de administración de agentes designados por FIPA "DF, AMS, MTS".

Un AMS, SMA en español, puede contener diferentes AP relacionadas o registradas en el mismo, se pueden registrar diferentes DF en un mismo DF para llegar a tener confederaciones. El MTS, es el método de comunicación por defecto entre agentes que se encuentran albergados en diferentes AP. Para que un agente pueda ser ubicado y recuperar su descripción, identificador y características debe ser registrado en el DF, lo anterior ha sido fundamentado en el documento de arquitectura SC00001 (FIPA, 2002a) que se encuentra en el sitio de FIPA.

Los descriptores y clasificadores para un AMS son relacionados en el documento de arquitectura, una implementación de un AMS, debe comprender unos ítems específicos según el estándar FIPA, este grupo de elementos se listan en el documento de arquitectura, los componentes de un AMS se encuentran en el cuadro de elemento abstractos de arquitectura FIPA (FIPA, 2002a).

La evaluación de cada uno de los documentos que proporciona el estándar FIPA, y las características que debe cumplir un SMA, muestra que varios de los elementos han sido enfocados a la parte técnica. La tabla 1, contiene la interpretación de características vs. elementos de arquitectura.

**Tabla 1**  
Interpretación de elementos FIPA vs características de SMA

Características-SMA	Característica-FIPA	Estándar-FIPA
Habilidad social (interacción, comunicación,)	Request Interaction Protocol Specification, Communicative Act Library Specification, FIPA SL Content Language Specification	SC00008, SC00026, SC00037, SC00061, SC00067, SI00091
Autonomía	Device Ontology Specification, FIPA Nomadic Application Support Specification, FIPA SL Content Language Specification	SC00008, SI00014, SI00091
Reactividad	FIPA Nomadic Application Support Specification,	SI00014(detección de cambio de ambiente "modo de conexión"),33(reacción a romper una comunicación y establecer cual agente da la respuesta "evita conflictos")
Proactividad	FIPA Agent Management Specification (Páginas blancas y amarillas),	SC00023,33(reacción a romper una comunicación y establecer cual agente da la respuesta "evita conflictos"), 34 (por que está pendiente el "recruiter" de los agentes y de sus conocimientos para poder enlazarlos)

Fuente: Elaboración propia

### 3.3. Interpretación de elementos FIPA vs características

La tabla 1 muestra la clasificación de los documentos que posee el estándar FIPA, cada uno fue leído y analizado dentro del contexto de las características que tienen los SMA propuestas por Wooldridge (2009).

**3.3.1. Habilidad social:** Dentro de un sistema la habilidad social está basada en la interacción que pueda tener un individuo con su entorno en el que participa, el acto comunicativo es uno de sus pilares, este puede soportar el contexto para lograr ejecutar tareas y finalmente alcanzar sus objetivos. Cada uno de los documentos que posee el estándar FIPA, ha sido leído y analizado dentro del contexto de las características que tienen los SMA, la interpretación de pertinencia para la habilidad social es la siguiente:

**3.3.1.1 Estándar SC00008:** El estándar SC00008 (Cernuzzi, Omicini, Molesini, & Zambonelli, 2011) es el esquema semántico, por medio de este se realizan los actos comunicativos y se realiza la comunicación. Este estándar está normalizado por el lenguaje semántico que brinda las constantes, numéricas, de fecha y de cadena, términos de variables y estándares de programación que se deben tener en cuenta para trabajar agentes con FIPA.

**3.3.1.2 Estándar SC00026:** El estándar SC00026 tiene en cuenta la necesidad de comunicación y que hacer para iniciar la comunicación, para esto se debe establecer un protocolo inicial como lo propone la especificación del protocolo de interacción, donde se establecen el acuerdo para iniciar una conversación, el rechazo, la falla, la salida de información y la salida de un resultado (basado en el estándar SC00026) (Zambonelli, Jennings, & Wooldridge, 2005).

**3.3.1.3 Estándar SC00037:** El estándar SC00037 trabaja el acto comunicativo que es una forma certera de interacción, se debe aclarar que el lenguaje de acto comunicativo "Communicative Act Library" ACL no es obligatorio de implementar en un SMA, excepto el "not-understood" (Zambonelli, Jennings, & Wooldridge, 2003). Un nombre adecuado para un acto comunicativo, debe ser una palabra en ingles que no cause conflicto con las librerías existentes. Los actos comunicativos están dados por aceptación de propuestas, acuerdo de inicio de conversación, cancelación de conversación, llamado de propuestas, "accept-proposal, agree, cancel, confirm, disconfirm, failure, inform, inform-if, inform-ref, not-understood, propagate, propose, proxy, query-if, query-ref, refuse, reject-proposal, request, request-when, request-whenever, suscribe". Cada una de las acciones comunicativas anteriores, posee un modelo formal y un ejemplo para cada uno, documentados en el SC00037 (Zambonelli et al., 2003).

**3.3.1.4 Estándar SC00061:** El estándar SC00061 (Zambonelli et al., 2005) describe la estructura del mensaje ACL, tomando elementos como: tipo de acto comunicativo, participantes en la comunicación, contenido del mensaje, y el control de conversación. El contenido del mensaje posee una descripción, este a su vez contiene: el idioma, la codificación y la ontología con lo cual se estructura el mensaje ACL.

**3.3.1.5 Estándar SC00067:** El estándar SC00067 (Moraitis, Petraki, & Spanoudakis, 2002) describe el mecanismo que es usado para transferir el mensaje ACL entre agentes. El "Message Transport Service" MTS es suministrado por un "Agent Communication Channel" ACC. El ACC contiene: interfaces estándar, propiedades de esas interfaces, un manejador del comportamiento de ese mensaje, una interpretación del envoltorio de ese mensaje, y una forma de manejar tanto los mensajes salientes como los recibidos, estos mensajes pueden ser únicos o múltiples.

**3.3.1.6 Autonomía:** Los agentes son fiables y son tolerantes a fallos (Telecom Italia SpA, 2018), los agentes son autónomos en la toma de sus decisiones. Resultado de la afirmación anterior se indica que "la autonomía es la noción central de agencia" (Gomez-Sanz, Fuentes, Pavón, & García-Magariño, 2008), la palabra autonomía sugiere que no es necesario que un humano realice acciones para poder completar su tarea o proceso. Los autores Rich, Knight, González Calero, & Bodega (1994) en su libro "Artificial Intelligence", habla acerca de la autonomía como una debilidad que se ve coaccionada por la colaboración o dependencia de otros agentes, eso "pone limite" a la autonomía; la autonomía es el foco de atención para varios autores, además se encuentra en el contexto de colaboración con otros agentes, la opinión de autores como Weiss (Weiss, 2000) y Norvig (Norvig & Russell, 2018) refuerzan ese concepto. La interpretación de pertinencia para la autonomía es la siguiente:

**3.3.1.7 Estándar SC00008:** El estándar SC00008 (Cernuzzi et al., 2011) trabaja la semántica para las fórmulas bien formadas "Well-Formed Formulas" WFF (Cernuzzi et al., 2011), se muestra cómo se puede realizar una comunicación desde las fórmulas atómicas y su combinación WFF para: "negation, conjunction, disjunction, implication, equivalence, universal quantification", ofrece ejemplos de la interacción con fórmulas atómicas, operadores referenciales, términos, términos funcionales, resultados de predicados, adicionalmente contiene notas sobre las reglas gramáticas descritas y sus formas proposicionales y restricciones de decisión.

**3.3.1.8 Estándar SI00014:** El estándar SI00014 (Botia, Gonzalez, Gomez, & Pavon, 2008) fue trabajado sobre la autonomía en movilidad y manejo de ontologías; la autonomía de movilidad está enfocada en facilitar a los usuarios nómadas su migración de trabajo como es el cambio de una Local Area Network LAN a una red móvil GPRS-UMTS. La Quality of Service QoS es orientado también hacia el monitoreo de cambio de redes con el mismo concepto de nomadidad, ya que tiene presente el cambio de dispositivos "Personal Computer, laptop, Tablet, smartphone", el cambio de tecnología de comunicaciones LAN, GPRS-UMTS entre otros aspectos que implican el movimiento del individuo de un lugar a otro.

**3.3.1.9 Estándar SI00091:** El estándar SI00091 (FIPA, 2002c) trata sobre ontología para dispositivos "device-ontology", incluye términos como "frame, ontology, parameter, description, presence, type, reserved values", los describe, indica su tipo, si es obligatorio y si tiene valores reservados, algo interesante es que involucra tanto hardware como software, es así que comprende: cpu, memoria, ui y connection type; por software incluye OS y agent-platform. El estándar SI00091 describe como se usan las ontologías en dispositivos, para el caso ejemplar usan un teléfono inteligente. Este estándar contiene un ejemplo enfocado en perfiles que se asignan según el ambiente de los dispositivos y el tipo de los mismos con una selección automática dependiendo del tipo de dispositivo.

**3.3.2. Reactividad:** se fundamenta en la reacción a la percepción de un cambio sucedido en el ambiente. La reactividad está encadenada a la arquitectura reactiva, se basa en el "estimulo-respuesta" (Norvig & Russell, 2018) que es ampliamente usada en la robótica (Wooldridge et al., 2000). Cada uno de los documentos que posee el estándar FIPA ha sido leído y analizado dentro del contexto de las características que tienen los SMA, la interpretación de pertinencia para la reactividad es la siguiente:

3.3.2.1 *Estándar SI00014 y SI00033*: El estándar SI00014 (detección de cambio de ambiente "modo de conexión") y el SI00033 (reacción a romper una comunicación y establecer cual agente da la respuesta "evita conflictos") son estándares que se dedican a reaccionar en el ámbito de conexión de los agentes focalizados en la movilidad.

**3.3.3. Proactividad**: comprende el desarrollo de tareas, que se espera se puedan lograr sin necesidad de la intervención del ser humano, eso redundaría en la proactividad (Kim, Masoumzadeh, & Joshi, 2012). La interpretación de los documentos del estándar FIPA de pertinencia para la proactividad es la siguiente:

3.3.3.1 Estándar SC00023: En el estándar SC00023 (FIPA, 2004) se especifican las funciones soportadas por el "Agent Management System" AMS, en este se involucran las páginas blancas en las cuales se registran los servicios de los agentes, y las páginas amarillas donde se registran los agentes "Agent Identifier" AID; dentro de las funciones ofrecidas por el AMS se encuentran "register, deregister, modify, search, get-description" las cuales son útiles para lograr una interacción efectiva entre agentes, de igual forma evitan contactos fallidos e interacciones inútiles con lo cual el agente solo se dedica a realizar su tarea.

3.3.3.2 Estándar SC00033: El estándar SC00033 (FIPA, 2002b) hace énfasis en el fipa-brokering, se trata de un agente encargado de romper una respuesta concurrente, y así direccionar la respuesta para que exista una retroalimentación de la consulta fipa-query, para lograr redireccionar la comunicación hacia el emisor.

3.3.3.3 Estándar SC00034: El estándar SC00034 (FIPA, 2002e) hace referencia a un agente llamado fipa-recruiting, que se encarga de recibir las consultas y evitar la carga del agente que va a realizar el proceso, con el objetivo de evitar sobrecarga de trabajo en el agente destino, ese agente hace la función de sniffer.

Los estándares SC00027, SC00028, SC00029, SC00030, SC00033, SC00035 y SC00036 son protocolos de comunicación que están creados para orientar la implementación, por lo tanto, se dedican a la parte técnica, este grupo de estándares no está dirigido a cumplir las características de los agentes, por el contrario, son soporte tecnológico para lograr una buena implementación y reducir el grado de complejidad de FIPA.

Los estándares SC00069, SC00088, SC00070, SC00071, SC00075, SC00084, SC00085 y SC00088, se encargan de mostrar cómo es la sintaxis para el transporte del mensaje con la codificación y la representación del tiempo.

El estándar SC00094 (FIPA, 2002d) es muy similar al SI00014 (Botia et al., 2008), el SI00014 habla de la definición de la ontología para representar la calidad del servicio QoS enfocado a la implementación de la SI00091 (FIPA, 2002c), este se enfoca en varios temas desde la perspectiva de análisis y diseño también se tienen ejemplos que orientan a llegar a una buena implementación.

Wooldridge (2009) ofrece las características estándar para SMA (proactividad, reactividad, autonomía y habilidad social), aunque es el amén de las características para SMA, en esta investigación se ha ampliado el espectro para incluir a diferentes investigadores que trabajan características que pueden parecer estar o no alineadas con las mismas, como resultado se obtiene que al analizar los fundamentos teóricos, las definiciones de los autores y compararlas con las características brindadas por el autor Wooldridge (2009), se notan redundantes o en contra posición a las definiciones estándar brindadas por él.

**3.3.4. Comunicación**: se basa en el acto comunicativo de interacción, cualquier sistema que trabaje en red, hace uso de la comunicación, ésta se debe dar por el contexto y conocimiento del agente que se encuentra basado en ontologías (Gómez-Sanz, Pavón, & Fuentes, 2005).

**3.3.5. Descentralización**: enfatiza en la autonomía que cada agente debe poseer, cuando un agente posee su propia visión del sistema, se puede decir que es un sistema descentralizado: no existe un agente sobre el cual se rinde control (Telecom Italia SpA, 2018).

**3.3.6. Sociabilidad**: es una característica que tiene un papel importante en las sociedades de agentes; para existir un sistema multiagente los agentes deben comunicarse con otros agentes (Gómez-Sanz et al., 2005) para poder cumplir con sus objetivos, el anterior ítem es lo que los convierte en entes sociables. Simplemente de no cumplirse ese principio no habría comunicación, lo cual negaría una de las premisas iniciales y que son el pilar de las sociedades; como dice (Gómez-Sanz et al., 2005) la dependencia de otros agentes "reduce la autonomía" pero así mismo aumenta la sociabilidad y la comunicación entre los mismos agentes para así lograr tener un sistema multiagente.

La comparación entre diferentes características de los SMA y el estándar FIPA proporciona una base sólida de entendimiento, para poder llegar a una buena implementación, ya que se tienen enfoques desde la perspectiva de análisis y diseño, pero además de todo, brindan un grupo de ejemplos con buenas prácticas que facilitan la implementación; de igual manera, reducen la ocurrencia de errores en los que se puede ver inmerso el uso de una plataforma o una librería, a su vez, establece un camino y muestra que el mismo estándar es usado en varios de los ítems que usa la propia FIPA.

**3.3.7. INGENIAS y GAIA**: Para realizar un estudio detallado de INGENIAS Y GAIA, se deben abordar las metodologías no solo desde la perspectiva teórica, sino también desde su uso. Para validar un modelo de SMA se debe usar con el objetivo de descubrir cuáles son las fortalezas y debilidades que presentan cada una de las metodologías, cuáles son los modelos que poseen, si se pueden encontrar algunos que sean análogos, y si su fortaleza está en las etapas de análisis y diseño, o, por otro lado, se encuentra en la implementación.

3.3.7.1 *Cuadro comparativo GAIA vs. INGENIAS*: En la tabla 2 se comparan los modelos entre las metodologías INGENIAS y GAIA, y se indica la existencia de algún modelo homólogo en la metodología contraria.

**Tabla 2**  
Comparación INGENIAS Y GAIA nivel teórico análisis y diseño

Funcionalidad	INGENIAS	GAIA	Observaciones
Asignación de responsabilidades	M. Organización	M. Roles	Al interpretar el dogma teórico de cada una de las metodologías, se encuentra que estos dos modelos son homólogos.
Direccionamiento de comunicación interna	M. Interacciones	M. Interacciones	Se encuentra en ambas metodologías, mientras en GAIA solo involucra el acto de comunicación, en INGENIAS adicionalmente involucra el conocimiento.
Caracterización de roles	M. Agentes	M. Agentes	En GAIA y en INGENIAS el modelo de agentes es un refinamiento del modelo de roles, además en INGENIAS involucra: creencias, estados mentales, creencias y acciones.
Especificación de funcionalidades	M. tareas y metas	M. Servicios	En GAIA es la caracterización de los servicios que va a desarrollar cada rol, mientras en INGENIAS, se puede homologar con el modelo de tareas y metas, pero involucra tanto los objetivos como las metas junto con pre y post condiciones de estas.
Manejo de conocimiento	Hechos	Ontologías	El manejo de ontologías no es tenido en cuenta en la metodología INGENIAS ya que se basa en los hechos de los agentes.
Direccionamiento de comunicación externa	M. Entorno	M. Interacciones	GAIA no tiene este modelo, pero INGENIAS si lo posee, es homologable en GAIA con el modelo de interacciones.

Fuente: Elaboración propia

### 3.4. Implementación

La implementación se da desde una perspectiva del uso de las metodologías de SMA INGENIAS y GAIA, se analizan los modelos y se hacen aportes que llenan los vacíos encontrados en cada una de ellas intentado completar tanto los modelos como las metodologías propuestas.

**3.4.1. Metodología GAIA:** En la etapa de análisis esta metodología se compone de los modelos de roles e interacciones y en la etapa de diseño comprende los modelos de agentes, servicios y conocimiento.

**3.4.1.1 Modelo de Roles:** la figura 1 plantea un ejemplo como ilustración del modelo de roles, en la parte izquierda se muestra parte del gráfico de uno de los casos de estudio abordados, el ejemplo es el siguiente:

**Figura 1**  
Ejemplo modelo de Roles con GAIA

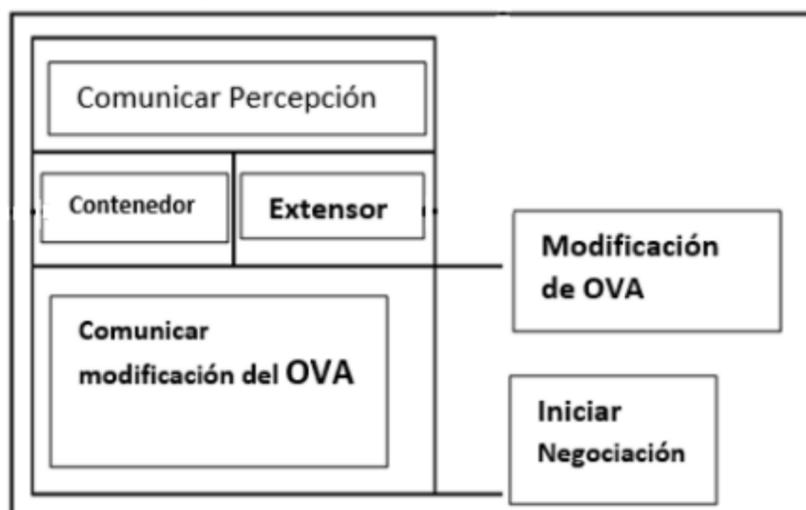
<b>Role Schema:</b>	<b>EXTENSOR</b>
<b>Description:</b>	Responsable de comunicar los cambios percibidos al realizar una extensión del OVA y negociar con el agente contenedor. Protocols: envío de datos al agente contenedor para inclusión en el contenido, negociación de inclusión de datos con el agente contenedor.
<b>Protocols and Activities:</b>	Activities: getOvaExtention, getNegociateAddContent. Protocols: enviar datos al agente contenedor para la inclusión del contenido, negociar inclusión de datos en caso de ser negados.
<b>Permissions</b>	Reads: <ul style="list-style-type: none"> <li>- Suministra getOvaExtention</li> <li>- Suministra getNegociateAddContent</li> </ul> Generates: <ul style="list-style-type: none"> <li>• Envía datos al contenedor</li> </ul>
<b>Responsibilities:</b>	
- <b>Liveness</b>	EXTENSOR = getOvaExtention EXTENSOR = getNegociateAddContent
- <b>Safety</b>	getNegociateAddContent = false -> getNegociateAddContent

Solución GAIA

Fuente: Elaboración propia, basado en (Wooldridge et al., 2000)

**3.4.1.2 Modelo de interacciones:** La Figura 2 plantea un ejemplo del modelo de interacciones que es parte de uno de los casos de estudio tenidos en cuenta para la caracterización.

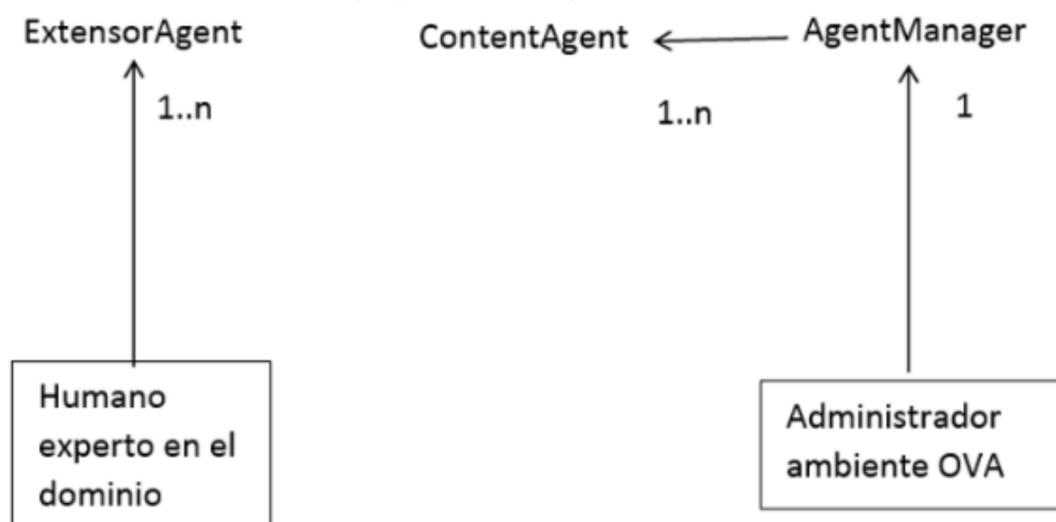
**Figura 2**  
Ejemplo modelo de interacciones con GAIA



Fuente: Elaboración propia, basado en (Wooldridge et al., 2000)

**3.4.1.3 Modelo de agentes:** La Figura 3 plantea un ejemplo del modelo de agentes que es parte de uno de los casos de estudio tenidos en cuenta para la caracterización.

**Figura 3**  
Ejemplo modelo de Agentes con GAIA



Fuente: Elaboración propia, basado en (Wooldridge et al., 2000)

**3.4.1.4 Modelo de servicios:** La Figura 4 muestra un ejemplo del modelo de servicios que es parte de uno de los casos de estudio tenidos en cuenta para la caracterización.

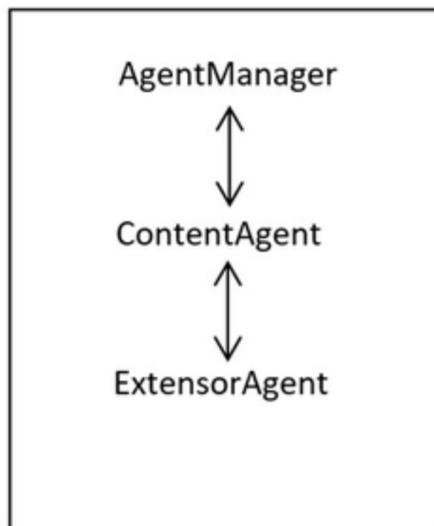
**Figura 4**  
Ejemplo modelo de servicios con GAIA

Service	Inputs	Outputs	Pre-Condition	Post-Condition
 Comunicar cambios percibidos	getOvaExtention	sendMetadataToContent	true	negociateContent
 Verificar pertinencia de cambio negociando	getNegociateAddContent	sendMetadataToManager	previousOVAModify	true
 Comunicación Socializar cambio a otros OVA	getComunicateAddContent	communicateMetadataToOtherOVAs	Extensor communicate to manager	true

Fuente: Elaboración propia, basado en (Wooldridge et al., 2000)

3.4.1.5 *Modelo de conocimiento*: La Figura 5 muestra un ejemplo del modelo de conocimiento que es parte de uno de los casos de estudio tenidos en cuenta para la caracterización.

**Figura 5**  
Ejemplo modelo de conocimiento con GAIA

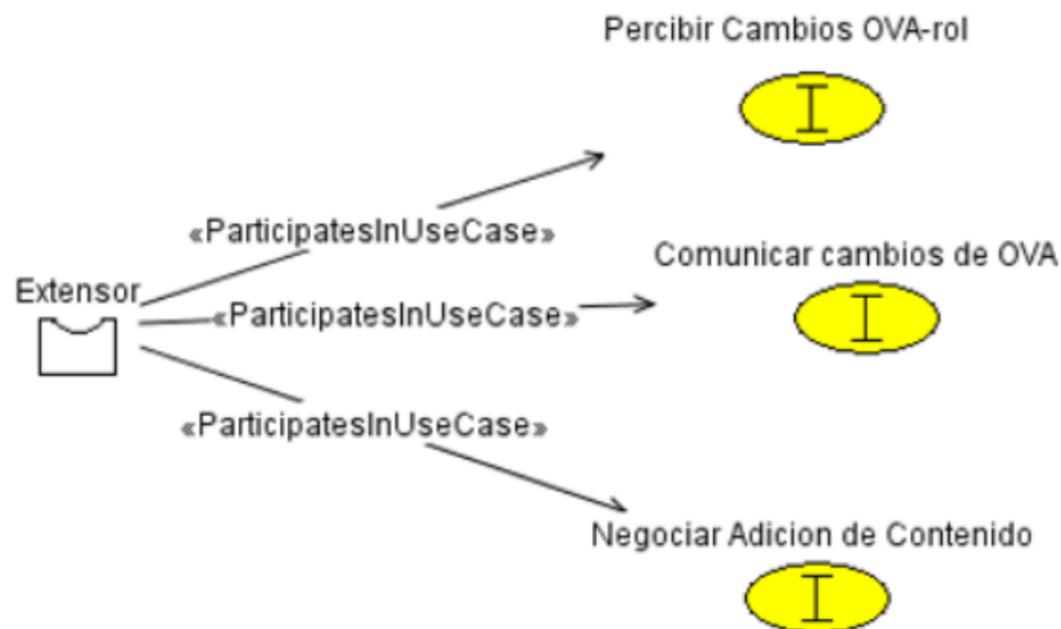


Fuente: Autor investigación, basado en (Wooldridge et al., 2000)

3.4.2 **Metodología INGENIAS**: En la etapa de análisis esta metodología se compone de los modelos de casos de uso, organización y tareas y metas, en la etapa de diseño comprende los modelos de roles, interacciones y entorno.

3.4.2.1 *Modelo de casos de uso*: La figura 6 muestra un ejemplo del modelo de casos de uso que es parte de uno de los casos de estudio tenidos en cuenta para la caracterización.

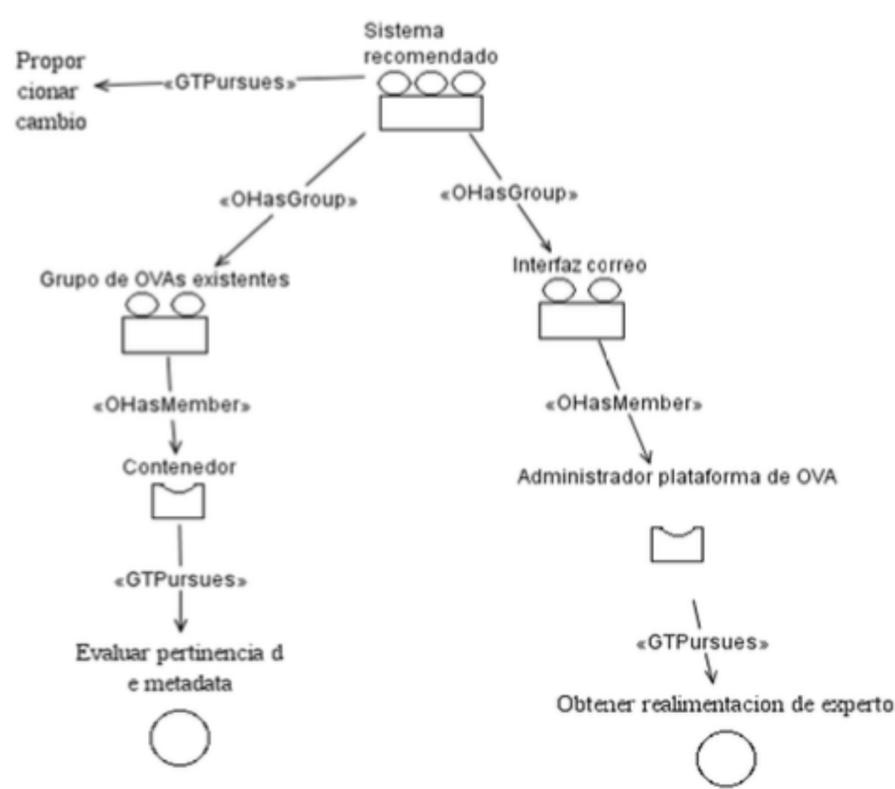
**Figura 6**  
Ejemplo modelo de casos de uso con INGENIAS



Fuente: Autor investigación, basado en (Sanz & Mestras, 2004)

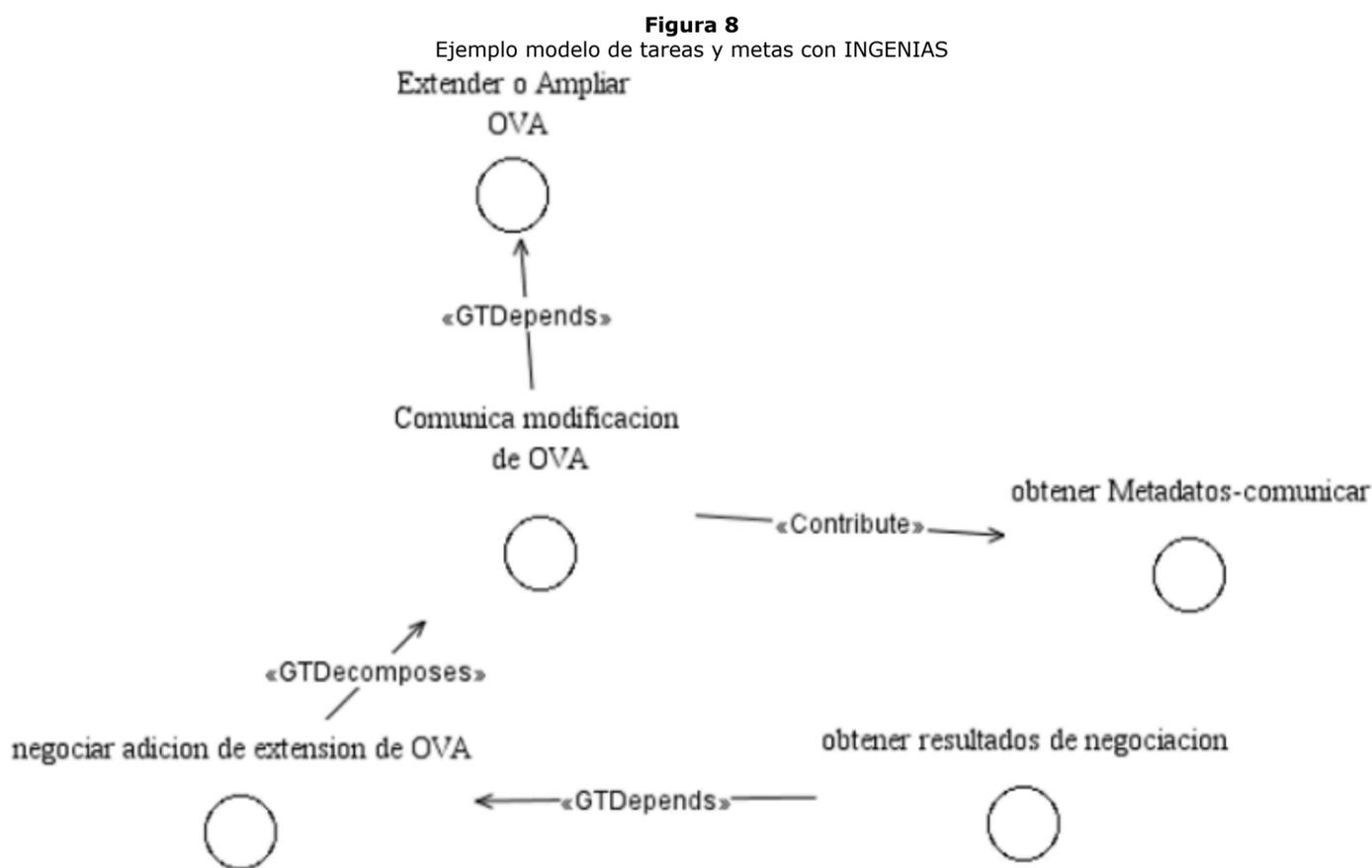
3.4.2.2 *Modelo de organización*: La figura 7 muestra un ejemplo del modelo de organización que es parte de uno de los casos de estudio tenidos en cuenta para la caracterización.

**Figura 7**  
Ejemplo modelo de organización con INGENIAS



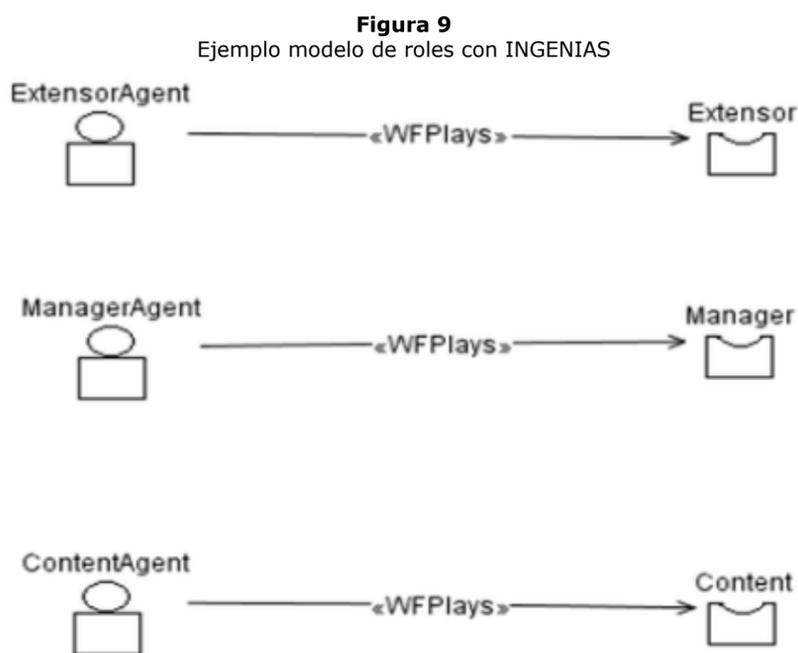
Fuente: Elaboración propia, basado en (Sanz & Mestras, 2004)

3.4.2.3 *Modelo de tareas y metas*: La figura 8 muestra un ejemplo del modelo de tareas y metas que es parte de uno de los casos de estudio tenidos en cuenta para la caracterización.



Fuente: Autor investigación, basado en (Sanz & Mestras, 2004)

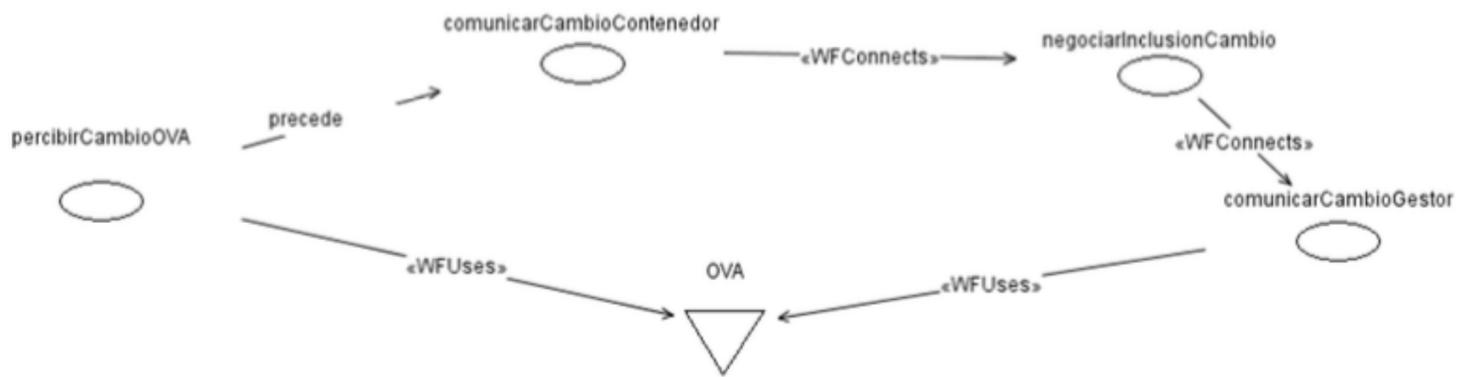
3.4.2.4 *Modelo de roles*: La figura 9 muestra un ejemplo del modelo de roles que es parte de uno de los casos de estudio tenidos en cuenta para la caracterización.



Fuente: Elaboración propia, basado en (Sanz & Mestras, 2004)

3.4.2.5 *Modelo de interacciones*: La figura 10 muestra un ejemplo del modelo de interacciones que es parte de uno de los casos de estudio tenidos en cuenta para la caracterización.

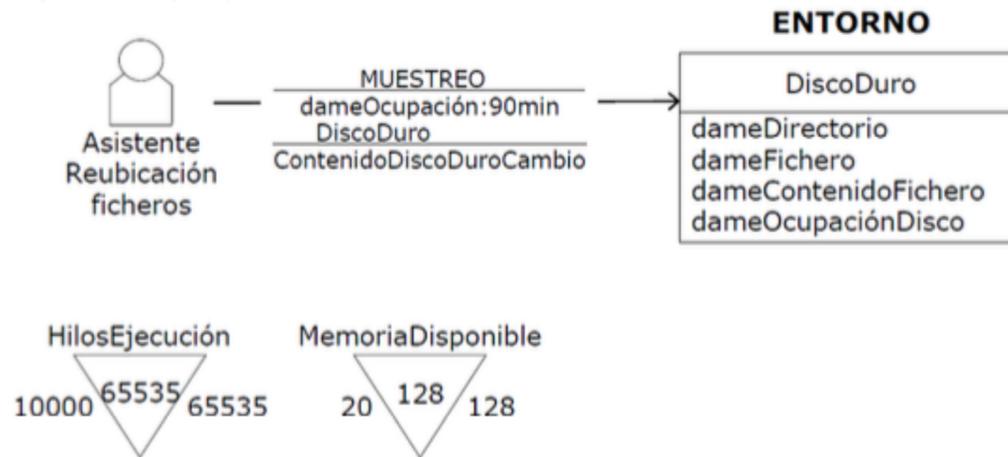
**Figura 10**  
Ejemplo modelo de interacciones con INGENIAS



Fuente: Elaboración propia, basado en (Sanz & Mestras, 2004)

3.4.2.6 *Modelo de entorno*: La figura 11 muestra un ejemplo del modelo de entorno que es parte de uno de los casos de estudio tenidos en cuenta para la caracterización.

**Figura 11**  
Ejemplo modelo de entorno con INGENIAS



Fuente: Tomado de curso de doctorado agentes inteligentes (Sanz & Mestras, 2004)

## 4. Discusión

Existen trabajos de caracterización de metodologías (Numi Tran, Low, & Williams, 2005), (Kim et al., 2012) y (Liu, 2011), pero no han tenido en cuenta INGENIAS que actualmente es una de las más acogidas. GAIA es revisada en varios trabajos (Cernuzzi et al., 2011), (Loizou, Hartley, Slater, Newman, & Pannese, 2012), (Zambonelli et al., 2003) y (Zambonelli et al., 2005) pero no es comparada con INGENIAS directamente, y el enfoque que se maneja no se detiene en las etapas de análisis y diseño.

Las etapas de análisis y diseño (Wooldridge et al., 2000) de un SMA son mucho más dispendiosas en el caso de acoger la metodología GAIA, ya que para esta metodología no se tiene un framework que apoye dichas etapas, y de esta manera poder realizar un modelado más sencillo (Cernuzzi et al., 2011).

En el caso de la implementación de un SMA con GAIA (Botia et al., 2008), (Moraitis et al., 2002) se deben usar de manera independiente (Moraitis et al., 2002) aplicaciones como JADE (Telecom Italia SpA, 2018), lo que indica que no se debe usar una herramienta específica (Numi Tran et al., 2005), (Kim et al., 2012), (Liu, 2011), (Cernuzzi et al., 2011), (Loizou et al., 2012), (Zambonelli et al., 2003) y (Zambonelli et al., 2005).

Evaluando el escenario de INGENIAS (Gomez-Sanz et al., 2008) en las etapas de análisis y diseño (García, Gómez, Gómez, & González, 2008), se encuentra la guía de un marco de trabajo (Gómez-Sanz et al., 2005) para facilitar y estandarizar estas etapas del desarrollo del SMA.

En la revisión bibliográfica realizada, se encuentra que la mayoría de trabajos son dedicados a hacer modelados (Gomez-Sanz et al., 2008), (García et al., 2008), (Gómez-Sanz et al., 2005) y (Botia et al., 2008), pero no discriminan las etapas de análisis y diseño.

A pesar de que INGENIAS en el modelo de entorno considera los recursos que va a usar el SMA, ficheros, directorios y cantidad de memoria, (Gomez-Sanz et al., 2008), en ninguno de los modelos proporciona una forma de calcular los recursos necesarios para la ejecución del SMA, lo anterior muestra un gran vacío en lo que refiere al deseo real de implementación del SMA.

GAIA por su parte no proporciona ninguna forma de calcular los recursos de implementación del SMA y tampoco de los recursos usados por el sistema. GAIA es una metodología basada en lenguaje natural y por esa razón proporciona una fácil aproximación al usuario sin mayor capacitación, es así que su guía de implementación es un corto texto que no requiere un alto grado de instrucción para poder usarlo, por el contrario, INGENIAS es una metodología orientada totalmente a la ilustración gráfica que proporciona facilidad de uso, pero también es muy subjetiva en la interpretación cuando existe cambio de usuarios.

## 5. Aportes

Como ninguna de las metodologías proporciona una forma de calcular los recursos de implementación del SMA, se provee una ecuación de cálculo de recursos físicos enfocado a la memoria de acceso aleatorio necesaria para aproximar al usuario a la implementación del SMA, esta ecuación se sugiere agregar en la etapa de diseño del modelo de agentes en GAIA ya que ese modelo incluye la cardinalidad del SMA.

**Ecuación 1**  
Calculo de memoria para implementar SMA

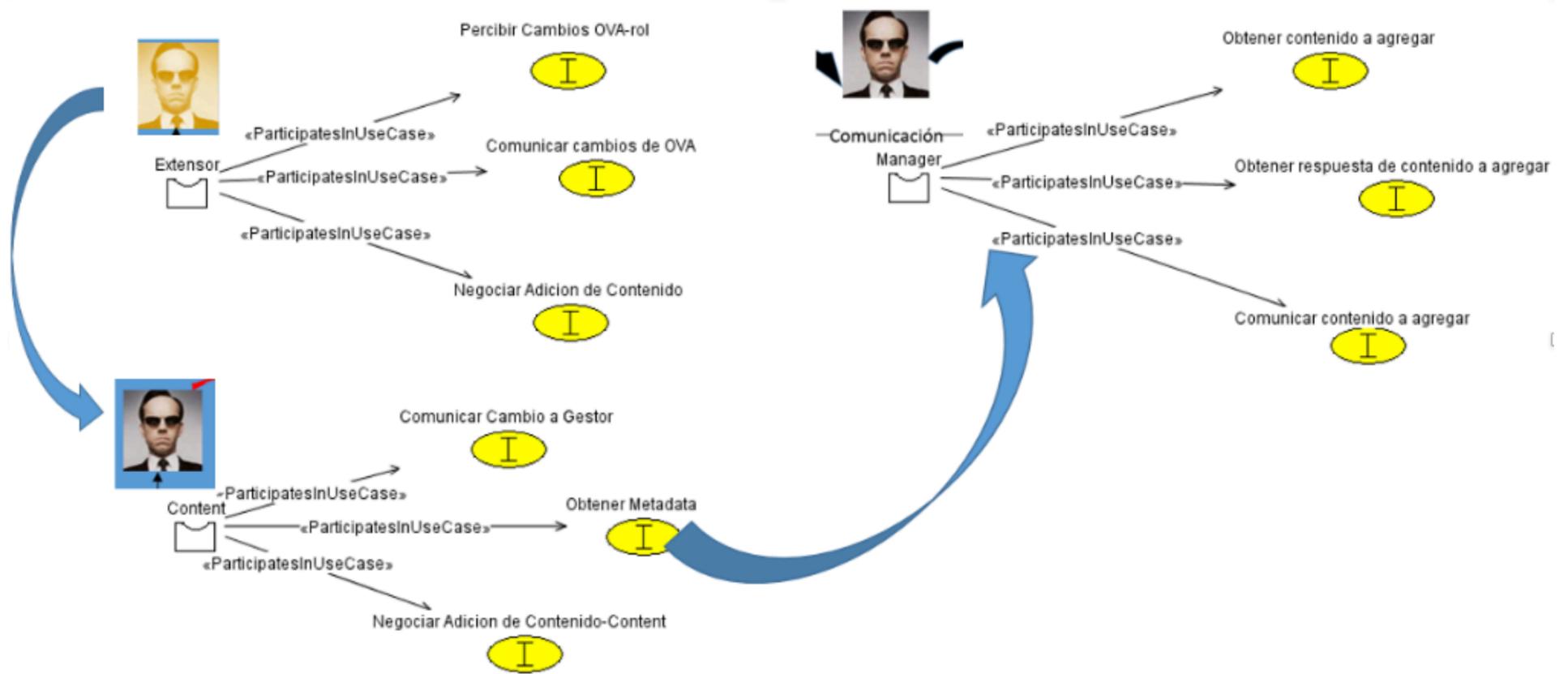
$$\text{Recursos de HW.RAM} = \sum X_i + \sum Y_i + \sum Z_i$$

X= ExtensorAg, Y= ContentAg, Z= AgentMan

Fuente: Elaboración propia

Las relaciones en los modelos de interacciones en las dos metodologías seleccionadas no son visibles, se deben incluir flechas que indiquen el flujo y así mismo el orden de las interacciones realizadas, eso se puede indicar desde la etapa de análisis como se hace en GAIA, pero como complemento al modelo de casos de uso, para obtener una perspectiva temprana de dependencia y orden de los módulos a desarrollar en el SMA. La figura 12 presenta el diagrama de interacción propuesto.

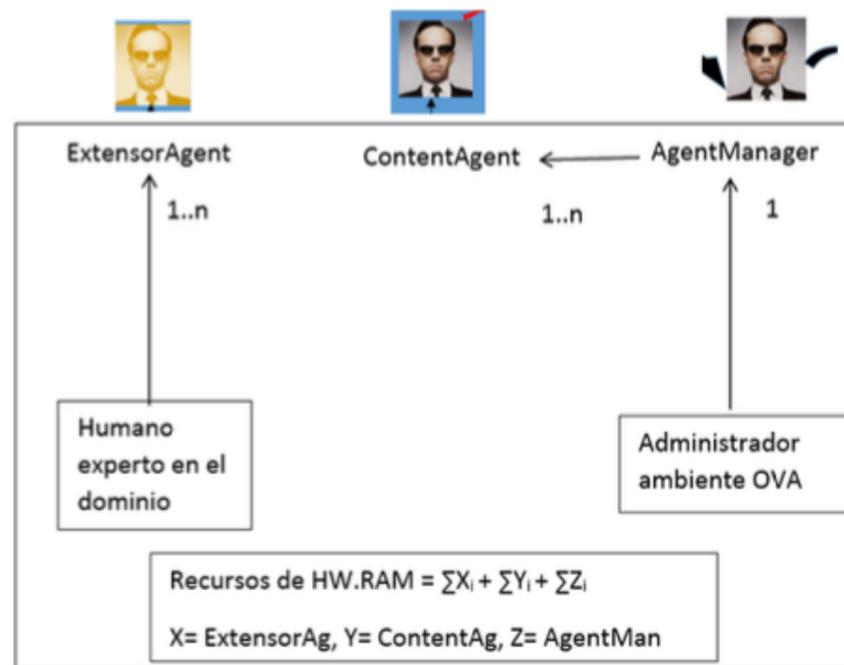
**Figura 12**  
Diagrama de interacción propuesto



Fuente: Elaboración propia

Aunque INGENIAS es una metodología fuertemente gráfica, no incluye imágenes que ilustren los roles que integra cada uno de los modelos, eso ayuda a la rápida ubicación del usuario de la metodología. La figura 13 presenta el diagrama de roles propuesto.

**Figura 13**  
Diagrama de roles propuesto



Fuente: Elaboración propia

Las dos metodologías se complementan ya que INGENIAS evidencia una fuerte representación gráfica y GAIA tiene inclinación al lenguaje natural, en la Tabla 2 se presentó la analogía de los modelos de cada una de las metodologías seleccionadas, la Tabla 2 sirve de guía para integrar las dos metodologías ya que una actúa en complemento de la otra.

En la metodología GAIA el modelo de servicios debe llevar una casilla en donde se involucre el rol al cual pertenece ese servicio específico, ya que es necesario regresar a ver el modelo de interacciones para verificar a que funcionalidad se está refiriendo y así se desubica al usuario.

## 6. Conclusiones

La integración de metodologías llena los vacíos encontrados en cada una de las ellas, así mismo las tablas que fueron fruto del estudio de revisión proporcionan el apoyo para realizar una buena integración.

La metodología INGENIAS proporciona una guía muy completa como es su marco de trabajo que facilita las etapas de análisis y diseño, pero ese marco compromete cierto grado de rigidez y no es conducente en la elaboración de los modelos de manera consecutiva, es así como siempre que se debe retomar el trabajo es necesario realizar una revisión de lo realizado anteriormente.

INGENIAS es una extensión de MESSAGE y es una metodología bastante completa ya que inicialmente cubría unos ámbitos de implementación, pero luego INGENIAS focalizada su trabajo sobre otros nichos específicos, por esa situación no comprende espacios genéricos para su uso, aunque esté basada en UML.

En INGENIAS y GAIA el modelo de interacciones presenta una debilidad ya que no se muestran flechas para orientar la interacción, lo cual confunde al usuario y por consecuencia puede recaer en resultados inesperados.

Los modelos de la metodología GAIA hacen el uso del lenguaje natural como principal herramienta de apoyo que, si bien es buena, riñe con la palabra modelo.

En la metodología GAIA se encuentra una clara inclinación al lenguaje natural, manteniendo su simplicidad y fortaleciendo el enfoque a modelos organizacionales y conduciendo con sus modelos a simplificar el proceso.

La metodología GAIA posee el modelo de conocimiento que es muy simple y no le aporta a la creación de un SMA puesto que en el modelo de interacciones se evidencia el conocimiento que deben tener los roles.

La metodología GAIA no incluye los recursos físicos que se van a usar, por consecuencia tampoco tienen en cuenta los recursos que son necesarios para que el SMA se pueda ejecutar, por lo tanto, se puede concluir que se está haciendo un sistema que no se tiene la certeza que se vaya a implementar.

En GAIA las etapas de análisis y diseño se pueden diferenciar sin mayor esfuerzo dado que poseen modelos específicos para cada una de ellas, eso facilita la ubicación del usuario en cualquiera de las etapas sea para subdividir o para retomar el trabajo, en el caso de estudio se verifica la necesidad de realizar revisiones; eso proporciona la facilidad de especializar el trabajo según las fortalezas que provea cada ingeniero en disciplinas como: requisitos, análisis, diseño y finalmente implementación.

En las etapas de análisis y diseño en la metodología GAIA, se observa una coherente alineación y seguimiento de los modelos organizacionales que son base para este trabajo, los roles que se encuentran como base de la metodología y además se continúa ese direccionamiento hasta el final de la metodología, mostrando el modelo de conocimiento con los roles tomados desde el análisis y con las relaciones objeto del modelo de interacciones

## Referencias bibliográficas

- Berlanga, A., Sanchis, A., Isasi, P., & Molina, J. M. (2002). Generalization capabilities of co-evolution in learning robot behavior. *Journal of Robotic Systems*, 19(10), 455–467. <https://doi.org/10.1002/rob.10054>
- Botia, J. A., Gonzalez, J. C., Gomez, J., & Pavon, J. (2008). The Ingenias Project: Methods And Tool For Developing Multiagent Systems. *IEEE Latin America Transactions*, 6(6), 529–534. <https://doi.org/10.1109/TLA.2008.4908186>
- Cambel, D. T., & Stanley, J. C. (1963). *Experimental and Quasi-experimental Designs for Research*. Experimental and Quasi-experimental Designs for Research. Boston: Houghton Mifflin Company. <https://doi.org/10.4135/9781483346229.n137>
- Cernuzzi, L., Omicini, A., Molesini, A., & Zambonelli, F. (2011). Adaptable Multi-Agent Systems: the Case of the Gaia Methodology. *International Journal of Software Engineering and Knowledge Engineering*, 21(4), 491–521. Recuperado de [https://www.researchgate.net/publication/220344726\\_Adaptable\\_Multi-Agent\\_Systems\\_the\\_Case\\_of\\_the\\_Gaia\\_Methodology](https://www.researchgate.net/publication/220344726_Adaptable_Multi-Agent_Systems_the_Case_of_the_Gaia_Methodology)
- FIPA. FIPA Abstract Architecture Specification (2002). Recuperado de [www.fipa.org/specs/fipa00001/SC00001L.html](http://www.fipa.org/specs/fipa00001/SC00001L.html)
- FIPA. FIPA Brokering Interaction Protocol Specification (2002). Recuperado de <http://fipa.org/specs/fipa00033/SC00033H.html>
- FIPA. FIPA Device Ontology Specification (2002). Recuperado de <http://www.fipa.org/specs/fipa00091/SI00091E.html>
- FIPA. FIPA Quality of Service Ontology Specification (2002). Recuperado de <http://fipa.org/specs/fipa00094/SC00094A.html>
- FIPA. FIPA Recruiting Interaction Protocol Specification (2002). Recuperado de <http://fipa.org/specs/fipa00034/SC00034H.html>
- FIPA. FIPA Agent Management Specification (2004). Recuperado de <http://fipa.org/specs/fipa00023/SC00023K.html>
- García, I., Gómez, A., Gómez, J., & González, J. C. (2008). INGENIAS-SCRUM Development Process for Multi-Agent Development. *International Symposium on Distributed Computing and Artificial Intelligence 2008*, 108–117. [https://doi.org/https://doi.org/10.1007/978-3-540-85863-8\\_14](https://doi.org/https://doi.org/10.1007/978-3-540-85863-8_14)
- Gomez-Sanz, J. J., Fuentes, R., Pavón, J., & García-Magariño, I. (2008). INGENIAS development kit: a visual multi-agent system development environment. *Proceedings of the 7th International Joint Conference on Autonomous Agents and Multiagent Systems: Demo Papers*, 1675–1676.
- Gómez-Sanz, J. J., Pavón, J., & Fuentes, R. (2005). The INGENIAS Methodology and Tools. Idea Group Inc, 236–273. Recuperado de <http://www.di.ufpb.br/clairton/DSBC/INGENIAS.pdf>
- Gómez Sanz, J. J. (2002). *Modelado de Sistemas Multi-Agente*. Universidad Complutense de Madrid. Recuperado de <http://grasia.fdi.ucm.es/jorge/tesis.pdf>
- González, E. J., Hamilton, A., Moreno, L., Marichal, G. N., & Mendez, J. A. (2006). Diseño e implementación de un sistema multiagente para la identificación y control de procesos. *Ingeniería Informática*, 12. Recuperado de [https://www.researchgate.net/publication/277265363\\_Diseño\\_e\\_implementación\\_de\\_un\\_sistema\\_multiagente\\_para\\_la\\_identificación\\_y\\_control\\_de\\_procesos](https://www.researchgate.net/publication/277265363_Diseño_e_implementación_de_un_sistema_multiagente_para_la_identificación_y_control_de_procesos)
- Guerrero, C. A., Londoño, J. M., Suárez, J. M., & Gutiérrez, L. E. (2014). Estudio comparativo de Marcos de Trabajo para el Desarrollo Software Orientado a Aspectos Comparative Study of Frameworks for the Development of Aspect-Oriented Software. *Información Tecnológica*, 25(2), 67–78. <https://doi.org/10.4067/S0718-07642014000200008>
- Hernández-Sampieri, R., Fernández-Collado, C., & Baptista Lucio, P. (2014). *Metodología de la investigación*. (E. M. Rocha, Ed.) (5th ed.). McGraw-Hill / Interamericana Editores, S.A. de C.V.
- Kim, M., Masoumzadeh, A., & Joshi, J. B. D. (2012). A survey of security issue in multi-agent systems. *Artificial Intelligence Review*, 37(3), 239–260. <https://doi.org/https://doi.org/10.1007/s10462-011-9228-8>
- Liu, G. (2011). The application of intelligent agents in libraries: a survey. *Program*, 45(1). <https://doi.org/https://doi.org/10.1108/00330331111107411>
- Loizou, M., Hartley, T., Slater, S., Newman, R., & Pannese, L. (2012). Emotions for intelligent agents in crisis management simulations: A survey. *17th International Conference on Computer Games (CGAMES)*, 213–219. <https://doi.org/10.1109/CGames.2012.6314578>
- Moraïtis, P., Petraki, E., & Spanoudakis, N. I. (2002). Engineering JADE Agents with the Gaia Methodology. *Agent Technologies, Infrastructures, Tools, and Applications for E-Services*, 77–91. [https://doi.org/10.1007/3-540-36559-1\\_8](https://doi.org/10.1007/3-540-36559-1_8)
- Norvig, P., & Russell, S. (2018). *Artificial Intelligence: A Modern Approach*. Recuperado de <http://aima.cs.berkeley.edu/>
- Numi Tran, Q.-N., Low, G., & Williams, M.-A. (2005). A Preliminary Comparative Feature Analysis of Multi-agent Systems Development Methodologies. In *Agent-Oriented Information Systems II* (pp. 157–168).
- Nwana, H. S. (1996). Software agents: an overview. *The Knowledge Engineering Review*, 11(3), 205–244.
- Odell, J. (2018). Welcome to the Foundation for Intelligent Physical Agents. Recuperado de <http://www.fipa.org/>
- Orrego, D., & Bolivar, J. (2006). Análisis del aprovechamiento de los agentes móviles en aplicaciones diseñadas para redes. Universidad Tecnológica de Pereira - UTP.
- Piedrahita Ospina, A. A. (2012). Modelo de interoperabilidad middleware para la integración de agentes móviles inteligentes con redes de sensores inalámbricos. Universidad Nacional de Colombia. Recuperado de <http://www.bdigital.unal.edu.co/6121/1/1034280312.2012.pdf>
- Rich, E., Knight, K., González Calero, P. A., & Bodega, T. (1994). *Inteligencia artificial* (2nd ed.). Madrid, España: McGraw-Hill.
- Sanz, J. G., & Mestras, J. P. (2004). Desarrollo de Sistemas Multi-Agente La metodología INGENIAS. In *Desarrollo de Sistemas Multi-Agente La metodología INGENIAS* (p. 21). Juan Pavón Mestras UCM 2004.
- Telecom Italia SpA. (2018). Introduction to JADE. Recuperado de <http://jade.tilab.com/documentation/tutorials-guides/introduction-to-jade/>
- Thangarajah, J., Padgham, L., & Harland, J. (2002). Representation and reasoning for goals in BDI agents. *Australian Computer Science Communications*, 24(1), 259–265. <https://doi.org/10.1145/563857.563831>
- Weiss, G. (2000). *Multiagent Systems: a modern approach to distributed artificial intelligence*. The MIT Press.
- Wooldridge, M. (2009). *An Introduction to Multiagent Systems* (2nd ed.). Reino Unido: John Wiley & Sons Ltd.
- Wooldridge, M., & Jennings, N. R. (1995). Agent theories, architectures, and languages: A survey (pp. 1–39). [https://doi.org/10.1007/3-540-58855-8\\_1](https://doi.org/10.1007/3-540-58855-8_1)
- Wooldridge, M., Jennings, N. R., & Kinny, D. (2000). The Gaia Methodology for Agent-Oriented Analysis and Design. *Autonomous Agents and Multi-Agent Systems*, 3(3), 285–312. <https://doi.org/10.1023/A:1010071910869>
- Zambonelli, F., Jennings, N. R., & Wooldridge, M. (2003). Developing Multiagent Systems: The Gaia Methodology. *ACM Transactions on Software Engineering and Methodology*, 12(3), 317–370. Recuperado de <https://www.cs.ox.ac.uk/people/michael.wooldridge/pubs/tosem2003.pdf>
- Zambonelli, F., Jennings, N. R., & Wooldridge, M. (2005). Multi-Agent Systems as Computational Organizations: The Gaia Methodology. Idea Group Inc, 136–171.

1. Facultad de Ingeniería de Sistemas. Universidad Santo Tomas. Tunja, Colombia. Magister en Tecnología Informática. Email: [ivan.leal@usantoto.edu.co](mailto:ivan.leal@usantoto.edu.co)
2. Programa Ingeniería de Sistemas. Unidades Tecnológicas de Santander. Bucaramanga, Colombia. Magister en Gestión, Aplicación y Desarrollo de Software. Email: [jomasupe@hotmail.com](mailto:jomasupe@hotmail.com).
3. Facultad de Ingeniería de Sistemas. Universidad Santo Tomas. Tunja, Colombia. Doctor (c) en Ingeniería. Email: [carlos.guerrero@usantoto.edu.co](mailto:carlos.guerrero@usantoto.edu.co)
4. Facultad de Ingeniería de Sistemas. Universidad Santo Tomas. Tunja, Colombia. Magister en Ingeniería área de Informática. Email: [decsistemas@ustatunja.edu.co](mailto:decsistemas@ustatunja.edu.co)