

**MULTIAGENT SYSTEM FOR THE ADMINISTRATION OF  
BIBLIOGRAPHICAL MATERIAL****SISTEMA MULTIAGENTE PARA LA GESTIÓN DE MATERIAL  
BIBLIOGRÁFICO**

**Ing. Fabian Andrés Giraldo Giraldo, MSc. Jaime Alberto Guzmán Luna,  
Ph.D. Demetrio Arturo Ovalle Carranza,**

*Universidad Nacional de Colombia Sede Medellín  
Escuela de Sistemas, Facultad de Minas, Núcleo Robledo  
fagirald@unalmed.edu.co, jaguzman@unalmed.edu.co, dovalle@unalmed.edu.co  
Medellín*

**Abstract:** In this article we want to show the implementation of a Multi-agent system for the administration of information of bibliographical material in a bookstore applying the methodology MA-CommonKADS.

**Resumen:** En el presente artículo se pretende mostrar la implementación de un sistema Multi-Agente para la gestión de información de material bibliográfico en una comercializadora de libros mediante la aplicación de la metodología MAS-CommonKADS.

**Keywords:** Agent-Oriented Programming , Multi-Agent Systems, MAS CommonKADS, Information Retrieval, Information Technologies.

## 1. INTRODUCCION

Una de las metodologías asociadas al desarrollo de Sistemas Multi-Agente (SMA) que ha demostrado tener muy buenos resultados es MAS-CommonKADS (Iglesias, 1998) que incluye una estructura clara y completa para el desarrollo de SMA. Esta Metodología incluye un conjunto de siete modelos que cubren todos los aspectos básicos del desarrollo del esta clase de sistemas.

En este trabajo se pretende mostrar la implementación de un Sistema Multiagente para la Gestión de Material Bibliográfico empleando la

metodología MAS-CommonKADS.

Este documento esta organizado de la siguiente manera: en la siguiente sección se muestran los fundamentos de los SMAs. En la sección 3 se describe la metodología MAS-CommonKADS. En la sección 4 se detalla el desarrollo del SMA para la gestión de material bibliográfico. Por último, en la sección 5 se relacionan las conclusiones y recomendaciones del trabajo.

## 2. LOS SISTEMAS MULTI-AGENTE

La Ingeniería de Software Orientada a Agentes es un nuevo paradigma de desarrollo de software cuyo principal fundamento es el desarrollo de entidades autónomas o Agentes con la capacidad de desarrollar tareas en nombre de los usuarios de una forma automática (Mosquera, 2001). No existe una definición universalmente aceptada de agente. Para el caso de este trabajo un Agente puede ser definido como una aplicación informática con la capacidad de decidir como debe actuar para alcanzar sus objetivos (Ferber, 1999). Las cualidades básicas que debe tener un agente inteligente son (Hendler 2002): Ser Comunicativo, Autónomo y Adaptable.

La Inteligencia Artificial Distribuida (IAD) se ha definido como un subcampo de la Inteligencia Artificial (IA) que se centra en los comportamientos inteligentes colectivos que son producto de la cooperación de diversos Agentes, conformando los Sistemas Multi-Agente (SMA). Las características principales de estos sistemas son (Jennings, 1998): cada agente del sistema tiene un punto de vista limitado (no tiene la información completa); no existe un control global para todo el sistema; los datos están descentralizados; la computación es asíncrona y por último permite su inter-operación con sistemas existentes.

Dado que actualmente los SMA son el centro de interés de muchas ramas de la ingeniería y en especial en la Inteligencia Artificial, se han propuesto diversas metodologías que guían el proceso de construcción de tales sistemas partiendo desde la conceptualización, pasando por el análisis y diseño y terminando con la implementación.

## 3. METODOLOGÍA MAS- CommonKADS

Es una metodología que ayuda a la construcción de SMA coherentes, completos, robustos y con alto grado de modularidad (Jennings, 1998).

La metodología MAS-CommonKADS (Iglesias, 1998), fue desarrollada como una extensión de la metodología de ingeniería del conocimiento CommonKADS (Scheober, 1994), agregando técnicas orientadas a objetos. Esta metodología gira entorno al modelo de la experiencia (Conocimiento) y esta pensada para desarrollar

Sistemas Expertos (SE) que interactúan con el usuario. Esta metodología se divide en tres fases: Conceptualización, Análisis y Diseño.

a) Fase de conceptualización: El objetivo de esta fase es la definición y el alcance funcional del proyecto. Los principales resultados de esta fase son la identificación del problema a resolver desde el punto de vista del usuario, la identificación de los actores que interactúan con el sistema y la determinación de los casos de uso que ayudan a entender los requisitos del sistema.

b) Fase de análisis: Esta fase propone seis modelos para la definición del sistema: el modelo del agente, el modelo de tareas, el modelo de la experiencia, el modelo de coordinación, el modelo de Comunicación, el modelo de la organización.

c) Fase de diseño: En esta fase se define como transformar las descripciones de los modelos de la fase de análisis en un sistema real. Para este caso MAS-CommonKADS propone el modelo de diseño, en donde además de todas las tareas de diseño clásico contempla la tarea del diseño de: los agentes, las bases de conocimiento, las características de red (En caso de que los agentes se muevan sobre alguna), y los protocolos de comunicación.

## 4. EL SMA DE GESTIÓN DE MATERIAL BIBLIOGRÁFICO

La aplicación que se desarrolló consiste en un sistema de agentes que da soporte a la gestión de material bibliográfico en una comercializadora de libros. Dado el tamaño de la aplicación nos dedicaremos a mostrar cierta parte de la metodología donde se presentaran los aspectos más importantes de cada modelo.

### 4.1 Fase de Conceptualización.

- **Problema a resolver:** Cuando se desea mantener, recuperar y supervisar la información del material bibliográfico que maneja una comercializadora de libros, en muchos casos nos vemos en dificultades para encontrar el material pertinente que se acomode a las diferentes necesidades de los procesos que requiere la empresa.

El SMA a desarrollar pretende facilitar a los administradores de la comercializadora introducir información relevante acerca del material bibliográfico que sea adquirido, realizar búsquedas que ayuden a encontrar libros que sean de su interés y por último permitir la diseminación selectiva de la información referente al nuevo material bibliográfico que posee la comercializadora.

- Dominio de la aplicación: Gestión y recuperación de material bibliográfico.

- Identificación de los actores: Actores externos: Usuarios, administradores. Actores internos: Supervisor, Indexador, Usuario, Administrador. En la tabla 1 se presenta la información correspondiente al actor usuario.

Tabla 1. Plantilla textual actor Usuario

<b>Actor:</b> Usuario
<b>Descripción:</b> El Usuario es la persona que esta interesada en la compra de material bibliográfico en la comercializadora de libros. Realiza operaciones de búsqueda y supervisión bibliográfica..

- Identificación y descripción de casos de Uso: La descripción de los casos de uso se registró en diferentes plantillas como la mostrada en la tabla 2.

Tabla 2. Plantilla textual caso de uso buscar material bibliográfico

<b>Caso de Uso:</b> Buscar material bibliografico
<b>Nombre:</b> Buscar material bibliografico
<b>Resumen:</b> Usuario ingresa datos claves que permitan encontrar libros de su interes
<b>Actores:</b> Usuario.
<b>Precondición:</b> El usuario desea buscar informacion.
<b>Descripción:</b> El usuario ingresa los campos especificados para realizar una busqueda de material bibliografico y da la orden de iniciar busqueda
<b>Excepciones:</b> Ninguna
<b>PostCondiciones:</b> Usuario recibe los resultados de la informacion requerida en un formato entendible por él.

#### 4.2 Fase de Análisis

- Modelo del Agente: En el sistema se identificaron el agente Usuario, agente Administrador, agente Indexador y el agente Supervisor. En la tabla 3 se presenta la plantilla propuesta por la metodología para la descripción del agente Indexador.

Tabla 3. Plantilla textual del agente Indexador

<b>Agente:</b> Indexador			
<b>Objetivos</b>	<b>Planes</b>	<b>Conocimiento</b>	<b>Servicio</b>
Recibir información enviada por agente administrador.	Esperar a que el agente administrador envíen resultados, y capturarlos	Tipo de información enviada por agente administrador.	Conocer material bibliográfico disponible en la empresa
Resolver consultas enviadas por agentes de usuarios.	Capturar las necesidades de información y buscar material que satisface las necesidades.	Información requerida por el usuario. Base de recursos del propio agente.	Buscar recursos útiles para los usuarios de la empresa

- Modelo de Tareas: Se identificaron y describieron las tareas que deben realizar los agentes en el sistema mediante el uso de diagramas de actividades y plantillas textuales. Una descripción de una de las tareas la podemos ver en la tabla 4.

Tabla 4. Plantilla textual para la tarea

<p><b>Tarea:</b> Recibir resultados de la operación realizado por Agente Administrador.</p> <p><b>Objetivo:</b> Conocer información sobre el los libros disponible en la comercializadora y los atributos mas determinantes de cada uno de ellos.</p> <p><b>Descripción:</b> Agente Indexador espera hasta recibir un mensaje del agente Administrador, cuando esto sucede inmediatamente trata de identificar cual es el informe de resultados (Identificación los libros y atributos mas determinantes de cada uno) enviados por este último.</p> <p><b>Entrada:</b> Mensaje enviado por agente Administrador.</p> <p><b>Salida:</b> Conocer libros que dispone la comercializadora para la venta.</p> <p><b>Precondiciones:</b> Identificación del mensaje enviado por Agente Información</p>
--

- Modelo de la Organización: De acuerdo a la metodología, se identificaron la estructura organizacional y las relaciones de herencia del sistema (Arenas 2003) como se detalla en la tabla 5.

Tabla 5. Identificación de la estructura organizativa del sistema

Clase	Subclase	Función
Agente Usuario		capturar requerimientos y presentación de resultados acerca del material bibliográfico
Agente Administrador		Interactuar con el usuario administrador para la captura de la información relacionada con el material bibliográfico
Agente Indexador	Indexador	Almacenar y recuperar información del material

		bibliográfico de la base de datos
Agente Supervisor	Supervisor	Diseminar selectivamente la información del material bibliográfico

- **Modelo de Coordinación:** Este modelo permite especificar las relaciones dinámicas existentes entre los agentes que conforman el sistema Multi-Agentes. La descripción de las conversaciones se realiza de forma gráfica y textual. Para la representación gráfica se emplean los diagramas de casos de uso como se muestra en la figura 2.

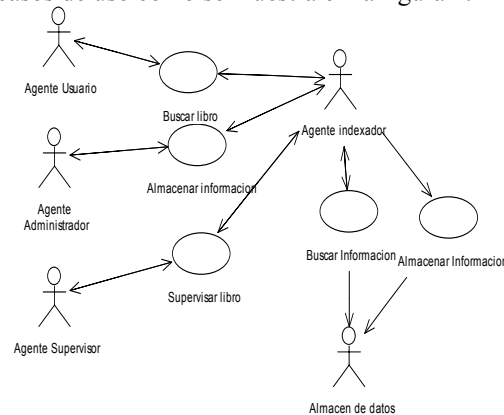


Fig. 2. Diagrama casos de uso internos del sistema

Las conversaciones están compuestas por elementos denominados intervenciones. Para su modelamiento se utilizan los diagramas de secuencia ofrecidos en la notación UML (Rumbaugh, 1999) como se muestra para el caso específico de la conversación “buscar material bibliográfico” de la figura 3.

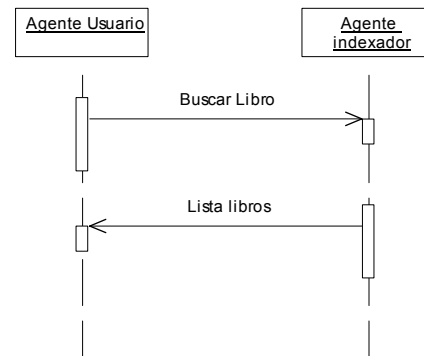


Fig. 3. Diagrama de secuencia para la conversación “buscar material bibliográfico”

Para llevar a cabo la descripción de las intervenciones se utiliza el lenguaje de comunicación KQML (Finn, 1997), utilizado para representar el flujo de comunicación entre un agente y otro como se detalla en la tabla 6.

**Tabla 6 Descripción de la intervención buscar material bibliográfico.**

<b>1) Agente Usuario envía Buscar libro a Agente Indexador</b>	
(ask one)	
:sender	Agente Usuario
:receiver	Agente Indexador
:in-reply-to	
:reply-with id0	
:language	KQML
:ontology	Libros
:content	"Buscar (titulo nombre temática titulo autor)"
<b>2) Agente Indexador lista libros Agente Usuario</b>	
(tell)	
:sender	Agente Indexador
:receiver	Agente Usuario
:in-reply-to	id0
:reply-with	
:language	KQML
:ontology	Libros
content	"libro (Titulo, Autor, Editorial, Lengua, Encuadernación, Número paginas, Resumen, precio)"

• **El Modelo de Comunicación:** El modelo de comunicación involucra el modelado de las interacciones entre los agentes humanos y el sistema mediante el uso de diagramas de casos de uso y plantilla textuales como se muestra en la figura 4.

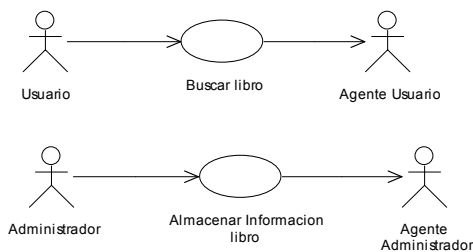


Fig. 4. Diagrama casos de uso modelo comunicación

• **Modelo de la Experiencia:** El modelo de la experiencia es el modelo que identifica a MAS – CommonKADS entre las demás metodologías de desarrollo orientadas a agentes. Este modelo incluye la identificación, descripción y estructuración del conocimiento que requieren los agentes para realizar sus tareas. Una de las actividades es el desarrollo del conocimiento del

dominio en el cual se identifican todos los conceptos mediante la descripción textual de este modelo utilizando una plantilla que describe el diagrama de conceptos (ver tabla 7).

**Tabla 7. Plantilla de diagrama de conceptos**

<b>Concepto:</b> Libro
<b>Descripción:</b> Concepto que representa todos los libros que se venden en la comercializadora.
<b>Propiedades:</b> Título, Autor, Editorial, Lengua, Encuadernación, Numero páginas, ISBN, Resumen, Tipo Ilustración, Número de Edición, Año de Edición, Plaza de Edición, Precio.

#### 4.3 Fase de diseño

Se centra en el desarrollo del *modelo de diseño* el cual pretende transformar las especificaciones de los modelos previos para que se puedan describir mediante un lenguaje de programación.

El modelo de diseño de MAS-CommonKADS distingue tres clases de decisiones de diseño: diseño de la red, diseño de los agentes y diseño de la plataforma. En nuestro caso el diseño de la red no se contemplo. En el caso del diseño de los agentes este se realizó siguiendo una plantilla como la mostrada en la tabla 8, según MAS-CommonKADS.

**Tabla 8 Diseño de los agentes**

<b>Agente Usuario</b>
<b>Nombre:</b> Agente Usuario
<b>Lenguaje:</b> Java
<b>Tiene subsistemas:</b> Presentar formulario de consulta, Capturar consulta hecha por usuario, Armar protocolo de comunicación, Enviar información Agente Indexador, Recibir resultados Agente Indexador, Presentar informe resultados Agente Usuario.
<b>Subsistemas:</b>
<b>Nombre:</b> Armar protocolo de comunicación
<b>Tipo:</b> Comunicación
<b>Funcionalidades:</b> A partir de la consulta de usuario formular el protocolo de comunicación KQML de tal manera que otros agentes del sistema lo comprendan.
<b>Implementa:</b> Captura de consulta y formar el protocolo.

En el diseño de la plataforma se seleccionó JAVA como lenguaje de programación para la implementación de los agentes, MYSQL para el almacén de datos y los agentes de interfaz fueron construidos con JSP.

#### 4.4 La implementación

La implementación del sistema cuenta con las siguientes etapas: Construcción de la base de datos que almacena información del material bibliográfico, la creación de la interfaz de usuario para cada uno de los usuarios del sistema (usuario, administrador) y la creación del sistema multiagente.

La base de datos almacena información referente al libro como: título, autor, editorial, Lengua, ISBN, Resumen, Edición, Precio, entre otras. La creación de interfaces se realizó con JSP. La creación de los agentes Usuario, Agente Administrador, Agente Indexado, Agente Supervisor. Cada uno de estos agentes fue desarrollado en el lenguaje JAVA, gracias a las facilidades para la comunicación TCP-IP que permite implementar además de la programación en paralelo que ofrece mediante el uso de multihilos.

#### 5. CONCLUSIONES

1. La utilización de las técnicas de Programación orientada a Agentes para el modelamiento de sistemas permite plantear sistemas modulares que pueden ser reutilizados en trabajos posteriores. Brinda la posibilidad de construir sistemas que se desenvuelvan en un ambiente concreto y que se adapten a las necesidades de los usuarios.
2. Tener metodologías como MAS-CommonKADS que guíen el proceso de desarrollo de sistema se convierte en un arma fundamental para los ingenieros de software que quieran desarrollar sistemas robustos y de gran calidad.
3. Entre los trabajos futuros que se piensan desarrollar se encuentra Anexar un módulo de recomendación de información con técnicas de inteligencia artificial como: redes neuronales, razonamiento basado en casos, clusterización. A demás de realizar un portal inteligente con marcado semántico, que permita la venta de material bibliográfico a través de Internet.

#### RECONOCIMIENTO

El trabajo descrito en este artículo es apoyado por el proyecto de tesis doctoral "Modelo Distribuido y cooperativo basado en agentes ontológicos y de planificación para la composición automática de servicios Web en la Web semántica" auspiciada por conciencias, Universidad Nacional de Colombia y el Banco Mundial, en el marco del *Programa de Apoyo a la Comunidad Científica Nacional a Través de los Programas de Doctorado 2004*.

#### REFERENCIAS

- Iglesias, C. (1998). Definición de una metodología para el desarrollo de Sistemas Multi-Agentes. Tesis Doctoral. Universidad Politécnica de Madrid. 1998.
- Ferber, J (1999). Multi- Agent System. Libro completo. Addison-Wesley.
- Finn, T., Labrou, Y. and Mayfield, J. (1997) "KQML as an Agent Communication Language", in software Agents, edited by J. Bradshaw, MIT Press, Cambridge.
- Hendler, James (1999). Is There an Intelligent Agent in Your Future? <http://www.nature.com/nature/webmatters/agents/agents.html>, 1999.
- Jennings, N. R. y M. Wooldridge (1998). Applications of Intelligent Agents. Queen Mary & Westfield College. University of London.
- Mosquera Celestino (2001). Análisis y estudio experimental de herramientas basadas en Agentes. Proyecto fin de carrera. Universidad de la Coruña.
- Nwana. Nwana, H. S., Wooldridge, M. (1996). Software Agent Technologies. British Telecommunications Technology Journal 14.
- Salton, G. and Mc Gill, M.J. (1983). Introduction to Modern Information Retrieval. New York: Mc Graw-Hill Computer Series.
- Scheober A. Th, Wielinga B. L., Akkermans J.M., and Van de Velde W. (1994). CommonKADS: A comprehensive methodology for KBS development, University of Amsterdam. Netherlands.
- Rumbaugh, J., Jacobson, I. y Booch, G. (1999) El Lenguaje Unificado de Modelado. Ed. Addison Wesley, Madrid.