MODELO BASADO EN AGENTES PARA LAS ETAPAS DE RECOPILACIÓN E INTEGRACIÓN DE DATOS EN EL PROCESO DE KDD

Tesis de Maestría en Ingeniería – Ingeniería de Sistemas
Línea de Investigación en Inteligencia Artificial

DANIEL BETANCUR CALDERÓN

Director
JULIÁN MORENO CADAVID. Msc.

Universidad Nacional de Colombia – Sede Medellín
Facultad de Minas – Escuela de Sistemas

AGRADECIMIENTO

El autor expresa sus agradecimientos a:

El director de la Tesis Julián Moreno Cadavid por su excelente asesoría, acompañamiento e importantes contribuciones durante la elaboración de este trabajo.

A los profesores de la Escuela de sistemas por su apoyo en las etapas iníciales de este trabajo y el desarrollo de proyectos relacionados.

Y en general a todas las personas que a través de su apoyo incondicional y contribuciones ayudaron en la realización de este trabajo.

RESUMEN

La transformación de grandes cantidades de datos en información útil y conocimiento es una inminente necesidad para la industria y la sociedad en general. Buscando cubrir esta necesidad surge el proceso de descubrimiento de conocimiento en bases de datos (Knowledge Discovery in Databases, KDD), el cual está compuesto por varias etapas. Un conjunto de estas etapas es conocido como *preparación de datos* y en la actualidad representa la mayor parte del esfuerzo destinado en las organizaciones al proceso de KDD. Sin embargo, llevar a cabo esa preparación de datos no es una labor fácil. Primero, porque dicha preparación es una labor demasiado amplia y segundo porque las principales investigaciones académicas se han concentrado en etapas como la selección, la transformación, la limpieza y la reducción de datos, dejando un poco descuidadas las etapas de recopilación e integración de datos. Algunos esfuerzos se han realizado en los últimos años, pero han provenido principalmente del ámbito privado, por lo que los diferentes modelos creados y soluciones propuestas en su mayor parte no son de dominio público.

Teniendo en cuenta esta problemática, se propone emplear el paradigma de Sistemas Multi-agente cuyos fundamentos teóricos lo hacen adecuado al problema abordado en estas tesis, para con éste lograr incrementar la automatización y la eficiencia de los procesos involucrados buscando reducir en la medida de lo posible el esfuerzo invertido. Sin embargo antes de emplear este enfoque, fue necesaria la estructuración de estas dos etapas (Recopilación e Integración de datos) con el objetivo de poder definir y entender a fondo cada una de las tareas que intervenían en dichos procesos.

El modelo propuesto se validó mediante un caso de estudio donde se pretende integrar la información disponible de estudiantes universitarios en diversos sistemas académicos, con la finalidad de identificar factores que puedan influir en su desempeño durante el transcurso de su ciclo académico.

ABSTRACT

The transformation of large amounts of data into useful information and knowledge is an imminent need for industry and society in general. Looking for fulfill this need, the process of Knowledge Discovery in Databases (KDD) arises, which consists of several stages. A set of these stages is known as data preparation and currently represents the bulk of the effort in the organizations during the KDD process. However, conducting such a data preparation is not an easy task. First, because it is too large and second because the major academic research has focused on stages as the selection, processing, cleaning and data reduction, leaving a little neglected the stages of data collection and integration. Some efforts have been made in recent years but have come mainly from the private sphere, so many of the different models and proposed solutions are not from public domain.

Given this problem, this thesis intends to employ the paradigm of multi-agent systems whose fundamentals make it suitable to the addressed problem because it may help to increase automation and efficiency in the involved processes and to reduce the possible the effort. But before using this approach, it was necessary a methodological structure of these two stages (data collection and integration) with the aim of being able of present a clear to define and understand clearly each of the tasks involved in such a processes.

The proposed model was validated with a study case which tries to integrate the available information of college students in various academic systems with the aim of identifying factors that may affect their performance during the course of his academic cycle.

TABLA DE CONTENIDO

LISTADO DE FIGURAS

LISTADO DE TABLAS

CAPITULO 1

INTRODUCCIÓN

1.1 Introducción

El desarrollo de la tecnología ha permitido en la industria de la información y en la sociedad en general la generación y el almacenamiento de grandes cantidades de datos. La transformación de estos datos en información útil y conocimiento es una inminente necesidad, debido a la gran utilidad que proporciona la adquisición de conocimiento desconocido o la corroboración de creencias sobre un entorno. Con el fin de suplir esta falencia surge el proceso de descubrimiento de conocimiento en bases de datos (Knowledge Discovery in Databases, KDD).

Este proceso iterativo e interactivo consta de varias etapas que envuelven desde la preparación de los datos hasta la interpretación de patrones descubiertos. Sin embargo durante el desarrollo de KDD, la mayor parte del tiempo y trabajo se concentra en la preparación de los datos, evitando que se destine un mayor esfuerzo a las etapas de adquisición de conocimiento tales como Minería de datos y evaluación e interpretación de patrones, etapas que representan fundamentalmente el objetivo del KDD.

Hasta hace varios años lo principales avances en investigación en KDD se centraban en las técnicas de Minería de datos. Ya en los años recientes debido al hecho de que mucho trabajo en el campo de minería de datos está basado en la existencia de datos de calidad (calidad que no se tiene generalmente, debido a la naturaleza de los datos producidos en mundo real) se comenzaron a realizar grandes esfuerzos en el área de la Preparación de los datos, con el fin de obtener precisamente datos de calidad adecuados para la aplicación de estas técnicas.

Aun así, la preparación de datos es un área muy amplia y las principales investigaciones académicas se han concentrado en las siguientes etapas: La selección, la transformación, la limpieza y la reducción de datos, dejando un poco descuidadas las etapas de Recopilación e Integración de datos. Sin embargo el desarrollo de esta últimas se ha logrado en el ámbito privado, donde se ha realizado un trabajo relativamente fuerte que ha dado grandes resultados, no obstante los diferentes modelos creados y soluciones propuestas en su mayor parte no son de dominio público.

1.2 Motivación

El incremento en la generación y almacenamiento de datos en las organizaciones ha dado lugar a que en la actualidad las empresas inviertan una gran cantidad de recursos en la recopilación e integración de datos con la finalidad de poder tomar decisiones estratégicas. Estos procesos complejos y multifacéticos, que incluso con la existencia en el mercado de herramientas especializadas que permiten reducir el esfuerzo invertido e incrementar la calidad de los resultados, aun continúan causando un gran número de dificultades, ya sea por la diversidad que presenta cada caso en específico o por el hecho de que un porcentaje de las tareas de estos procesos continúan desarrollándose de forma manual. Adicional a esto muchas de las herramientas disponibles no son de dominio público, lo que restringe el avance científico al nivel general en los procesos.

Es por tanto que la posibilidad de incrementar la automatización de estas dos etapas (Recopilación e Integración de datos), las restricciones de modificación y acceso a muchos desarrollos privados, y la capacidad de mejorar el desarrollo de las tareas manuales, son algunas de las motivaciones que fundamentan el desarrollo de esta Investigación.

1.3 Aportes

Los aportes de esta tesis son de tipo metodológico, conceptual y aplicado: En lo metodológico se estructuran las etapas de Recopilación e Integración de datos, logrando con esto incrementar la eficiencia en el desarrollo de las tareas manuales y automáticas. En lo conceptual se presentan modelos desde la Inteligencia Artificial para el desarrollo de las etapas de interés que logran complementar los trabajos presentados en las aproximaciones más conocidas y que también incrementan los niveles de automatización. En lo aplicado se muestra la efectividad de un prototipo basado en el paradigma de Sistemas Multi-agente dentro del contexto de las etapas iníciales de un proceso de minería de datos para la identificación de patrones influyentes en el desempeño de estudiantes universitarios durante el transcurso de su ciclo académico.

1.4 Definición del problema

Cuando se quiere desarrollar un proceso de KDD generalmente se busca realizar tareas complejas como análisis, planificación y predicción para la ayuda a la toma de decisiones a mediano o largo plazo. Para poder llevar a cabo este proceso usualmente los datos de una sola base de datos transaccional no son suficientes y

se debe recurrir a datos pertenecientes a diferentes organizaciones, a distintos departamentos de la misma organización, incluso puede ocurrir que algunos datos necesarios nunca hayan sido recopilados en el entorno de la organización por no ser necesarios para sus aplicaciones, y se tengan que adquirir de bases de datos públicas o privadas, o incluso por medio del internet [Hernández et al. 2004]. Luego de esto se debe llevar los datos a elementos temporales en donde serán transformados para luego ser integrados en una fuente común. Todo esto representa quizás el reto más grande y costoso del proceso de KDD, gastando alrededor del 40% del presupuesto destinado al mismo [Bernstein & Haas, 2008].

Este costo se debe a que cada paso de estas dos etapas es a menudo complejo y sensible a cualquier tipo de cambio, requiriendo una gran intervención manual y por tanto haciendo parecer que la participación humana es inevitable. Esto no quiere decir que un mayor nivel de automatización en estas etapas y por tanto una reducción del esfuerzo requerido no sea posible.

1.5 Preguntas de Investigación

De acuerdo a la problemática anteriormente mencionada, y considerando las limitaciones encontradas en los trabajos revisados en el estado del arte, surgen las siguientes preguntas de investigación:

- ¿Cuáles son los problemas más relevantes que se presentan en las etapas de recopilación e integración de datos, y que ocasionan grandes gastos de tiempo y esfuerzo en el desarrollo del proceso de KDD?
- ¿Es posible diseñar un modelo que además de permitir la automatización de algunos pasos de estas etapas, también logre integrar el desarrollo de las mismas produciendo una reducción considerable en el esfuerzo humano asignado?
- Teniendo en cuenta la naturaleza de los Sistemas Multi-Agente para el trabajo paralelo-distribuido, no centralizado ¿Cuales son las ventajas que podría ofrecer este enfoque al desarrollo de las etapas de recopilación e integración masiva de datos?
- ¿Qué es necesario tener en cuenta para diseñar un modelo adaptable a cambios, que permita solucionar el problema actual de la fragilidad del proceso de integración ante alteraciones en las fuentes?

1.6 Objetivos

General

Construir un modelo basado en agentes de software para dar soporte a las etapas de recopilación e integración del proceso de KDD maximizando en lo posible su automatización y validándolo mediante un prototipo en un caso de estudio apropiado.

Específicos

1. Caracterizar las actividades que conforman las etapas de recopilación e integración de datos e identificar sus niveles de posible automatización.
2. Identificar los problemas más relevantes que se pueden presentar en las etapas analizadas y describir las características que las posibles soluciones deberían cumplir.
3. Diseñar un modelo basado en agentes que permita dar solución a los problemas encontrados que sean más susceptibles a ser automatizados.
4. Validar el modelo propuesto mediante un caso de estudio con diversas fuentes de información.

1.7 Alcance

Con respecto a las etapas de Recopilación e Integración de Datos se tienen tareas manuales, semiautomáticas y automáticas, todas serán abordadas desde un punto de vista investigativo con la finalidad de mejorar el desarrollo de las mismas. Sin embargo dentro de la propuesta presentada, incluyendo el prototipo de validación, se tendrá como supuesto que las actividades manuales ya están realizadas y este se concentrará en el desarrollo de las tareas semiautomáticas y automáticas.

En cuanto a la automatización se deja en claro que aunque existen muchas actividades que pueden ser automatizadas, las limitaciones y complejidad que presentan las etapas de recopilación e integración de datos, dejan entre ver que la automatización de las mismas en varios casos no es posible en su totalidad y se debe tener presente la participación humana, como un componente ineludible.

1.8 Metodología de trabajo

A través de este numeral se presentan las fases que conforman la metodología de trabajo con sus respectivos pasos.

Caracterización de las Etapas de Recopilación e Integración de datos

Con esta fase se busca analizar a fondo las etapas de recopilación e integración de datos. En el desarrollo de esta se revisará la literatura buscando lograr definir adecuadamente la estructura de cada una de ellas, teniendo en cuenta la especificación clara de cada una de las actividades que conforman las mismas.

Durante esta fase también se analizarán las actividades que han sido automatizadas en la literatura y los métodos utilizados para realizar las mismas. También se logrará identificar con claridad el alcance de tanto la etapa de Recopilación como la de Integración de datos

Esta fase toma en cuenta:

- Identificación de niveles de automatización: Este proceso consiste en definir el grado de automatización que poseen las diferentes actividades de las etapas de interés que son automatizables o semi-automatizables. También se identificarán las actividades que pueden ser realizadas por medio de procesos asistidos y se contemplará la posibilidad de incluirlas en la investigación.
- Definición e identificación de los problemas que se presentan en el desarrollo de las etapas: Con este proceso se busca caracterizar los problemas encontrados, a su vez se buscará si se les ha dado solución anteriormente y en los casos en los que se encuentren soluciones, éstas serán analizadas.
- Definición de las características de las soluciones propuestas a los problemas identificados: En este proceso se estudiarán las soluciones encontradas y se definirán los requerimientos de las posibles soluciones escogidas.

Diseño del modelo basado en agentes

- Estudio de las metodologías para el desarrollo de agentes de software: Se estudiarán las metodologías más relevantes en el desarrollo de agentes y se seleccionará una adecuada para el desarrollo de la investigación.
- Selección de los modelos a tener en cuenta en la metodología: Se definirán que modelos que sugiere la metodología se construirán y cuáles no, esto

debido a que en ocasiones algunos modelos no son relevantes para los objetivos de investigación. También en caso de que se requieran modelos de otras metodologías se caracterizarán y se documentarán los motivos de la inclusión.

- Construcción de los modelos escogidos para el desarrollo del Sistema Multi-agente: Se desarrollan todos los procesos que conforman los diferentes modelos escogidos.

Validación del modelo propuesto

- Identificación y estudio de un caso de aplicación en el cual se pueda implementar el modelo propuesto: Se buscará un caso de estudio en el cual se haga necesario la realización de las etapas de recopilación e integración de datos, teniendo en cuenta la existencia de múltiples fuentes y la presentación de los problemas a los cuales se les dio solución. Luego se debe realizar un estudio profundo del caso de aplicación para familiarizarse con el mismo.
- Selección de las herramientas a utilizar para la implementación: Se escogerán y estudiarán herramientas adecuadas que permitan la implementación del modelo basado en agentes inteligentes y la interacción pertinente con el caso de aplicación seleccionado.
- Implantación de un prototipo del modelo propuesto: En este proceso se hace uso de las herramientas seleccionadas para desarrollar un prototipo que implementa tanto el trabajo realizado en las etapas de diseño y el análisis del modelo propuesto.
- Revisión de implementación: Con éste se busca hacer un chequeo que permita identificar la adecuada construcción del prototipo con respecto al modelo propuesto.
- Análisis de resultados: Con este proceso se pretende identificar si el prototipo presenta una adecuada solución a los problemas y demás componentes automatizados, generando indicadores de eficiencia y de cubrimiento de problemas.
- Generación conclusiones sobre el modelo propuesto: Se concluirá sobre los resultados obtenidos, y se realizara un comparativo entre los objetivos de investigación deseados y lo logrado.
- Redacción de informes: Este proceso documentará los aportes investigativos realizados y la estructuración y desarrollo de todo el transcurso de la investigación.

1.9 Organización del Documento

La organización del resto del documento continúa de la siguiente manera: en el Capítulo 2 se presentan los antecedentes y el marco teórico de la problemática a tratar y los enfoques más comunes para atacarla, así como una descripción del estado del arte; en el Capítulo 3 se describe una propuesta de estructuración para las etapas de interés; en el Capítulo 4 se presenta el análisis y diseño del modelo propuesto; en el Capitulo 5 se presenta la validación del modelo propuesto y del prototipo respectivo por medio de un caso de estudio apropiado; en el Capítulo 6 se presentan las conclusiones y trabajo futuro. Adicionalmente, en el Anexo A se presentan algunos detalles técnicos sobre las plataformas y herramientas empleadas para la construcción de un prototipo de software que soporta el modelo propuesto.

1.10 Difusión de Resultados

A continuación se muestran las publicaciones y ponencias en congresos que se han realizado a la fecha a raíz de los resultados obtenidos en la realización de esta tesis de maestría.

"Hacia un Modelo basado en Agentes de Software para las etapas de Recopilación e Integración de datos en el proceso de KDD". Capítulo del Libro, Tendencias en Ingeniería de Software e Inteligencia Artificial – Volumen 3 IBSN: 978-958-44-1344-4. 2009

"Estructuración Metodológica para el Desarrollo de la Etapa de Recopilación de Datos del proceso KDD". Encuentro Nacional de Investigación en Posgrados – (ENIP 2009). 2009

"Modelo Multi-Agente para el Desarrollo de las Etapas de Integración y Recopilación Datos en el Proceso KDD". V Congreso Colombiano de Computación. 2010 (En Revisión)

CAPITULO 2

APROXIMACIONES PARA EL DESARROLLO DE LAS ETAPAS DE INTEGRACIÓN Y RECOPILACIÓN DE DATOS

El objetivo de este capítulo es presentar el marco teórico y el estado del arte que dan contexto a esta investigación y justifican su realización. Inicialmente se presentan algunas generalidades sobre el proceso de KDD y las etapas de interés (Recopilación e Integración de datos). Luego se presentan algunas aproximaciones que han sido utilizadas para dar solución al problema resaltando sus características y su aplicabilidad y finalmente se presentan algunos de los trabajos más representativos de cada aproximación.

2.1 Generalidades sobre el proceso de KDD

El KDD puede definirse como un proceso no trivial que busca identificar patrones válidos, novedosos, potencialmente útiles y en última instancia comprensibles a partir de los datos [Fayyad et al, 1996] [Cios et al, 2007]. A continuación en la Figura 2.1 se muestran las principales etapas del proceso de KDD.

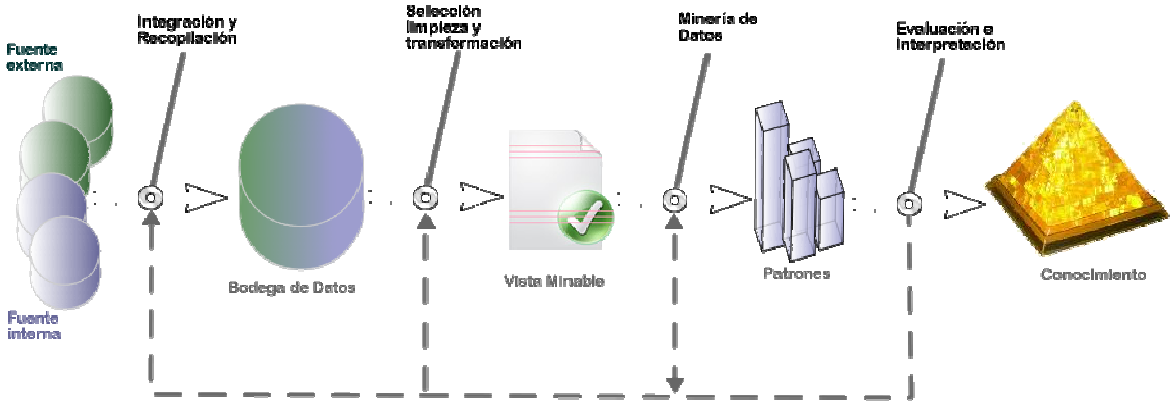


Figura 2.1 Etapas del proceso de KDD

La Etapa 1 (Integración y Recopilación) y la Etapa 2 (Selección, limpieza y Transformación) son conocidas con el nombre de preparación de datos y son

técnicas concernientes con el análisis de datos en bruto y que conllevan a producir calidad en los datos [Zhang & Zhang, 2003].

La recopilación de datos consiste en identificar las fuentes de información útiles, descubrir dónde encontrarlas y como accederlas [Hernández et al. 2004]. Por otro lado la integración de datos busca transformar los datos de diferentes fuentes en un formato común [Bernstein & Haas, 2008], frecuentemente mediante una bodega de datos que consiga unificar de manera operativa toda la información recogida, detectando y resolviendo las inconsistencias.

Ya en la etapa de selección, limpieza y transformación, se tratan los datos incorrectos y se decide la estrategia a seguir con los datos faltantes dependiendo del caso. También se realiza una selección que busca considerar únicamente aquellas variables o atributos que para el caso en particular son relevantes, con el objetivo de facilitar la tarea de minería y a su vez lograr mejores resultados.

En la etapa de minería de datos, se decide cual es la tarea a realizar (clasificar, agrupar, etc.) y se elige el método que se va utilizar. En la fase de evaluación e interpretación se evalúan los patrones y se analizan por los expertos. Esto incluye resolver los conflictos con el conocimiento que se disponía anteriormente.

Para esta investigación se tienen como etapas de interés las etapas de recopilación e Integración datos, éstas se exponen de una manera más extensa a continuación.

Recopilación de datos: Como se vio en el principio de esta sección la recopilación de datos consiste en adquirir los datos necesarios para realizar el proceso de KDD, los datos pueden provenir tanto de fuentes internas como de externas y puedes ser estructurados (Bases de datos) o no estructurados (Internet), generalmente estas información debe ser llevada a un base de datos temporal en la cual se le realizan algunas transformaciones para luego ser integrada a la bodega de datos [Hernández et al. 2004]. Los procesos identificados en esta etapa son:

- Identificación de datos requeridos: Busca analizar e identificar cuáles datos son requeridos para realizar el proceso de KDD.
- Selección de las fuentes de datos: Consiste en seleccionar las fuentes en las cuales es posible y se desea adquirir los datos identificados.
- Adquisición de los datos sobre las fuentes: Se obtiene información relevante sobre la fuente: Estructura, Métodos de acceso e Información sobre datos de interés (Formato, grados de agregación, etc.).

Integración de datos: Con esta etapa se que busca unificar y adaptar datos de diferentes fuentes o de una misma fuente en un almacén de datos coherente. Los principales tópicos identificados por [Han & Kamber, 2006] en esta fase son:

- Identificación de la entidad: Consiste en hace coincidir entidades equivalentes en el mundo real, provenientes de múltiples fuentes de datos. Busca por ejemplo hacer que un analista de datos o un computador este seguro que id_consumidor en una base de datos y cedula_consumidor en otra se refieren a la misma entidad. Por general se logra mediante el uso de metadatos que relacionan las diferentes sintaxis de los mismos elementos.
- Control de redundancia en los datos: Un atributo puede ser redundante si puede ser derivado de otra tabla. Las inconsistencias en el nombramiento de una dimensión o atributo pueden también causar redundancias en el conjunto de datos resultante. Es decir puede la no identificación adecuada de la entidad producir claramente redundancia en los datos. Algunas redundancias pueden ser detectadas a través del análisis correlacional.
- Detección y resolución de conflictos en valores de datos: Para la misma entidad del mundo real, los valores de un atributo de diversas fuentes pueden diferir. Esto puede ser debido a las diferencias en la representación, escalamiento o codificación. Por ejemplo el atributo peso puede ser almacenado en unidades métricas en un sistema y en unidades imperiales (inglesas) en otro.

La cuidadosa integración de los datos de múltiples fuentes puede ayudar a reducir y evitar redundancias e inconsistencias en el conjunto de datos resultante. Esto puede ayudar a mejorar la precisión y la velocidad del posterior proceso de minería [Han & Kamber, 2006].

2.2 Aproximaciones basadas en Mediadores

En el enfoque basado en mediadores [Rousset, 2002], la solución propuesta para el desarrollo del 2 etapas de interés, mantiene los datos en sus fuentes y construye vistas abstractas por medio de un mediador que trata de satisfacer las consultas del usuario. La arquitectura actual de este enfoque se basa en un sistema mediador-wrappers [Goasdoué et al, 2000]. Permite la consulta de fuentes de datos distribuidas y heterogéneas. Funciona como un sistema centralizado y homogéneo, donde el mediador realiza la integración de los datos, proporcionando al usuario una visión global y homogénea del sistema. El mediador está encargado de reformular las consultas del usuario en función de los distintos contenidos de las fuentes de datos accesibles. Varios wrappers conforman a este mediador, uno para

cada fuente de datos. La arquitectura de este enfoque se puede apreciar la siguiente figura 2.2.

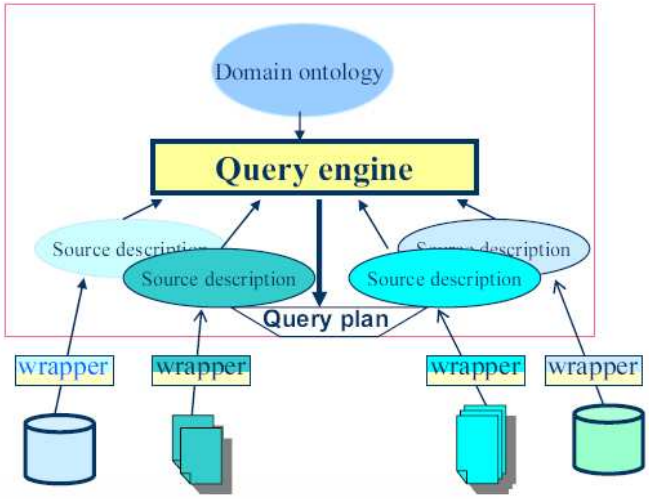


Figura 2.2 Arquitectura del Modelo Basado en Mediadores [Rousset, 2004]

2.3 Aproximaciones orientadas a Bodegas de Datos

La aproximación orientada a bodegas de datos [Inmon, 2005] consiste en la construcción de una nueva base de datos en la cual se almacenarán los datos de las diversas fuentes. En este caso las etapas de recopilación e integración corresponden al proceso de ETL (Extracting, Transforming and Loading), con el cual se limpia y se transforman los datos heterogéneos para luego ser cargados a la Bodega de Datos. Las bodegas de datos son modelos de datos orientados a análisis, donde los datos representan indicadores (medidas) que pueden ser observados de acuerdo a los ejes de análisis (dimensiones). El modelo que presentan éstas es multidimensional y caracteriza un contexto de análisis. En la figura 2.3 se presenta una ilustración de la arquitectura básica de una bodega de datos.

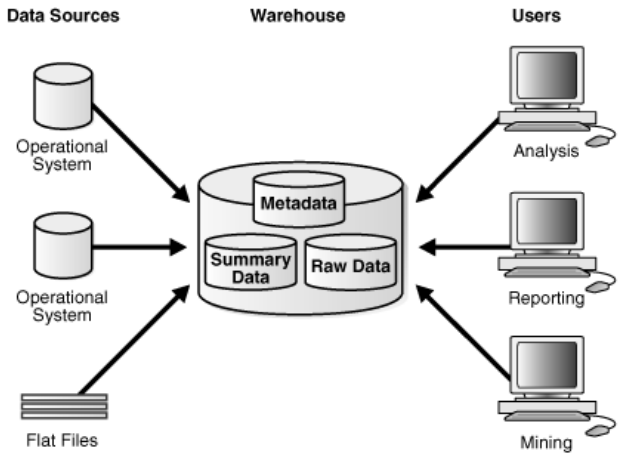


Figura 2.3 Arquitectura Básica de una Bodega de Datos [Oracle, 2008]

2.4 Aproximaciones desde el paradigma Multi-agente

Desde la perspectiva del paradigma Multi-agente [Jennings et al, 1998], donde se busca explotar la ventajas que ofrece este paradigma en cuanto al trabajo distribuido, paralelo y no centralizado, se han realizado trabajos específicamente en las áreas de interés de esta investigación y también se han desarrollado algunos que incluyen otras áreas adicionales del proceso KDD. En este ámbito algunas de las bondades que pueden ser más explotadas de los agentes son: la Adaptabilidad característica elemental dado que generalmente la diversidad de la fuentes puede requerir del aprendizaje de los agente para volver óptimo el trabajo desarrollado en las mismas; la Reactividad ya que un agente puede actuar adecuadamente tomando en cuenta los cambios que ocurren en su entorno y la ocurrencia de cambios en la Recopilación e integración es casi constate; y por último la Sociabilidad que permite a los agentes atacar de una mejor forma los problemas ocurridos mediante la comunicación de éstos a otros agentes ó incluso con otras entidades[Franklin et al, 1996].

2.5 Trabajos Relacionados

A continuación se presentan algunos de los trabajos más representativos de cada uno de los enfoques anteriormente descritos.

En la aproximación orientada a mediadores un trabajo destacado es el presentado en [Chawathe et al, 1994] conocido como el proyecto Tsimmis, el cual busca desarrollar herramientas que faciliten la rápida integración de las fuentes de información heterogéneas incluyendo tanto datos estructurados y no estructurados. En éste se ubica un wrapper en cada una de las fuentes que

convierte los conceptos de datos subyacentes a un modelo común de información. Algunos otros artículos sobre este enfoque son [Goasdoué & Rousset, 2004] y [Zhou et al, 1996].

Algunos trabajos que se han realizado sobre este enfoque ETL- Bodegas de Datos son: El trabajo propuesto en [Simitsis et al. 2004] donde se busca principalmente la optimización del flujo de trabajo de ETL, la orientación del articulo se concentra en como optimizar algoritmos que ya se encuentran automatizados, logrando que se respondan mejor a los tiempos de respuesta requeridos. Otro trabajo relevante es el presentado en [Vassiliadis et al, 2002] allí se muestra un modelo conceptual del proceso de ETL, que logra definir la actividades que se realizan en el proceso y proporcionan fundamentos formales para su representación conceptual. Una de las grandes ventajas de este modelo es que es personalizable y extensible de tal forma que el diseñador del proceso pueda enriquecerlo. Otros trabajos realizados sobre el proceso de ETL son: [Viana et al, 2005] y [Squire, 1995].

Algunos trabajo propuestos del enfoque Multi-Agente son: El propuesto por [Boussaid et al, 2003] donde se logra la integración de datos complejos, a través del uso de agentes y Bodegas de datos, plateando una arquitectura flexible y evolutiva que permite agregar o eliminar tareas de integración, entendiendo éstas como servicios ofrecidos por los agentes. El segundo trabajo es el presentado por [Imtiaz et al, 2005] allí se define un framework que usa agentes para la unificación de los procesos de extracción de información y data mining, en este trabajo los agentes son basados en modelos probabilísticos, y presentan muy buenos resultados, sin embargo la magnitud de alcance propuesto, evita un desarrollo exhaustivo de las etapas abarcadas. Adicionalmente otros trabajos que utilizan el paradigma Multi-agente y que involucran las etapas de interés son: [Kargupta et al, 1997], [Xing et al 2003] y [Di Fatta & Fortino, 2007].

Existen otros trabajos que buscan atacar problemas específicos que se presentan generalmente en la recopilación e integración de datos y que resultan independientes del enfoque que se aborde. Algunos de estos trabajos son: El presentado en [Schallehn et al, 2004] que trata los problemas de conflictos e inconsistencias a nivel de datos, esto incluye la eliminación de la duplicación de conceptos de datos causada por la superposición semántica de algunas fuentes, así como el establecimiento de una relación complementaria entre los datos de estas fuentes. La forma que se propone para afrontar estos problemas es a través de la alteración de los operadores grouping y join. Otro trabajo propuesto se expone en [Luján-Mora et al, 2001] donde se presenta un método automático para la reducción de inconsistencias en la integración de datos de diversas fuentes y a su

vez busca obtener calidad en los datos. El método trabaja por medio de algoritmos de agrupación que generan grupos por grados de similitud en cadenas de texto. Otros artículos relacionados son: [Lim et al, 1993], [Roddick & Vries, 2006] y [Haas, 2007].

2.6 Limitaciones

Las limitaciones encontradas se presentan a continuación dividas según las respectivas aproximaciones.

En el enfoque basado en mediadores.

- Existe una gran complejidad en la generación de consultas, debido a que cada consulta que se realiza debe ser reescrita para cada fuente, para luego realizar la recuperación, transformación, fusión y conciliación de los resultados arrojados por la misma.
- El hecho de que las consultas se realicen sobre las fuentes de información, produce que exista la incertidumbre de que las fuentes sean afectadas por algún cambio, es decir la eliminación total o parcial de las mismas, la modificación de los modelos de datos, etc. Al ocurrir estos cambios se afectaría considerablemente el desarrollo de las etapas de interés.

El enfoque basado en el proceso de ETL – Bodegas de datos.

- Está el problema de que el manteniendo y la realización del proceso es realmente costosa, debido a que aún requiere mucho esfuerzo humano por parte de la organizaciones.
- El hecho de que en el proceso básico de ETL los datos se toman y son guardados en una Bodega de Datos, para luego ser analizados produce que los resultados de consultas sean en algunas ocasiones basados en datos no actualizados.

El enfoque basado en el paradigma Multi-Agente

- Si bien este enfoque es prometedor dado su flexibilidad, modularidad, automatización y capacidad para aprovechar sistemas de recursos distribuidos, aún faltan muchos aportes por realizar desde áreas como la automatización y eficiencia de procesos.

En cuanto a las limitaciones en las aproximaciones que buscan atacar tópicos específicos, puede decirse que aún existen muchos problemas y actividades, a los cuales se les puede dar una solución automatizable con el fin de reducir el esfuerzo asignado a los mismos.

Como conclusión sobre la revisión del Estado del Arte se tiene que los trabajos estudiados han contribuido considerablemente al desarrollo de las etapas de recopilación e integración de datos. Sin embargo existen aun algunas falencias en ambas etapas y aún no se han logrado construir estructuras claramente especificadas, eficientes y con alto grado de automatización que permitan un desarrollo apropiado y a su vez logren una adecuada integración de estas etapas.

CAPITULO 3

ESTRUCTURACIÓN DE LAS ETAPAS DE RECOPILACIÓN E INTEGRACIÓN DE DATOS

Durante el desarrollo de este capítulo se presenta una estructuración de los procedimientos y actividades que deben considerarse para el desarrollo de las etapas de recopilación e integración de datos. El contenido de este capítulo es producto del autor, ya que en la bibliografía estudiada no se encontró una estructura definida de las tareas y procesos pertenecientes a las etapas de interés.

Para mejorar el entendimiento de este capítulo se definió la siguiente convención, con el propósito de unificar términos en los diferentes tipos de fuentes que pueden existir.

Termino	**Significado**
Concepto	Abstracción del mundo real ó Agrupación de atributos
Atributo	Característica que posee un Concepto
Registro	Instanciación de un Concepto

Tabla 3.1 Terminología utilizada

3.1 Estructuración de la Recopilación de Datos

La Recopilación de Datos para esta investigación fue dividida en dos tipos principales según la periodicidad de acceso sobre una fuente.

Recopilación Periódica: Se define como la recopilación en la cual el acceso a las fuentes se realizar de manera constante o con cierta periodicidad. Se usa principalmente en procesos de KDD reiterativos.

Recopilación Aperiódica: Se define como la recopilación en la cual el acceso a las fuentes se presenta una sola vez ó por demanda en un proceso de KDD.

Para esta etapa se definieron cuatro procesos (Figura 3.1) con los cuales se busca abarcar el desarrollo de la misma, logrando identificar y recuperar las fuentes y datos necesarios para llevar a cabo las etapas posteriores del KDD.

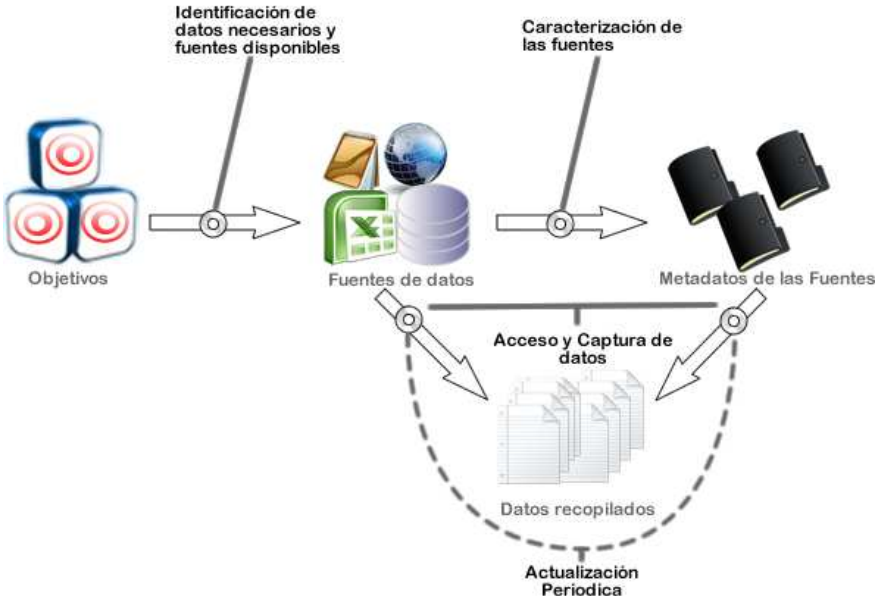


Figura 3.1 Procesos de la Etapa de Recopilación de datos.

Los primeros tres procesos son independientes al tipo de recopilación presentada y se presentan a continuación

- Identificación de Atributos necesarios y fuentes disponibles
- Caracterización de Fuentes.
- Acceso y Captura de Registros.

  Para los casos en los cuales se presenta una *Recopilación Periódica* se define un cuarto proceso: Actualización Periódica.

A continuación se presenta una descripción ampliada de cada uno de estos procesos junto con los respectivos pasos propuestos para llevarlos a cabo satisfactoriamente.

3.1.1 Identificación de Atributos necesarios y fuentes disponibles

Además de las etapas descriptas anteriormente el proceso de KDD, existe una fase previa en la que se analiza las necesidades de la organización y se presenta la definición del problema [Two Crows, 1999], en ésta se establecen también los objetivos del todo el proceso de KDD.

El primer proceso de la recopilación de datos consiste en identificar los atributos necesarios que permitan alcanzar posteriormente los objetivos propuestos y a su vez realizar un sondeo de los atributos que se tienen disponibles tanto al nivel

interior como exterior. La finalidad de este proceso es concluir cuales objetivos pueden ser alcanzados y cuáles no, que fuentes se encuentran disponibles para el desarrollo de todo el proceso y obtener a su vez una noción básica del estado los registros. A continuación una descripción de los pasos definidos para este proceso.

3.1.1.1 Atributos necesarios según Objetivos perseguidos: La realización de este paso tendrá como resultado posibles atributos por cada concepto que serán considerados como elementos informativos valiosos para la consecución de los objetivos trazados. Durante el análisis posterior de las fuentes es muy posible encontrar nuevos conceptos que contribuyan a un mejor desarrollo del proceso de KDD, debido a esto el esquema que surge a partir de esta primera vista eso solo una noción inicial. En este paso se realizan las siguientes sub-actividades.

- Construir una listado con los objetivos propuestos y sus respectivas descripciones.

  Ejemplo:

Objetivo	Descripción
1	Identificar relaciones presentes entre los productos comercializados.
2	Analizar la influencia de factores económicos sobre las ventas.
3	….

- Con base en el listado obtenido, realizar una tabla donde se muestren los posibles conceptos que puedan estar involucrados por cada objetivo.

  Ejemplo:

Objetivo	Conceptos
1	• Productos • Ventas • Clientes
2	• Precios Históricos de los Producto • Ventas • Índice de Precios al Consumidor (IPC)

Al final de esta actividad se tendrá una noción de los conceptos requeridos para llevar a cabo la etapa posterior de Minería de Datos.

3.1.1.2 *Fuentes locales:* Se identifican todas las fuentes de datos en el ámbito local de las que se puedan extraer datos de interés para el proceso de KDD. En la mayoría de los casos se recomienda tener en cuenta todas las fuentes disponibles, ya que en muchas ocasiones existirán algunas que a primera instancia podrían no tener conceptos con relación en el proceso actual, pero no se deben descartar sin previa revisión, ya que es posible encontrar conceptos de interés para los objetivos perseguidos.

Ejemplo:

Fuente	Ubicación.Tipo.Nombre
1	Local.BD.Ventas
2	Local.Doc. Finanzas
3	Local.Web.Empresa

3.1.1.3 *Conceptos disponibles en fuentes locales:* Este paso busca encontrar cuáles conceptos de los supuestos en el *paso 3.1.1.1* se encuentran disponibles dentro de las fuentes locales; también pretende identificar algunos conceptos de interés que no hayan sido tomados en cuenta inicialmente.

Dado que los registros recolectados en las organizaciones no siempre suelen poseer la calidad adecuada para un estudio, se realiza un sondeo básico del estado de los registros. Las sub-actividades que se realizan para llevar a cabo este paso son:

a. Búsqueda de Conceptos (Ámbito local): Procura encontrar los conceptos que puedan ser de utilidad dentro de las fuentes disponibles a nivel local. Se recomienda realizar este paso en todas la fuentes locales.

Ejemplo:

Fuente	Concepto Real	Concepto Supuesto
1	Producto x Venta	Ventas
	Punto de Venta	--
	Productos Industriales	Productos
	Productos de Consumo	Productos

b. Estado de Registros (Local): Tiene como propósito obtener una noción inicial del estado de los registros con los que posteriormente se va a trabajar. Para la evaluación del estado se toman características como valores atípicos, valores faltantes e inconsistencias presentes.

Ejemplo:

Fuente	Atributo	% Atípicos	% Faltantes	Grado Inconsistencias
1	Producto x Venta	0.1%	10%	Bajo
	Punto de Venta	0%	0%	Nulo

Los registros pueden ser válidos aunque necesiten procesos de limpieza dependiendo de las características encontradas y necesidades. También se puede decir si un atributo no debe ser tomado en cuenta debido a sus precarias condiciones.

3.1.1.4 *Conceptos no disponibles en fuentes locales:* Para tener una noción clara de los conceptos que no fueron encontrados en las fuentes locales o que se encuentran pero no son confiables o son obsoletos, se construye una lista con los conceptos que requieren ser buscado en fuentes externas.

Ejemplo:

Concepto no Disponible
IPC

3.1.1.5 *Fuentes externas:* Dependiendo de la naturaleza del estudio se requieren conceptos alojados en fuentes externas, estos conceptos pueden estar tanto en fuentes públicas como privadas. Para los casos donde ser requiera tener acceso a fuentes privadas es muy importante clarificar todas las condiciones de acceso, con la finalidad de lograr realizar el proceso de la mejor manera posible. Este paso consiste en adquirir y licitar todas las fuentes externas necesarias para llevar a cabo el proceso de KDD. Una vez todos los parámetros legales y operativos estén definidos se complementa la tabla presentada en el paso *3.1.1.2* con las fuentes externas.

Ejemplo:

Fuente	Ubicación.Tipo.Nombre	Acceso
4	Externa.Web.DANE	Publico

3.1.1.6 *Conceptos no disponibles:* Ya conocidos que conceptos se encuentran a nivel local y a nivel externo, se construye una tabla que permita ver que conceptos no fueron encontrados.

Ejemplo:

Objetivo	Conceptos No Encontrados
1	• Compras (Materia Prima)

3.1.1.7 *Objetivos no alcanzables y continuidad del proceso:* En este punto se toman los resultados arrojados en el paso 3.1.1.6 y se identifican que objetivos son alcanzables y cuáles no. Luego se decide si se continúa el proceso de KDD o se inicia un proceso de adquisición de los conceptos faltantes.

Ejemplo:

Objetivo	Estado	Causa
1	Alcanzable	Los registros que permiten desarrollar un estudio para llevar a cabo este Objetivo se en encuentra disponibles y en buen estado.
2	No Alcanzable	Algunos de los registros relacionados no se encuentran en el mejor estado para realizar un análisis que permita llevar a cabo la consecución de este objetivo.

3.1.2 Caracterización de las fuentes.

Luego de identificar los objetivos alcanzables, los conceptos y las fuentes relacionadas a ellos, se procede con una especificación de las mismas. Esta especificación consiste en la recopilación de toda la información necesaria para un trabajo exitoso con cada fuente ya sea local o externa.

Pasos:

3.1.2.1 *Estructura de las fuentes:* Para cada fuente se debe hacer un análisis de la estructura que permita identificar aspectos claves de trabajo con las mismas. Consecutivamente se muestran los parámetros más representativos.

Tipo de fuente: Clase de fuente según su estructura. En este trabajo se proponen los siguientes 3 tipos de fuente.

- Estructurada: Fuentes tales como bases de datos y estructuras de almacenamiento que dispongan de restricciones, métodos de alteración y ordenamiento de datos.
- Semi-estructurada: Principalmente hojas de cálculo y documentos que permitan acceso por índices de localización a cualquiera de sus datos. También archivos de texto y documentos que almacenan datos mediante separadores y órdenes planos.

- No estructurada: Fuentes en las que los datos se encuentran almacenados sin un orden predefinido, tal como información disponible en páginas web o adquirida por medio de algunos servicios web.

Ejemplo:

Fuente	Tipo
1	Estructurada
2	Semi-estructurada
3	No estructurada

Información estructural: Características generales acerca de la estructura de cada fuente que permitan identificar requerimientos para el trabajo con ellas. A continuación se muestran ejemplos para los 3 tipos de fuente definidos anteriormente.

- Estructurada: En las fuentes estructuradas se identifican características tales como: Motor gestor de la base de datos (Oracle, MySql, SqlServer) y tipo de base de datos (Relacional, Espacio-Temporal, etc).

Ejemplo:

Fuente	SGBD	Tipo
1	Oracle	Base de datos relacional

- Semi-estructurada: Tipo de documento, métodos de acceso a datos (Basados en posición o separadores), programa en el cual se generó el documento.

Ejemplo:

Fuente	Tipo	Met. de Acceso	Programa Asociado
2	Hoja de Calculo	Basado en Posición	M. Excel

- No estructurada: Tipo de documento, palabras claves, etiquetas, códigos que indiquen el inicio y el final de los registros de interés.

Ejemplo:

Fuente	Tipo	Código Inicio	Código Fin
3	HTML	<table id=t_noticias>	</table>

3.1.2.2 *Información general de las fuentes:* En este paso se obtienen información sobre las propiedades principales de cada una de las fuentes. Las propiedades que se propone tomar en cuenta se presentan seguidamente.

- Parámetros de Acceso: Ruta de Acceso, Nombre de la fuente, usuarios, contraseñas y códigos adicionales que sean necesarios para lograr acceder a la fuente.
- Ubicación: Consiste en decir si la fuente es local o externa.
- Conceptos de interés: Listado de conceptos y sus atributos que deben ser capturados de cada fuente.
- Disponibilidad de la fuente: Se toman en cuenta dos casos
    - Continua: Se puede acceder a la fuente en cualquier momento
    - Discontinua. El acceso es controlado. Para este caso se deben especificar las restricciones de disponibilidad.
- Permisos: Especifica el control que se dispone dentro de la fuente (Lectura, Escritura y Ejecución).

Ejemplo:

Id	**Nombre**	**Tipo**		**Acceso**				
		General	**Aplicación**	**Ubicación**		**Autenticación**		
1	Ventas	General	Oracle	***Dirección IP***	***Puerto***	***Tipo***	***User***	***Pass***
				10.1.12.241	8086	1	ABC	123

3.1.2.3 *Especificación de Conceptos:* En este paso se recopila toda la información posible sobre los conceptos de interés. Esta información es de gran utilidad en la construcción del esquema de integración y en la lectura de los registros. Seguidamente se muestra los componentes básicos que deben ser tomados en cuenta:

- Fuente: ID de la fuente en la que se encuentra ubicado.
- Nombre del Concepto: Nombre mediante el cual es conocido el concepto de interés en la fuente.
- Información de Acceso: Información para acceder a los registros que se encuentran asociados a este concepto (Consulta en Bases de datos, Columna en hojas de cálculo, posición en documentos planos, etc.)
- Atributos de interés: Características que son tomadas en cuenta de cada concepto.

- Tipo de Atributo: Información sobre el tipo de atributo que puede ser almacenado (Numérico, Booleano, alfanumérico, etc.)
- Formato: En caso de que el atributo tenga un formato en específico se llena este campo.
- Estado: Representa la calidad que poseen los registros segmentados por cada atributo. Para este campo tenemos 4 posibles valores: '1' - Adecuado: Cuando los registros no requieren limpieza ni transformación. '2' – Transformación: Registros que requieren cierta transformación, '3' –Limpieza: Cuando los registros requieren un proceso de limpieza antes de poder trabajar con ellos y '4' – Limpieza y transformación: Además de ser limpiados deben ser transformados. Para este punto tomar en cuenta en el literal *b de 3.1.1.3*

Ejemplo:

Id Fuente	**Concepto**							
1	Venta							
	Atributo				**Atributo**			
	Nombre	**Tipo**	**Formato**	**Estado**	**Nombre**	**Tipo**	**Formato**	**Estado**
	Id_venta	Integer		1	fecha	date	dd/mm/aa	1

3.1.3 Acceso y Captura de Registros

Este proceso es la esencia de la etapa de recopilación de datos. Sin embargo sin los procesos anteriormente expuestos se convierte en un trabajo aún más difícil y complejo. Se puede realizar de forma manual en algunos casos, sin embargo en muchos casos se dispone de tantos conceptos, registros y fuentes por lo que se recomienda su desarrollo de la forma más automática.

Este proceso es soportado por los dos procesos anteriores y tiene como finalidad la adquisición de los datos de interés de una manera automática. Seguidamente se presentan los elementos que componen este proceso.

3.1.3.1 *Desarrollo y validación de algoritmos:* Se construyen y validan los algoritmos para acceder a las fuentes y capturar los datos. Estos se basan en la información general de las fuentes y la especificación de los datos.

- Acceso o Conexión a la fuente: Se construye la primera parte de los algoritmos la cual evaluará y accederá en los casos posibles a cada fuente de interés tomando en cuenta información general de las

fuentes como restricciones de acceso y ubicación, tipo de fuente, etc. (Ver 3.1.2.1).

- Captura de registros: Parte algorítmica que accede a la especificación de los datos realizada en *3.1.2.3*, construyendo con esta información métodos de captura de los registros de interés.
- Envío de registros: Componente final que lleva los registros de la ubicación inicial hacia la ubicación definida para su almacenamiento, en muchos casos la ubicación de la base temporal (ver 3.2.1).
- Verificación y corrección de algoritmos: Se prueba que los algoritmos si estén accediendo y capturando los datos adecuadamente. En caso de encontrar fallas se debe realizar su reparación y de nuevo se debe iniciar el proceso de pruebas.

3.1.3.2 Ejecución de Algoritmos: Una vez definidos los algoritmos de acceso y se tenga construida la base temporal y la bodega de datos (Ver 3.2.1 y 3.2.3) se puede proceder con inicio de la recopilación.

3.1.4 Actualización periódica

Si procesos de análisis de datos se desean realizar de manera periódica es adecuado definir un proceso adicional que permita la identificación de cambios en las fuentes y a su vez permita mantener los datos actualizados en la medida de lo posible.

3.1.4.1 Evaluar restricciones de preferencia y acceso: Se evalúa la información relacionada con preferencias y restricciones de acceso para saber con qué periodicidad y en momento en específico acceder a una fuente.

3.1.4.2 Especificación de algoritmos de actualización: Se definen algoritmos basados en disparadores, cambio de estado o identificadores de inserción que permitan obtener tanto los datos nuevos como los modificados en una fuente.

3.1.4.3 Especificación de algoritmos de monitoreo: Se construyen algoritmos que informen de cambios en la estructura de una fuente que puedan afectar el proceso de actualización.

Un aporte adicional que se realiza a partir de esta investigación es la propuesta de definir el conocimiento que va permitir la captura y actualización adecuada de los registros de interés, mediante el uso plantillas diseñadas en el lenguaje etiquetado extensible (XML), permitiendo que esta información pueda ser usada de forma automática y a su vez logrando con esto poder actualizar, perfeccionar y corregir

gran parte de este conocimiento sin perjudicar los procesos de actualización periódica. A continuación se presenta una plantilla de conocimiento genérica para la actualización de registros sobre una fuente.

```
<?xml version=" 1.0 " encoding=" UTF-8 " standalone= "yes" ?>
<fuente>
        <ID>1</ID>
        <nombre>Ventas</nombre>
        <tipo>
                <general>estructurada</general>
                <aplicacion>Oracle</aplicación>
        </tipo>
        <acceso>
                <ubicacion>
                        <general></general>
                        <direccionip>10.1.12.241</direccionip>
                        <puerto>8086</puerto>
                </ubicacion>
                <autenticacion>
                        <tipo>1</tipo>
                        <usuario>ventasCID</usuario>
                        <contrasena>ventas2009</contrasena>
                </autenticacion>
        </acceso>
        <concepto>
                <nombre>producto</nombre>
                <atributo>
                        <nombre>id_producto</nombre>
                        <tipo>numerico</tipo>
                        <formato></formato>
                        <estado>1</estado>
                </atributo>
                <atributo>
                        <nombre>nombre_producto</nombre>
                        <tipo>String</tipo>
                        <formato></formato>
                        <estado>1</estado>
                </atributo>
        </concepto>
</fuente>
```

3.2 Estructuración de la Integración de Datos

En la mayoría de bases de datos existe mucha información incorrecta con respecto al dominio de la realidad que se desea representar. Estos problemas se acentúan cuando realizamos la integración de distintas fuentes, dado que puede ocurrir que varias fuentes diferentes pueden afirmar cosas distintas sobre el mismo concepto.

La integración también produce una disparidad de formatos, nombre, rangos, etc., que podría no existir, o en menor medida, en las fuentes originales. Esto dificulta en gran medida los procesos de análisis y extracción de conocimiento.

La estructuración que se presenta en este apartado tiene como objetivo reducir los problemas de inconsistencias y datos erróneos que se presentan en el momento de realizar una integración de varias fuentes de datos.

Para esta etapa de Integración de datos se definieron 3 procesos generales con las cuales se busca abarcar el desarrollo de la misma, logrando integrar datos necesarios, reduciendo en la medida de lo posible las inconsistencias presentadas.

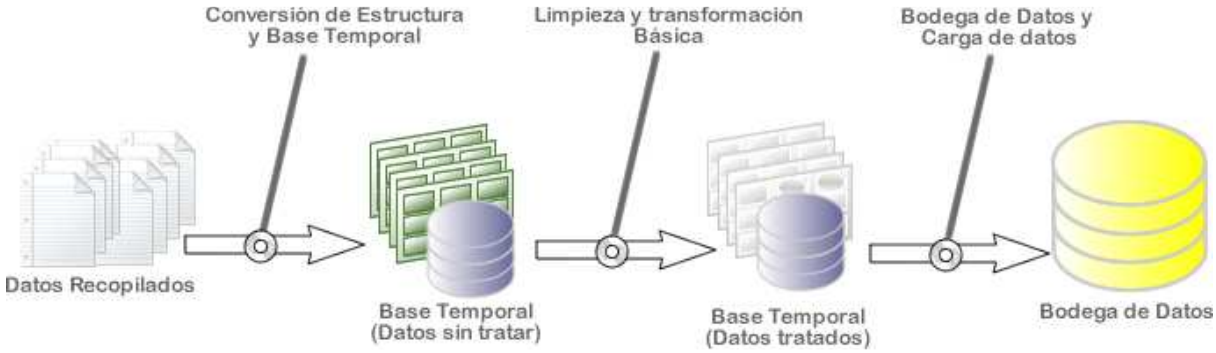


Figura 3.2 Procesos de la Etapa de Integración de datos.

Los procesos definidos son:

- Conversión de Estructura y Creación de Base Temporal
- Limpieza y Transformación básica.
- Bodega de datos y Carga de datos.

Seguidamente se presenta una descripción amplia de cada uno de estos procesos junto con los respectivos pasos propuestos para llevarlos a cabo satisfactoriamente.

3.2.1 Conversión de Estructura y Creación de Base Temporal.

3.2.1.1 *Modelo de diseño lógico y físico de la Base Temporal*: La base de datos temporal es un repositorio en el cual se almacenan los registros mientras no puedan ser llevados a la Bodega de datos. En está también se realizan la transformaciones necesarias para poder llevar los registros posteriormente a la bodega de datos. Seguidamente se presenta una descripción de cómo desarrollar este modelo.

- Modelo de diseño lógico: Debido a que el objetivo de esta base de datos es brindar a los registros capturados un almacenamiento temporal, el modelo de datos debe construirse semejante a los modelos de las fuentes recopiladas y no pensando en un modelo orientado a los procesos de análisis y toma de decisiones.

  Sin embargo este modelo al representar una fuente de datos estructurada, debe definir nuevas entidades que le permitan almacenar de forma adecuada los registros obtenidos a partir de las fuentes semi-estructuradas y no estructuradas.

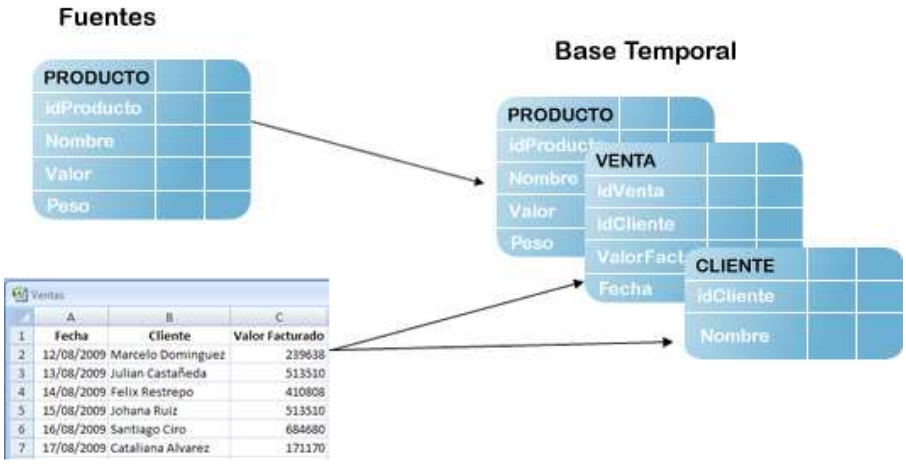


Figura 3.3 EJ: Mapeo de Conceptos de fuentes a la base temporal.

La base temporal no debe almacenar los registros ya integrados, ya que existen casos donde los registros que conforman un registro integrado no se generen simultáneamente.

El resto del proceso del diseño del lógico de la Base Temporal se rige por los fundamentos tradicionales del diseño de Bases de datos, tales

como las dependencias funcionales, la normalización y los requerimientos de seguridad.

- Modelo de diseño físico: El objetivo de este modelo es producir una descripción de la implementación de la base de datos, la cual incluye estructuras de almacenamiento, métodos de acceso, optimización y mecanismos de seguridad. Para desarrollar este modelo se propone seguir lo presentado en [Connolly et al, 1996] donde el diseño físico se divide en cuatro fases, cada una de ellas compuesta por una serie de pasos vistos a continuación.

    - Traducir el esquema lógico global para el Sistema Gestor de Base de Datos (SGBD) específico: La primera fase del diseño lógico consiste en traducir el esquema lógico global en un esquema que se pueda implementar en el SGBD escogido. Para ello, es necesario conocer toda la funcionalidad que éste ofrece. La actividades de esta fase son:

        1. Diseñar las relaciones base para el SGBD específico.
        2. Diseñar las reglas de negocio para el SGBD específico.

    - Diseñar la representación física: Uno de los objetivos principales del diseño físico es almacenar los datos de modo eficiente. Para medir la eficiencia hay varios factores que se deben tener en cuenta tale como la productividad de transacciones (número de transacciones que se quiere procesar en un intervalo de tiempo); Tiempo de respuesta (tiempo que tarda en ejecutarse una transacción; desde el punto de vista del usuario, este tiempo debería ser el mínimo posible); y Espacio en disco (cantidad de espacio en disco que hace falta para los registros de la base de datos). La actividades de esta fase son:

        1. Analizar las transacciones.
        2. Escoger las organizaciones de registros.
        3. Escoger los índices secundarios.
        4. Considerar la introducción de redundancias controladas.
        5. Estimar la necesidad de espacio en disco.

- Diseñar los mecanismos de seguridad: Los datos constituyen un recurso esencial para cualquier organización, por lo tanto su seguridad es de vital importancia. Durante el diseño lógico se especifican los requerimientos en cuanto a seguridad que en esta fase se deben implementar. Para llevar a cabo esta implementación, el diseñador debe conocer las posibilidades que ofrece el SGBD que se vaya a utilizar. La fase tiene las siguiente actividad:

    1. Diseñar las reglas de acceso.

- Monitorizar y afinar el sistema: Una vez implementado el esquema físico de la base de datos, se debe poner en marcha para observar sus prestaciones. Si éstas no son las deseadas, el esquema deberá cambiar para intentar satisfacerlas. Una vez afinado el esquema, no permanecerá estático, ya que tendrá que ir cambiando conforme lo requieran los nuevos requisitos de los usuarios. Los SGBD proporcionan herramientas para monitorizar el sistema mientras está en funcionamiento.

3.2.1.2 *Adecuación e inserción de datos recopilados:* Con la base temporal ya construida es momento de llevar los registros recopilados a ella, sin embargo primero se debe construir un mapeo que permita identificar como están representados los registros de la fuentes dentro de la base temporal y a su vez que tipos de transformaciones deben sufrir para poder adaptarse al modelo de datos de la misma. Los sub-pasos para realizar es numeral se presentan a continuación.

- Adecuación de datos: La base temporal posee un conjunto de restricciones en su modelo de datos (Formato, tipo de dato, etc.). Por tanto los registros que serán cargados en ella deben cumplir con estas restricciones. Debido a esto deben definirse las operaciones que tienen que sufrir los registros recopilados antes de ser cargados en la Base temporal.

  Ejemplo:

  El atributo Peso (lbs) del Concepto Producto en la fuente con ID 1 debe ser transformado a la medida estándar en Kilogramos (kgs) mediante el siguiente procedimiento:

  nuevo_valor=valor_registro* 0,45359237

- Mapeo de Carga: Al existir la diversidad de fuentes, debe existir un mecanismo o conocimiento que permita saber los registros de una fuente en específico donde deben ser alojado en la base temporal.

  Ejemplo:

  El atributo Peso del Concepto Producto en la fuente con ID 1 debe ser cargado en la Tabla BT_Producto en el atributo Peso.

- Migración de datos: Una vez se adecuen los registros y exista el conocimiento que permita alojarlos en la base temporal, se procede con la carga de estos registros en sus respectivas estructuras en la base temporal. Para este sub-paso se recomienda la construcción de algoritmos que permitan la automatización del mismo.

Para este numeral también se propone el uso de plantillas diseñadas en el lenguaje etiquetado extensible (XML), con la finalidad que sirvan como conocimiento definido para la automatización de este proceso de carga de registros en la base temporal. Seguidamente se presenta un ejemplo que ilustra lo propuesto.

```
<?xml version=" 1.0 " encoding=" UTF-8 " standalone= "yes" ?>
<transformacion>
    <fuente>
        <id_fuente>1</nombre>
        <concepto>
            <nombre>Producto</nombre>
            <atributo>
                <nombre>Peso</general>
                <operacion>
                    <tipo> multiplicacion</tipo>
                    <valor1>THIS</valor1>
                    <valor2> 0,45359237</valor2>
                </operacion>
            </atributo>
                .
                .
        </concepto>
            .
            .
    </fuente>
```

```
</transformacion>
<mapeo>
    <fuente>
        <id_fuente>1</nombre>
        <concepto>
            <nombre>Producto</nombre>
            <atributo>
                <nombre>Peso</general>
                <tablaBT>Producto</tablaBT>
                <atributoBT>Peso</atributoBT>
            </atributo>
                .
                .
        </concepto>
            .
            .
    </fuente>
</mapeo>
```

Esta plantilla xml puede enriquecerse mediante un pseudocódigo que permita dar más libertad a las operaciones, siempre y cuando se construya dentro del sistemas final un interpretador de este pseudocódigo.

3.2.2 Limpieza y transformación básica.

Aunque estas dos tareas son generalmente complejas y se realizan posteriormente en otras etapas del proceso de KDD, parte de la limpieza y la transformación necesaria para organizar los datos en la bodega de datos se realiza en la base temporal con el fin de evitar datos redundantes, inconsistentes, estandarizar medidas, formatos, fechas, tratar valores nulos, etc.

3.2.2.1 *Tratamiento simple de valores faltantes*: Los valores faltantes pueden ser reemplazados por varias razones. En primer lugar, el método de minería de datos que se utilice puede no trabajar bien con valores nulos o faltantes. En segundo lugar se puede querer agregar los datos (especialmente los numéricos) para realizar otras vistas minables y que los valores faltantes no permitan trabajar correctamente (totales, varianzas, etc.). En tercer lugar, si el método es capaz de permitir campos faltantes es posible que ignore todo el registro (produciendo un sesgo) o es posible que tenga un método de sustitución de campos faltantes que no sea adecuado debido a que no conoce el contexto asociado al atributo faltante.

Tanto para la detección, como para el tratamiento posterior es importante saber la causa de los valores faltantes, ya que de esto dependerá el uso de un tratamiento adecuado o no. Las posibles acciones sobre datos faltantes son: ignorar, eliminar, filtrar la fila, reemplazar el valor, segmentar o esperar a que estén disponibles.

Ejemplo:

Si el registro del atributo *Estatura* del Concepto *Deportista* en la fuente con ID 6 es nulo, aplicar el siguiente procedimiento:

1) Poner en Cola hasta terminar ingreso de registros asociados al concepto
2) El registro de este atributo será igual al promedio de los registros del atributo asociado en la base temporal.

3.2.2.2 *Tratamiento simple de valores erróneos*: Del mismo modo que para los campos faltantes, para los campos erróneos o inválidos se ha de distinguir entre la detección y el tratamiento de los mismos. La detección de campos erróneos se puede realizar de maneras muy diversas, dependiendo del formato y origen del cambio. En el caso de datos nominales, la detección dependerá fundamentalmente de conocer el formato o los posibles valores del campo. En el caso de la detección de valores erróneos en datos numéricos, ésta suele empezar por buscar valores anómalos, atípicos o extremos, también llamados datos aislados, exteriores o periféricos.

Las acciones a tomar para el tratamiento de datos anómalos o erróneos pueden ser: Ignorar, filtrar la columna, filtrar la fila, reemplazar el valor y discretizar.

Ejemplo:

Si el registro del atributo Edad del Concepto Deportista en la fuente con ID 6 es menor 5, aplicar el siguiente procedimiento:

1) Registro igual a vacio.

3.2.2.3 *Estandarizar datos*: Para la misma entidad del mundo real los valores de un atributo de diversas fuentes pueden diferir. Este problema puede ser debido a las diferencias en la representación, escalamiento o codificación. Por ejemplo el atributo *Peso* puede ser almacenado en unidades métricas en un sistema y en unidades imperiales (inglesas) en otro. Este paso tiene como objetivo estandarizar medidas, formatos, fechas y todo tipo de datos que puedan significar lo mismo pero tener diferentes representaciones.

Ejemplo:

Si el registro del atributo estado_civil del Concepto Deportista en la fuente con ID 6 es 'soltero', aplicar el siguiente procedimiento:

1) Registro igual a 1.

Con la perspectiva de automatizar estos procesos para los actuales y futuros registros recopilados, se propone definir como transformaciones adicionales dentro de documento XML propuesto en el numeral 3.2.1, el conocimiento generado en este numeral. Se debe aclarar que no todas las operaciones desarrolladas en este numeral podrán ser llevadas a la plantilla XML, esto debido a limitaciones del lenguaje.

3.2.3 Bodega de datos y carga de datos

A continuación se presentan los pasos definidos para llevar a cabo en este numeral.

3.2.3.1 *Desarrollo de Bodega de datos:* El concepto de bodegas de datos nace hace más de una década [Inmon 1992] ligado al concepto de EIS (Executive Information System) o Sistema de Información Ejecutivo de una Organización. La definición original de bodega de datos según este autor es: "Colección de datos, orientada a un dominio, integrada, no volátil y variante en el tiempo para ayudar en las decisiones dirección". En la realidad las bodegas de datos pueden utilizarse de muy diferentes maneras, y pueden agilizar muchos procesos diferentes de análisis tales como: herramientas de consultas e informes, herramientas EIS, herramientas OLAP y herramientas de minería de datos.

Las bodegas de datos no son imprescindibles para hacer extracción de conocimiento a partir de datos. En realidad, se puede hacer minería de datos incluso sobre un simple archivo de texto. Sin embargo, las ventajas de organizar una bodega de datos se resaltan increíblemente a mediano y largo plazo. Esto es especialmente patente cuando existen grandes volúmenes de datos, o éstos aumentan con el tiempo, provienen de fuentes heterogéneas o se van a querer combinar de maneras arbitrarias y no predefinidas. Tampoco es cierto que una bodega de datos sólo tenga sentido si se tiene una base de datos transaccional inicial. Incluso si todos los datos originalmente no provienen de bases de datos puede ser conveniente la creación de una bodega de datos.

Con el objetivo de tener procesos eficientes, los sistemas de bodegas de datos pueden implementarse utilizando dos tipos de esquemas físicos.

- ROLAP (Relational OLAP): Físicamente el almacén se construye sobre una base de datos relacional.
- MOLAP (Multidimensional OLAP): Físicamente el almacén de datos se construye sobre estructuras basada en matrices multidimensionales.

Para el desarrollo de la bodega de datos se propone utilizar la metodología encontrada en [Kimball 1998], la cual presenta las siguientes fases.

- Planeación y Administración del Proyecto: Consiste en identificar el área de demanda del proyecto, su viabilidad y alcance. Para esta fase se realiza la estructuración del proyecto, la preparación de la organización, se construye un enfoque basado en requerimientos, se justifica el proyecto y planea el desarrollo del mismo.

- Análisis de Requerimientos: Fase que busca identificar lo que requiere la organización y puede obtenerse a partir de los datos que se encuentran disponibles.

- Modelamiento dimensional: Fase basada en técnicas de diseño lógico que busca presentar los datos de una forma intuitiva y que proporcione acceso de alto desempeño. Cada modelo dimensional se compone de una tabla con múltiples llaves foráneas, llamada tabla de hecho (fact table), y un conjunto de tablas más pequeñas, llamadas tablas de dimensión. Existen dos modelos dimensionales que predominan en las soluciones de bodega de datos: El modelo estrella y el modelo copo de nieve.

- Diseñó técnico de la Arquitectura: Fase de definición de un modelo físico con el que se pretende un desarrollo más confiable y eficiente. Con la definición de la arquitectura se mejora la comunicación entre las diferentes áreas del proyecto, el planeamiento del proyecto, la flexibilidad y el manteamiento del mismo.

- Selección e instalación de productos: Envuelve los procesos de evaluación, selección e instalación de herramientas que facilitan el trabajo con los datos. En esta fase se realizan análisis rigurosos de cada herramienta, tomando en cuenta ventajas, desventajas y alcance de las mismas.

- Características de aplicaciones para usuarios finales: Fase que tiene como norte proporcionar interfaces al usuario donde se presenten reportes y análisis multidimensionales, que serán la base en la toma de decisiones.

- Mantenimiento y crecimiento de una bodega de datos: Fase que pretende mantener en un estado óptimo las bodegas de datos. Está conformada por el análisis y corrección de resultados inadecuados, asesoría en procesos de adaptación en la organización y revisiones de estado-en-punto a partir de operadores tales como: Infraestructura técnica, desempeño general y mantenimiento de datos y metadatos.

3.2.3.2 *Unificación y Carga de Datos:* Con la Bodega de Datos ya construida es momento de llevar los registros alojados en la base temporal a ella, sin embargo primero se debe construir un mapeo que permita identificar como están representados los registros de la base temporal dentro de la Bodega de datos y a su vez que tipos de integraciones y transformaciones deben sufrir para poder adaptarse al modelo de datos de la misma. Los sub-pasos para realizar es numeral se presentan a continuación.

- Adecuación de datos: La bodega de datos posee generalmente un modelo de datos diferente al disponible en la Base temporal por tanto deben definirse las operaciones respectivas que deben sufrir los registros alojados en la Base temporal antes de ser cargados en la Bodega de Datos. La operación que más se presentan para este numeral es la integración.

- Mapeo de Carga: Al existir un cambio de modelo de datos entre la Base temporal y la bodega de datos, se debe proveer un mecanismo o conocimiento que permita saber los registros de la base temporal donde deben ser alojado en la Bodega de Datos.

  Ejemplo:

  > El atributo *Peso* del Concepto BT_Producto en la fuente con ID 1 debe ser cargado en la Tabla BD_Producto en el atributo *Peso*.

- Carga de registros: Una vez se adecuen los registros y exista el conocimiento que permita alojarlos en la Bodega de datos, se procede con la carga de estos registros en sus respectivas estructuras.

Para este sub-paso se recomienda la construcción de algoritmos que permitan la automatización del mismo. La carga de datos debe ser planificada por fases para evita afectar en ocasiones procesos de análisis realizados sobre la bodega de datos. Al ser tantos datos se recomienda que la primera carga se realice de manera automática y monitoreada y las cargas posteriores sean automáticas y revisadas a través de reportes de efectividad.

De la manera en que se viene haciendo se propone también que el conocimiento generado acerca de los datos en este numeral, se especifique a través de una plantilla XML con el objetivo facilitar la automatización de la carga e integración de los registros.

CAPITULO 4

MODELO MULTI-AGENTE PARA EL DESARROLLO DE LAS ETAPAS DE INTEGRACIÓN Y RECOPILACIÓN DATOS EN EL PROCESO KDD

Uno de los objetivos de este trabajo de investigación consiste en definir un modelo Multi-Agente para el desarrollo de las etapas de integración y recopilación datos en el proceso KDD. En ese orden de ideas, se presenta en este capítulo las fases de construcción del modelo Multi-Agente de acuerdo a la Metodología MAS-CommonKADS definida por Carlos Iglesias en [Iglesias, 1998]. El modelo de ciclo de vida para el desarrollo de Sistemas Multi-Agente con MAS-CommonKADS se compone principalmente de las siguientes fases:

- Conceptualización: Esta fase es necesaria para obtener una primera descripción del problema y la determinación de los casos de uso que pueden ayudar a entender los requisitos informales y a probar el sistema.

- Análisis: Determinación de los requisitos del sistema partiendo del enunciado del problema. Durante esta fase se desarrollan los siguientes modelos: organización, tareas, agente, comunicación, coordinación y experiencia.

- Diseño: Determinación de cómo los requisitos de la fase de análisis pueden ser logrados mediante el desarrollo del modelo de diseño. Se determinan las arquitecturas tanto de la red Multi-Agente como de cada agente.

En este capítulo se desarrollan las fases de conceptualización, análisis y diseño de la metodología MAS-CommonKADS. En los apartados posteriores se presenta la Implementación y análisis de resultados.

4.1 Conceptualización

En la fase de conceptualización, se delimita el problema y se define el alcance del proyecto. En esta fase se identifican las entidades involucradas en el desarrollo de

las etapas de Recopilación e Integración, así como sus objetivos, tareas e interacciones; y de manera general, el funcionamiento del sistema. Esto conllevará a la elaboración de un primer acercamiento basado en diagramas de casos de uso, ya que estos ayudan a comprender los procesos que se llevan a cabo y definir el alcance funcional de esta investigación

4.1.1 Casos de Uso

Este modelo es utilizado en esta fase debido a que describe las diferentes relaciones entre las entidades que componen el sistema a un nivel de abstracción bastante elevado. Como primer paso se procede a identificar los actores (entidades) que componen el sistema y desempeñan funciones al interior y exterior de este. Para cada actor se trata de identificar su o sus casos de uso, que representan las funciones que desempeñan.

De este modo el conjunto de actividades que se seguirán en el desarrollo de la fase de conceptualización son: identificación de los actores, descripción de los actores, relaciones e interacciones entre actores (identificación de los casos de uso y descripción de los casos de uso) [Iglesias, 1998]

4.1.1.1 *Identificación y descripción de los Actores:* Los actores considerados en este modelo para el desarrollo de las etapas de Recopilación e Integración de datos son:

Recolector: Realiza las tareas que intervienen en el acceso y extracción de datos en una fuente de interés. Una tarea adicional de este actor es comunicar los datos extraídos para su posterior transformación e integración y a su vez debe también informar de cambios y eventualidades ocurridas en las fuentes que él explora.

Almacenista: Recibe los datos recopilados y los ubica en una base temporal donde se conserva una estructura muy similar a la de la cual fueron extraídos. También realiza las primeras transformaciones sobre los datos recopilados y posteriormente cuando sea pertinente los envía a la bodega de datos para que sufran las transformaciones finales y sean almacenados con la estructura del esquema unificado.

Integrador: Se encarga de realizar las transformaciones que exige el sistema unificado sobre los datos localizados en la base temporal y de verificar si los datos transformados pueden ser almacenados en la bodega de datos, o deben

ser descartados. Genera también reportes con las posibles eventualidades en las que el sistema no esté preparado para responder adecuadamente (registros no trasladables al esquema unificado, inconsistencias generales, etc.).

Coordinador: La principal función de este actor es dirigir y coordinar el proceso de recopilación, administrando tanto el conocimiento de todas las fuentes como la asignación de los diferentes recursos a los demás actores. Se encarga también de comunicar y monitorear los cambios y actualizaciones que se realizan en todo el proceso.

Analista: Se encarga de tomar acciones sobre todo el proceso ya sea debido a problemas que se presenten en las etapas, o a actualizaciones que puedan mejorar los resultados arrojados. También analiza los informes de resultados de los procesos en general.

4.1.1.2 *Relaciones e Interacciones entre Actores*: Luego de identificar los actores se procede a reconocer las relaciones o interacciones de cada actor usando la notación propuesta por [Jacobson et al, 2000]. En estas figuras (Ver Figuras 4.1 a la 4.5) se pueden observar los casos de uso que fueron identificados para cada actor en el sistema.

Cada caso de uso presentado en esta notación va acompañado de un tabla que es mucho mas diciente (Ver Tablas 4.1 a la 4.15) pues contiene información que dice que actor interviene en que caso de uso, que precondiciones y poscondiciones deben darse, cuales son los pasos que se siguen para realizar la funcionalidad representada por el caso de uso (incluyendo bifurcaciones) y mucha más información que complementa al anterior modelo.

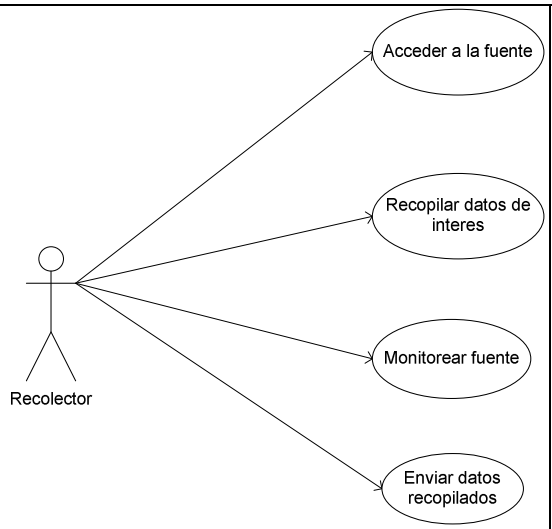


Figura 4.1 Casos de Uso Actor Recolector

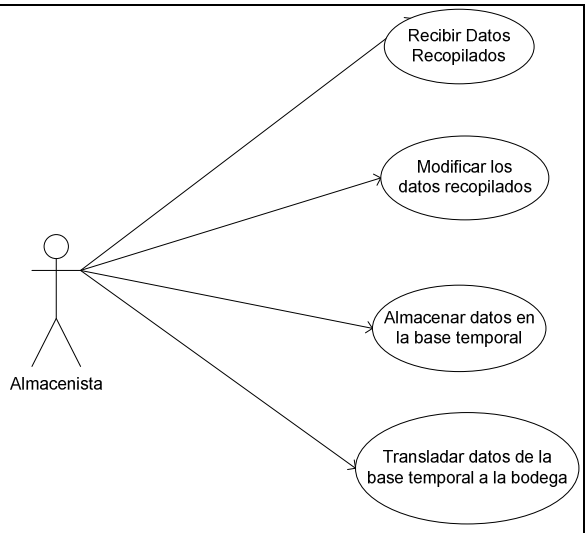


Figura 4.2 Casos de Uso Actor Almacenista

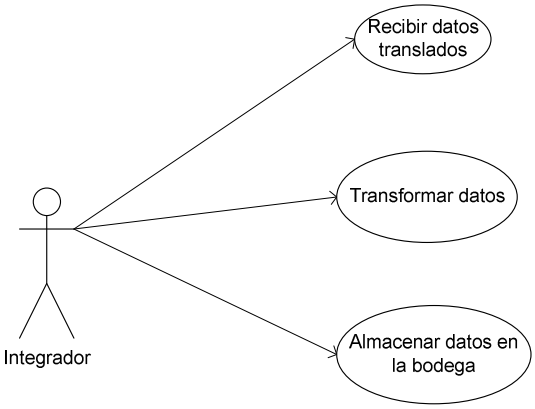


Figura 4.3 Casos de Uso Actor Integrador

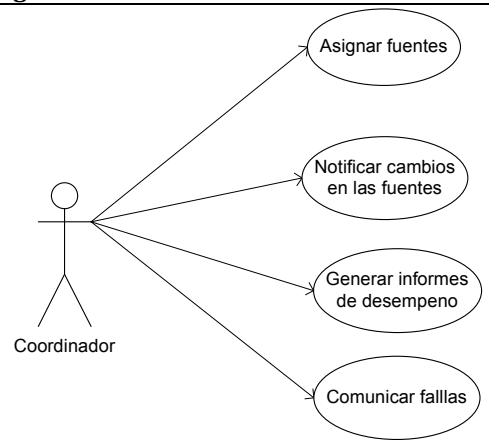


Figura 4.4 Casos de Uso Actor Coordinador

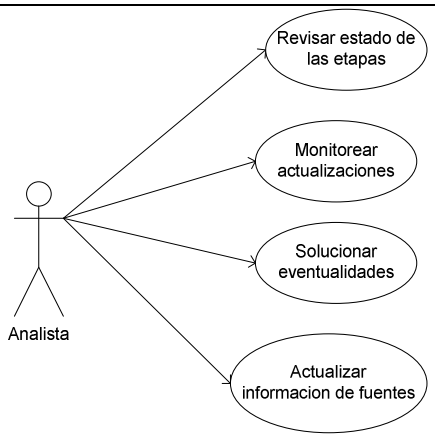


Figura 4.5 Casos de Uso Actor Analista

Caso de Uso	Acceder a la fuente		
Versión	01	Fecha:	15 -08 – 2009
Autores	Daniel Betancur Calderón		
Actores involucrados	Recolector		
Objetivos Asociados	Lograr establecer una conexión con la fuente en la cual se encuentran los datos de interés.		
Descripción	Operación que permite lograr acceder a una fuente para la posterior extracción de datos.		
Precondiciones	Parámetros básicos de acceso definidos, Medios de acceso disponible.		
Flujo de Eventos			
Secuencia normal o Flujo Básico			
No	Recolector	Sistema	
1	Solicita información de acceso y autentificación	Envía información pertinente	
2	Inicia un enlace con la fuente deseada		
3	Se autentica y accede a la fuente		
Secuencia alternativa o flujos alternativos			
No	Paso		
2	Si no se puede iniciar el enlace reintenta de nuevo en cierto periodo de tiempo. Si en 10 intentos no logra el enlace, comunica este hecho como una falla.		
3	Si no se logra la autenticación se genere reporte falla		
Poscondiciones	Se puede acceder a los datos de la fuente		

Tabla 4.1. "Acceder a la Fuente"

Caso de Uso	Recopilar datos de Interés		
Versión	01	Fecha:	15 -08 - 2009
Autores	Daniel Betancur Calderón		
Actores involucrados	Recolector		
Objetivos Asociados	Recopilar todos los datos de interés que se encuentren en la fuente.		
Descripción	Mecanismo que permite capturar todos los datos de interés sobre una fuente en particular		
Precondiciones	Enlace de acceso establecido, Información sobre la estructura de la fuente disponible		
Flujo de Eventos			
Secuencia normal o Flujo Básico			
No	Recolector	Sistema	
1	Solicita información estructural de la fuente	Envía información pedida	
2	Captura datos basándose en el información entregada		
Secuencia alternativa o flujos alternativos			
No	Paso		
2	En caso de que algunos datos no se puedan recuperar se debe hacer una notificación para posteriormente evaluarla.		
Poscondiciones	Datos de la fuente capturados		

Tabla 4.2. "Recopilar datos de interés"

Caso de Uso	Monitorear Fuente		
Versión	01	Fecha:	15 -08 - 2009
Autores	Daniel Betancur Calderón		
Actores involucrados	Recolector		
Objetivos Asociados	Identificar cuando se produce cambios en una fuente de interés.		
Descripción	Mecanismo mediante el cual se identifican cambios ocurridos en un momento dado sobre una fuente.		
Precondiciones	Información estructural de la fuente disponible		
Flujo de Eventos			
Secuencia normal o Flujo Básico			
No	Recolector	Sistema	
1	Acceder a la fuente		
2	Identificar cambios ocurridos		
Secuencia alternativa o flujos alternativos			
No	Paso		
Poscondiciones	Cambios ocurridos identificados		

Tabla 4.3. "Monitorear Fuente"

Caso de Uso	Enviar y recibir datos recopilados		
Versión	01	Fecha:	15 -08 - 2009
Autores	Daniel Betancur Calderón		
Actores involucrados	Recolector, Almacenista		
Objetivos Asociados	Trasladar los datos capturados en una fuente a la base temporal		
Descripción	Operación donde los datos son enviados de una fuente en específico y recibidos para su almacenamiento.		
Precondiciones	Datos de la fuente capturados		
Flujo de Eventos			
Secuencia normal o Flujo Básico			
No	Recolector	Almacenista	
1	Realiza petición de envío de datos	Recibe petición y confirma disponibilidad.	
2	Envía los datos recopilados.	Recibe los datos y confirma transferencia exitosa.	
Secuencia alternativa o flujos alternativos			
No	Paso		
1	En casos donde el almacenista no se encuentre disponible, se construye una cola de peticiones, para dar respuesta a ellas posteriormente.		
Poscondiciones	Datos recopilados recibidos		

Tabla 4.4 "Enviar y recibir datos recopilados"

Caso de Uso	Modificar los datos recibidos		
Versión	01	Fecha:	15 -08 - 2009
Autores	Daniel Betancur Calderón		
Actores involucrados	Almacenista		
Objetivos Asociados	Modificar los datos recibidos con la finalidad de adaptarlos a la estructura de la base temporal.		
Descripción	Mecanismo que aplica modificaciones simples en los datos.		
Precondiciones	Datos Recopilados disponibles		
Flujo de Eventos			
Secuencia normal o Flujo Básico			
No	Almacenista	Sistema	
1	Solicita información sobre datos recopilados	Envía información solicitada	
2	Aplica modificaciones correspondientes a cada tipo de datos		
Secuencia alternativa o flujos alternativos			
No	Paso		
Poscondiciones	Datos preprocesados		

Tabla 4.5. "Modificar los datos recibidos"

Caso de Uso	Almacenar datos en la bodega temporal		
Versión	01	Fecha:	15 -08 - 2009
Autores	Daniel Betancur Calderón		
Actores involucrados	Almacenista		
Objetivos Asociados	Lograr agregar todos los datos recopilados a la base temporal		
Descripción	Operación mediante la cual se agregan los datos recopilados a la base temporal		
Precondiciones	Datos preprocesados		
Flujo de Eventos			
Secuencia normal o Flujo Básico			
No	Almacenista	Sistema	
1	Mapea estructura de datos recopilados a la base temporal		
2	Realiza la inserción de los datos recopilados en la base temporal		
Secuencia alternativa o flujos alternativos			
No	Paso		
1	En caso de que algunos datos no se logre hacer el mapeo, se separan esos datos y se comunica la falla presentada		
Poscondiciones	Datos almacenados en la Base Temporal		

Tabla 4.6. "Almacenar datos en la bodega temporal"

Caso de Uso	Enviar y recibir datos de la base Temporal		
Versión	01	Fecha:	15 -08 - 2009
Autores	Daniel Betancur Calderón		
Actores involucrados	Almacenista, Integrador		
Objetivos Asociados	Trasladar los datos almacenados en la base temporal para su posterior almacenamiento en la bodega de datos.		
Descripción	Mecanismo que permite el envió y recepción de los datos almacenados en la base temporal.		
Precondiciones	Datos almacenados en la base temporal		
Flujo de Eventos			
Secuencia normal o Flujo Básico			
No	Almacenista	Integrador	
1	Realiza petición de envío de datos	Recibe petición y confirma disponibilidad.	
2	Envía los datos almacenados en la base temporal.	Recibe los datos y confirma transferencia exitosa.	
Secuencia alternativa o flujos alternativos			
No	Paso		
1	En casos donde el almacenista o la bodega no se encuentren disponibles, se construye una cola de peticiones, para dar respuesta a ellas posteriormente.		
Poscondiciones	Datos preprocesados recibidos		

Tabla 4.7. "Enviar y recibir datos de la base Temporal"

Caso de Uso	Transformar datos		
Versión	01	Fecha:	16 -08 - 2009
Autores	Daniel Betancur Calderón		
Actores involucrados	Integrador		
Objetivos Asociados	Aplicar operaciones sobre los datos que permitan almacenarlos en el esquema unificado definido.		
Descripción	Operación mediante la cual se transforman los datos con la finalidad de poder ser almacenados posteriormente en la bodega de datos.		
Precondiciones	Recibidos datos de la base temporal		
Flujo de Eventos			
Secuencia normal o Flujo Básico			
No	Integrador	Sistema	
1	Relaciona los datos disponibles con su respectiva estructura en la bodega temporal		
2	Aplica transformaciones requeridas.		
Secuencia alternativa o flujos alternativos			
No	Paso		
Poscondiciones	Datos transformados al esquema unificado		

Tabla 4.8. "Transformar datos"

Caso de Uso	Almacenar datos en la bodega		
Versión	01	Fecha:	16 -08 - 2009
Autores	Daniel Betancur Calderón		
Actores involucrados	Integrador		
Objetivos Asociados	Insertar los datos en la bodega para posteriores análisis.		
Descripción	Mecanismo mediante el cual todos los datos capturados de la base temporal son almacenados en la Bodega de datos.		
Precondiciones	Datos disponibles en el formato unificado predefinido.		
Flujo de Eventos			
Secuencia normal o Flujo Básico			
No	**Integrador**		**Sistema**
1	Revisa información relacionada con la estructura de la bodega de datos.		
2	Inserta datos en la bodega		
Secuencia alternativa o flujos alternativos			
No	**Paso**		
2	Se produce error al insertar algunos datos		
3	Aísla datos y comunica falla.		
Poscondiciones	Datos almacenados en la bodega		

Tabla 4.9. "Almacenar datos en la bodega"

Caso de Uso	Solicitar y Asignar Fuente		
Versión	01	Fecha:	16 -08 - 2009
Autores	Daniel Betancur Calderón		
Actores involucrados	Coordinador, Recolector		
Objetivos Asociados	Distribuir la diferentes fuentes de interés		
Descripción	Mecanismo para asignar las fuentes de interés de las cuales se pretende capturar datos.		
Precondiciones			
Flujo de Eventos			
Secuencia normal o Flujo Básico			
No	**Recolector**		**Coordinador**
1	Solicita asignación fuente		Analiza disponibilidad fuentes.
2			Responde petición con la información básica de la fuente asignada.
Secuencia alternativa o flujos alternativos			
No	**Paso**		
2	Responde petición con rechazo debido a la no disponibilidad de fuentes		
Poscondiciones	Fuente asignada		

Tabla 4.10. "Solicitar y Asignar Fuente"

Caso de Uso	Notificar y recibir reporte de cambios en las fuentes		
Versión	01	Fecha:	16 -08 - 2009
Autores	Daniel Betancur Calderón		
Actores involucrados	Coordinador, Analista		
Objetivos Asociados	Comunicar cambios que ocurren en la fuentes que puedan perjudicar el desarrollo del proceso de recopilación		
Descripción	Operación que permite al Coordinador dar a conocer cambios ocurridos en las fuentes de interés al analista con la finalidad de que este tomo acciones sobre estos hechos.		
Precondiciones	Cambio ocurrió en una Fuente de interés		
Flujo de Eventos			
Secuencia normal o Flujo Básico			
No	**Coordinador**		**Analista**
1	Comunica reporte de cambios encontrados		Recibe reporte
2			
Secuencia alternativa o flujos alternativos			
No	**Paso**		
Poscondiciones	Reporte de cambios notificado		

Tabla 4.11. "Notificar cambios en las fuentes"

Caso de Uso	Generar informes de desempeño		
Versión	01	Fecha:	16 -08 - 2009
Autores	Daniel Betancur Calderón		
Actores involucrados	Coordinador, Recolector, almacenista, integrador		
Objetivos Asociados	Generar un informe del estado en el que se encuentran los procesos de recopilación e integración.		
Descripción	Se pide un reporte de estado tanto a los Recolectores como al almacenista e integrador.		
Precondiciones	Activación del proceso		
Flujo de Eventos			
Secuencia normal o Flujo Básico			
No	**Coordinador**		**Recolector, Almacenista o integrador**
1	Solicita información de estado de actividades		Envía información de estado de actividades
2	Genera reporte de estado de desempeño		
3			
Secuencia alternativa o flujos alternativos			
No	**Paso**		
Poscondiciones	Informe de desempeño construido		

Tabla 4.12. "Recuperar Contenidos Educativos"

Caso de Uso	Comunicar y Solucionar eventualidades		
Versión	01	Fecha:	16 -08 – 2009
Autores	Daniel Betancur Calderón		
Actores involucrados	Coordinador, Analista		
Objetivos Asociados	Lograr identificar y solucionar eventualidades se presenten durante el desarrollo de las etapas de recopilación e integración		
Descripción	Mecanismo que permite comunicar eventualidades y buscar posibles soluciones a las mismas.		
Precondiciones	Se genera una eventualidad		
Flujo de Eventos			
Secuencia normal o Flujo Básico			
No	**Coordinador**	**Analista**	
1	Comunica eventualidad	Recibe comunicado	
2		Genera solución y la implementa	
Secuencia alternativa o flujos alternativos			
No	**Paso**		
Poscondiciones	Eventualidad comunicada y solucionada		

Tabla 4.13. "Comunicar y Solucionar eventualidades"

Caso de Uso	Monitorear actualizaciones		
Versión	01	Fecha:	16 -08 - 2009
Autores	Daniel Betancur Calderón		
Actores involucrados	Analista		
Objetivos Asociados	Identificar cuando se implementa bien y cuando no una actualización		
Descripción	Se verifica que la actualización sea realizada en las diferentes fuentes de información		
Precondiciones	Ocurre una Actualización		
Flujo de Eventos			
Secuencia normal o Flujo Básico			
No	**Analista**	**Sistema**	
1	Revisa todos lo puntos de información que debieron haber cambiado para la respectiva actualización		
2	Encuentra errores y los corrige		
Secuencia alternativa o flujos alternativos			
No	**Paso**		
2	No se encuentran errores		
Poscondiciones			

Tabla 4.14. "Monitorear actualizaciones"

Caso de Uso	Actualizar información de fuentes		
Versión	01	Fecha:	17 -08 - 2009
Autores	Daniel Betancur Calderón		
Actores involucrados	Analista		
Objetivos Asociados	Actualizar información de las fuentes con la finalidad de lograr mejorar los procesos e incrementar el alcance de los mismos.		
Descripción	Mecanismo que permite modificar información que se tiene sobre las fuentes		
Precondiciones	Cambios en la fuentes , Expansión en proceso de recopilación.		
Flujo de Eventos			
Secuencia normal o Flujo Básico			
No	**Analista**	**Sistema**	
1	Selecciona documentos a modificar		
2	Modifica información alojada en los documentos seleccionados.		
Secuencia alternativa o flujos alternativos			
No	**Paso**		
Poscondiciones			

Tabla 4.15. "Actualizar información de fuentes"

4.2 Análisis

MAS-CommonKADS propone los siguientes modelos para esta fase [Iglesias, 1998]:

Modelo de Agente (AM): Específica las características de un agente: sus capacidades de razonamiento, habilidades, servicios, sensores, efectores, grupos de agentes a los que pertenece y clase de agente. Un agente puede ser un agente humano, software, o cualquier entidad capaz de emplear un lenguaje de comunicación de agentes.

Modelo de Tareas (TM): Describe las tareas que los agentes pueden realizar: los objetivos de cada tarea, su descomposición, los ingredientes y los métodos de resolución de problemas para resolver cada objetivo.

Modelo de Organización (OM): Es una herramienta para analizar la organización humana en que el Sistema Multi-Agente va a ser introducido y para describir la organización de los agentes software y su relación con el entorno.

Modelo de la Experiencia (EM): Describe el conocimiento necesitado por los agentes para alcanzar sus objetivos. Sigue la descomposición de CommonKADS y reutiliza las bibliotecas de tareas genéricas.

Modelo de Comunicación (CM): Describe las interacciones entre un agente humano y un agente software. Se centra en la consideración de factores humanos para dicha interacción.

Modelo de Coordinación (CoM): Describe las interacciones entre agentes software.

Modelo de Diseño (DM): Mientras que los otros cinco modelos tratan del análisis del Sistema Multi-Agente, este modelo se utiliza para describir la arquitectura y el diseño del Sistema Multi-Agente como paso previo a su implementación.

La construcción de estos modelos constituye la fase de análisis, el resultado de esta fase es la especificación del sistema compuesto a través del desarrollo de los modelos descritos anteriormente. Los pasos de esta fase son

1 *Estudio de viabilidad*: El modelo de organización permite modelar la organización en que va a ser introducido el Sistema Multi-Agente, para estudiar las áreas posibles de aplicación de sistemas inteligentes, su viabilidad y su posible impacto.

2 *Delimitación*: Delimita el Sistema Multi-Agente de los sistemas externos. El desarrollo del modelo de organización habrá delimitado la interacción del Sistema Multi-Agente con el resto de sistemas de la organización. Los sistemas externos (predefinidos) deben ser encapsulados en agentes, modelados mediante el desarrollo de ejemplares del modelo de agente, y modelando sus interacciones mediante el desarrollo de modelos de coordinación. En el caso de que haya interacción con agentes humanos, esta interacción se describe en el modelo de comunicación.

3 *Descomposición*: El sistema se analiza mediante el desarrollo solapado de varios puntos de vista:

   a. Descomposición funcional: se descompone el sistema en funciones (tareas u objetivos) que deben ser realizados, mediante el desarrollo de un modelo de tareas global.
   b. Descomposición en ejecutores: el sistema se descompone en agentes que realizan las funciones anteriormente desarrolladas, mediante el desarrollo del modelo de agente.

4 *Descripción de las interfaces*: Tomando como punto de partida la fase de conceptuación y el modelo de agente, se describen las relaciones estáticas que determinan los canales de comunicación con otros agentes y con el exterior mediante el desarrollo del modelo de la organización de los agentes. Es necesario identificar los flujos de comunicación de la organización, que serán motivados, principalmente, por la necesidad de información y para evitar o resolver conflictos.

5 *Identificación y descripción de interacciones*: La identificación y descripción de las relaciones dinámicas entre los agentes se realiza de la siguiente manera:

   a. Las interacciones dinámicas con otros agentes software se describen en el modelo de coordinación.
   b. Las interacciones dinámicas con agentes humanos se describen en el modelo de comunicación.

6 *Descripción del razonamiento de los agentes*: Para cada agente, debemos modelar el conocimiento que necesita para llevar a cabo sus objetivos, mediante el desarrollo del modelo de la experiencia.

7 *Validación*: Cada vez que un agente es descompuesto en nuevos agentes, éstos deben ser consistentes lógicamente con la definición previa:

   a. Los subagentes son responsables de los objetivos del agente.

b. Los subagentes deben ser consistentes con el modelo de coordinación y mantener las mismas interacciones externas.
c. Validación cruzada con los otros modelos.
d. Los conflictos detectados en los escenarios deben tener determinado al menos un método de resolución de conflictos.

4.2.1 Modelo de Agente

Este modelo recoge las características genéricas de los agentes y sirve de puente entre el resto de modelos. Este modelo sirve para especificar las características de los agentes involucrados en la resolución de un problema y está pensado para recoger los requisitos que debe tener un agente para poder realizar las tareas (responsabilidades) asignadas.

A continuación se presentará detalladamente la información inicial, el/los objetivo(s) principal(es) y el/los servicio(s) de cada agente [1].

Información Inicial: Contiene una breve descripción del agente, de tal manera que se puedan identificar las características principales de este y su razón de ser. Esta información se compone por 4 ítems: Nombre, Tipo, Papel o Rol y Descripción.

Objetivo: Responsabilidad asignada o adoptada por un agente. La ejecución de esta responsabilidad puede realizarse mediante la ejecución de una determinada tarea o mediante un mecanismo de planificación.

Servicio: Tarea que un agente ofrece a otros agentes. Esta tarea puede ser de "alto nivel", como por ejemplo "Reserva un vuelo barato a Granada para el 4 de junio", que requiera una descomposición en varias tareas o un servicio "directamente ejecutable" que se realice mediante la ejecución de una función. En cualquier caso, la oferta de un servicio no implica que este se vaya a ejecutar cuando se demande. Será el agente el que decida si lo realiza o no, y bajo qué condiciones, tal y como se recoge en la entidad Restricciones.

Los agentes que fueron identificados en el sistema son los siguientes:

- Agente de software Recolector (Ver Agente 1)
- Agente de software Almacenista (Ver Agente 2)
- Agente de software Integrador (Ver Agente 3)
- Agente de software Coordinador (Ver Agente 4)
- Agente de humano Analista (Ver Agente 5)
- Agente de software Analista (Ver Agente 6)

Nombre: Agente Recolector.
Tipo: Agente de software de medios diversos (internet, BD locales, BD externes, etc.).
Papel o Rol: Recuperador de Datos
Descripción: Por medio de este agente será posible acceder a fuentes tanto internas como externas con la finalidad de obtener datos de interés para la posterior realización de la minería de datos.
OBJETIVO 1 **Nombre:** Acceder a una fuente de datos **Parámetros de Entrada:** Información de acceso, Validación de medio de acceso. **Parámetros de Salida:** Informe de acceso exitoso. **Condición-Activación:** Petición de recopilación de datos en una fuente. **Condición-Finalización:** Alcance de la cantidad máxima de intentos fallidos permitidos (Valor definido por el desarrollador). **Condición-Éxito:** Se obtiene acceso a la fuente de datos. **Descripción:** Con este objetivo, se pretende establecer un canal que permita posteriormente la captura de los datos de interés en la fuente.
OBJETIVO 2 **Nombre:** Recopilar datos de interés. **Parámetros de Entrada:** Estructura e información relevante de los datos de interés. **Parámetros de Salida:** Estadísticas de datos de capturados. **Condición-Activación:** Petición de recopilación de datos en una fuente. **Condición-Finalización:** Cancelación de la recopilación de estos datos. **Condición-Éxito:** Datos capturados. **Descripción:** Con este objetivo, se logra capturar los datos de interés sobre una fuente en particular
SERVICIO 1 **Nombre:** Construcción y envió de estado de tareas. **Parámetros-entrada:** Fecha de corte. **Parámetros-salida:** No hay parámetros de salida **Lenguaje par representar el conocimiento:** FIPA ACL **Ontología:** La desarrollada para el sistema (según el modulo del dominio y el modulo de información general).

Tabla 4.16. Agente 1 - "Agente Recolector"

Nombre: Agente Almacenista
Tipo: Agente de software estacionario
Papel o Rol: Receptor y almacenador de datos
Descripción: Este agente se encarga de recibir los datos recopilados, almacenarlos, modificarlos y posteriormente enviarlos para su almacenamiento en la bodega de datos.
OBJETIVO 1 **Nombre:** Recibir datos recopilados **Parámetros de Entrada:** Metadatos de la fuente **Parámetros de Salida:** Tiempo duración transferencia **Condición-Activación:** Solicitud de transferencia de datos **Condición-Finalización:** Perdida de Conexión **Condición-Éxito:** 100% de los datos recibidos **Descripción:** Este objetivo, busca que todos los datos recopilados sean recibidos a nivel local para ser luego almacenados.
OBJETIVO 2 **Nombre:** Modificar datos recibidos **Parámetros de Entrada:** Datos e información de modificaciones respectivas. **Parámetros de Salida:** Porcentaje de modificaciones aplicadas **Condición-Activación:** Recepción de nuevos datos **Condición-Finalización:** Cancelación del proceso de integración. **Condición-Éxito:** Todas las modificaciones necesarias son realizadas. **Descripción:** Consiste en realizar algunas modificaciones principalmente de formato que facilitaron la subsiguiente integración de los datos
OBJETIVO 3 **Nombre:** Almacenar datos modificados en la base temporal **Parámetros de Entrada:** Metadatos de la estructura del base temporal y de los datos a almacenar. **Parámetros de Salida:** Estadísticas de almacenamiento **Condición-Activación:** Datos modificados y sin almacenar **Condición-Finalización:** Cancelación del proceso de integración. **Condición-Éxito:** Todos los datos almacenados en la base temporal. **Descripción:** Este objetivo busca llevar los datos recopilados a una base temporal, en la cual estarán mientras no puedan ser enviados a la bodega de datos
SERVICIO 1 **Nombre:** Transferencia de datos alojados en la base temporal **Parámetros-entrada:** Tiempo aproximado de envió, Información básica de lo que se envía. **Parámetros-salida:** Estado de transferencia **Lenguaje par representar el conocimiento:** FIPA ACL **Ontología:** La desarrollada para el sistema (según el modulo del dominio y el modulo de información general).

Tabla 4.17. Agente 2 - "Agente Almacenista"

Nombre: Agente Integrador
Tipo: Agente de software estacionario.
Papel o Rol: Unificador de datos
Descripción: Este agente se encarga de alojar datos de fuentes diversas en un solo esquema unificado definido dentro de una bodega de datos.
OBJETIVO 1 **Nombre:** Recibir datos alojados en la base temporal **Parámetros de Entrada:** Metadatos de la estructura del base temporal y de la bodega de datos. **Parámetros de Salida:** Estadística de almacenamiento exitoso. **Condición-Activación:** Solicitud de transferencia de datos. **Condición-Finalización:** Pérdida de Conexión. **Condición.-Éxito:** 100% de los datos recibidos. **Descripción:** Este objetivo, busca recibir los datos almacenados en la base temporal para su posterior transformación.
OBJETIVO 2 **Nombre:** Almacenar datos en la bodega temporal. **Parámetros de Entrada:** Metadatos de la Bodega de datos y de los datos a almacenar. **Parámetros de Salida:** Estadísticas de almacenamiento **Condición-Activación:** Datos transformados y listos para almacenamiento final. **Condición-Finalización:** Cancelación del proceso de integración **Condición-Éxito:** Todos los datos almacenados en la bodega. **Descripción:** Mediante este objetivo, se completa el proceso de integración, llevando todos los datos ya transformados al medio donde se aplicaran las demás etapas del KDD.
SERVICIO 1 **Nombre:** Envió de datos recopilados. **Parámetros-entrada:** Tiempo aproximado de envió, localización de datos. **Parámetros-salida:** No hay parámetros de salida **Lenguaje par representar el conocimiento:** FIPA ACL **Ontología:** La desarrollada para el sistema (según el modulo del dominio y el modulo de información general).

Tabla 4.18. Agente 3. "Agente Integrador"

Nombre: Agente Coordinador
Tipo: Agente de software estacionario.
Papel o Rol: Administrador de actividades
Descripción: Se responsabiliza de coordinar el desarrollo de los procesos de recopilación e integración de datos,
OBJETIVO 1 **Nombre:** Asignar fuentes de datos **Parámetros de Entrada:** Metadatos de la fuentes y número total de fuentes **Parámetros de Salida:** No posee parámetros de salida **Condición-Activación:** Inicio del proceso de recopilación **Condición-Finalización:** Cancelación proceso **Condición-Éxito:** Todas las fuentes asignadas **Descripción:** Cada fuente es asignada a un agente al cual es entregado tanto la información de acceso de la misma como los metadatos para su exploración.
OBJETIVO 2 **Nombre:** Monitorear avance de procesos **Parámetros de Entrada:** No posee parámetros de entrada **Parámetros de Salida:** Reporte de avance de proceso **Condición-Activación:** Solicitud reporte de avance **Condición-Finalización:** Cancelación de petición **Condición-Éxito:** Generación de reporte de avance **Descripción:** Se solicita información en todos los puntos de los procesos con la finalidad de identificar como a avanzado el proceso.
SERVICIO 1 **Nombre:** Comunicación de eventualidades **Parámetros-entrada:** Fecha, ubicación y nivel de la eventualidad **Parámetros-salida**: No posee parámetros de salida. **Lenguaje par representar el conocimiento:** FIPA ACL **Ontología:** La desarrollada para el sistema (según el modulo del dominio y el modulo de información general).

Tabla 4.19. Agente 4. "Agente Coordinador"

Nombre: Agente Usuario Analista
Tipo: Agente Humano
Papel o Rol: Es quien se encarga de actualizar la información de las fuentes y analizar el desempeño de todo el proceso.
Descripción: Persona(s) que se encarga de monitorear y velar por el buen desarrollo de las etapas de interés.
OBJETIVO 1 **Nombre:** Solucionar la ocurrencia de fallas **Parámetros de Entrada:** Información de la(s) falla(s) **Parámetros de Salida:** Reporte de hallazgo y corrección de fallas. **Condición-Activación:** Reporte de una falla. **Condición-Finalización:** Aplicado tratamiento a falla. **Condición-Éxito:** Falla corregida **Descripción:** Con este objetivo se pretende corregir la mayor cantidad de fallas ocurridas en el menor tiempo posible y dando la posibilidad a que no se reinicie el proceso, en vez de esto que se continúe.
OBJETIVO 2 **Nombre:** Analizar reportes de avances **Parámetros de Entrada:** Reporte generado **Parámetros de Salida:** No posee **Condición-Activación:** Reportes disponibles sin analizar. **Condición-Finalización:** Postergación de análisis. **Condición-Éxito:** Reporte revisado y concluido. **Descripción:** Objetivo que busca identificar la efectividad de las tecnologías, arquitectura y paradigmas utilizados.
OBJETIVO 3 **Nombre:** Actualizar manualmente metadatos de las fuentes de interés **Parámetros de Entrada:** Nueva información **Parámetros de Salida:** No posee parámetros de salida **Condición-Activación:** Cambios por realizar en las fuentes **Condición-Finalización:** Cambios realizados. **Condición-Éxito:** Todos los cambios hechos. **Descripción:** Busca ampliar, mejorar y corregir, el conocimiento que se tiene para trabajar con los datos de interés.
SERVICIO(S) Debido a que este agente es de tipo humano no ofrece ninguna tarea a otros agentes del sistema y por lo tanto no ofrece ningún servició. Este agente usuario requiere de los servicios de otros agentes, pero no presta ningún servicio.

Tabla 4.20. Agente 5. "Agente Usuario Analista"

Nombre: Agente Analista
Tipo: Agente de software estacionario.
Papel o Rol: Presentador y creador de informes y contenidos
Descripción: Persona(s) que se encarga de monitorear y velar por el buen desarrollo de las etapas de interés.
OBJETIVO 1 **Nombre:** Reportar ocurrencia de fallas **Parámetros de Entrada:** Tipo de falla, localización, fecha **Parámetros de Salida:** Reporte de falla. **Condición-Activación:** Ocurrencia de una falla. **Condición-Finalización:** Envió de reporte **Condición-Éxito:** Generación y envió de reporte de falla **Descripción:** Con este objetivo se pretende mantener control sobre la ocurrencia de fallas durante todo el desarrollo de las etapas de interés.
OBJETIVO 2 **Nombre:** Construir reportes de avances **Parámetros de Entrada:** Reporte generado **Parámetros de Salida:** No posee **Condición-Activación:** Reportes disponibles sin analizar. **Condición-Finalización:** Postergación de análisis. **Condición-Éxito:** Reporte revisado y concluido. **Descripción:** Objetivo que busca identificar la efectividad de las tecnologías, arquitectura y paradigmas utilizados.
OBJETIVO 3 **Nombre:** Presentar contenidos **Parámetros de Entrada:** Datos o reportes a presentar **Parámetros de Salida:** No posee **Condición-Activación:** Reportes disponibles sin analizar. **Condición-Finalización:** Postergación de análisis. **Condición-Éxito:** Reporte revisado y concluido. **Descripción:** Objetivo que busca identificar la efectividad de las tecnologías, arquitectura y paradigmas utilizados.
SERVICIO(S) Este agente no posee servicios

Tabla 4.21. Agente 6. "Agente Analista"

4.2.2 Modelo de Tareas

El modelo de tareas permite describir las actividades relacionadas para alcanzar un objetivo. La meta de este modelo es documentar la situación actual y futura de la organización, facilitar la gestión de cambios, y ayudar a estudiar el alcance y viabilidad del sistema inteligente que se desea desarrollar. Las tareas cognitivas que se deseen implementar se detallarán en un modelo de la experiencia, mientras que las tareas de comunicación se detallarán en un modelo de comunicación (comunicación humana) o coordinación (comunicación con agentes).

Una tarea se refiere al conjunto de actividades que se realizan para conseguir un objetivo en un dominio dado. Para cada tarea se especifican los siguientes ítems:

- Nombre: Es una cadena de texto corta y debe ser única.
- Objetivo: Especificación de cómo transformar las entradas en salidas.
- Entrada: Identifica el tipo de ingrediente que se emplea como entrada de una tarea.
- Salida: Identifica el tipo de ingrediente producido o transferido por la tarea.
- Sub-tareas: Sub-actividades de la tarea para conseguir su objetivo.
- Uso de modelos: Relaciones de dependencia que establece con otros modelos de la arquitectura.

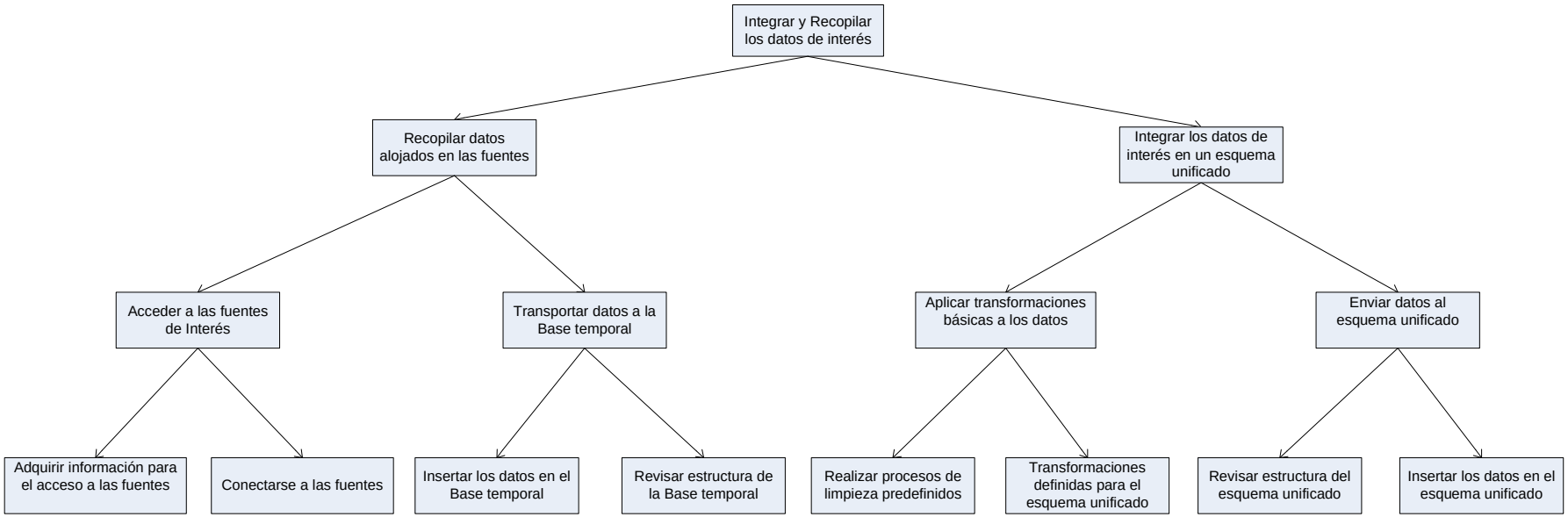


Figura 4.6 Modelo de Tareas.

A continuación se describirán cada una de las tareas que fueron identificadas (Figura 4.7):

Nombre: Recopilar datos en las fuentes
Objetivo: Capturar todos los datos necesarios para la realización del proceso de KDD
Entrada: Metadatos sobre acceso y estructura de las fuentes de datos.
Salida: Datos recopilados
SubTareas: 1.1 Acceder a las fuentes de interés, 1.2 Transportar datos a la base temporal.
Uso de Modelos: Modelo del Dominio

Tabla 4.22. Tarea 1. Recopilar datos en las fuentes

Nombre: Acceder a las fuentes de interés
Objetivo: Lograr establecer un enlace con las fuentes para consiguientemente captura los datos
Entrada: Información de acceso a la fuente
Salida: Estado del enlace de acceso
SubTareas: 1.1.1 Adquirir información para acceder a la fuentes, 1.1.2 Realizar conexión a las fuentes
Uso de Modelos: Modelo del Dominio

Tabla 4.23. Tarea 1.1 Acceder a las fuentes de interés

Nombre: Transportar datos a la base temporal
Objetivo: El objetivo de esta tarea es trasladar los datos que fueron recopilados en las fuentes a un lugar local.
Entrada: Datos capturados
Salida: Datos alojados en la base temporal
SubTareas: 1.2.1 Analizar estructura de la base temporal, 1.2.2 Insertar datos recopilados en la base temporal.
Uso de Modelos: Modelo del Dominio

Tabla 4.24. Tarea 1.2 Transportar datos a la base temporal

Nombre: Adquirir información para acceder a la fuentes
Objetivo: Lograr conseguir los metadatos necesarios para construir un enlace con una fuente en especifico
Entrada: Identificador de la fuente
Salida: Metadatos de la fuente
SubTareas: Ninguna
Uso de Modelos: Modelo del Dominio

Tabla 4.25. Tarea 1.1.1 Adquirir información para acceder a la fuentes

Nombre: Realizar conexión a las fuentes
Objetivo: Establecer un enlace con la fuente que permita la adquisición posterior de los datos de interés.
Entrada: Metadatos de la fuente
Salida: Enlace establecido con la fuente
SubTareas: Ninguna
Uso de Modelos: Modelo del Dominio

Tabla 4.26. Tarea 1.1.2 Realizar conexión a las fuentes

Nombre: Analizar estructura de la base temporal
Objetivo: Entender la estructura de la base temporal con el propósito de lograr llevar los datos recopilados a la misma.
Entrada: Información Base temporal
Salida: Mapeo de estructura de datos recopilados en la base temporal.
SubTareas: Ninguna
Uso de Modelos: Modelo del Dominio

Tabla 4.27. Tarea 1.2.1 Analizar estructura de la base temporal

Nombre: Insertar datos recopilados en la base temporal
Objetivo: Lograr trasladar los datos de interés de las diferentes fuentes a una base de datos en la cual puedan ser alterados.
Entrada: Mapeo base temporal – datos recopilados
Salida: Datos alojados en la base temporal
SubTareas: Ninguna
Uso de Modelos: Modelo del Dominio

Tabla 4.28. Tarea 1.2.2 Insertar datos recopilados en la base temporal

Nombre: Integrar datos de interés en un esquema unificado
Objetivo: Aplicar todos los procesos necesarios para llevar los datos que han sido recopilados a un esquema unificado mediante el cual se facilitara el desarrollo de las etapas posteriores del KDD
Entrada: Datos recopilados
Salida: Datos transformados y alojados en la bodega de datos.
SubTareas: 2.1 Aplicar transformaciones a los datos, 2.2 Alojar datos en la Bodega de datos.
Uso de Modelos: Modelo del Dominio

Tabla 4.29. Tarea 2 Integrar datos de interés en un esquema unificado

Nombre: Aplicar transformaciones a los datos
Objetivo: Modificar los datos para que puedan ser almacenados en la bodega de datos
Entrada: Datos almacenados en la base temporal
Salida: Datos transformados
SubTareas: 2.1.1 Realizar procesos de limpieza predefinidos, 2.1.2 Transformar datos al esquema unificado
Uso de Modelos: Modelo del Dominio

Tabla 4.30. Tarea 2.1 Aplicar transformaciones a los datos

Nombre: Alojar datos en la Bodega de datos
Objetivo: Depositar los datos en la bodega para facilitar la realización del resto de las etapas del KDD.
Entrada: Datos transformados
Salida: Datos almacenados en la bodega de datos.
SubTareas: 2.2.1 Analizar el esquema unificado, 2.2.2 Realizar la inserción de datos en la bodega de datos
Uso de Modelos: Modelo del Dominio

Tabla 4.31. Tarea 2.2 Alojar datos en la Bodega de datos

Nombre: Realizar procesos de limpieza predefinidos
Objetivo: Tratar datos que presentan características diferentes a los demás datos asociados a ellos mediante transformaciones simples.
Entrada: Datos alojados en la base temporal
Salida: Datos Semi-tratados (Limpieza parcial)
SubTareas: Ninguna
Uso de Modelos: Modelo del Dominio

Tabla 4.32. Tarea 2.1.1 Realizar procesos de limpieza predefinidos

Nombre: Transformar datos al esquema unificado
Objetivo: Realizar transformaciones que sean requeridas por los datos
Entrada: Datos semi-tratados
Salida: Datos transformados
SubTareas: Ninguna
Uso de Modelos: Modelo de Dominio

Tabla 4.33. Tarea 2.1.2 Transformar datos al esquema unificado

Nombre: Analizar el esquema unificado
Objetivo: Entender el esquema unificado con la finalidad de poder llevar a él, datos que vienen de esquemas diversos.
Entrada: Metadatos del esquema unificado y de la base temporal
Salida: Mapeo base temporal a esquema unificados
SubTareas: Ninguna
Uso de Modelos: Modelo del Dominio

Tabla 4.34. Tarea 2.2.1 Analizar el esquema unificado

Nombre: Realizar la inserción de datos en la bodega de datos
Objetivo: llevar todos los datos transformados a la bodega de datos, logrando con esto terminar la etapa de integración de datos.
Entrada: Datos transformados y Mapeo Base - esquema
Salida: Datos almacenados
SubTareas: Ninguna
Uso de Modelos: Modelo del Dominio

Tabla 4.35. Tarea 2.2.2 Realizar la inserción de datos en la bodega de datos

4.2.3 Modelo de la experiencia

La función principal de este modelo es identificar la ontología o las ontologías que son utilizadas por los agentes en el dominio del problema.

El manejo de las ontologías juega un papel muy importante en el desarrollo del sistema, ya que amplía el conocimiento acerca de las variables que se deben tener en cuenta a la hora de tratar un problema determinado.).

En este problema en particular se va a manejar un tipo de ontología enfocada al manejo de los procesos involucrados en el desarrollo de la Recopilación e Integración de datos, y es descrita haciendo uso de diagramas de clase como se muestra en la figura 4.7.

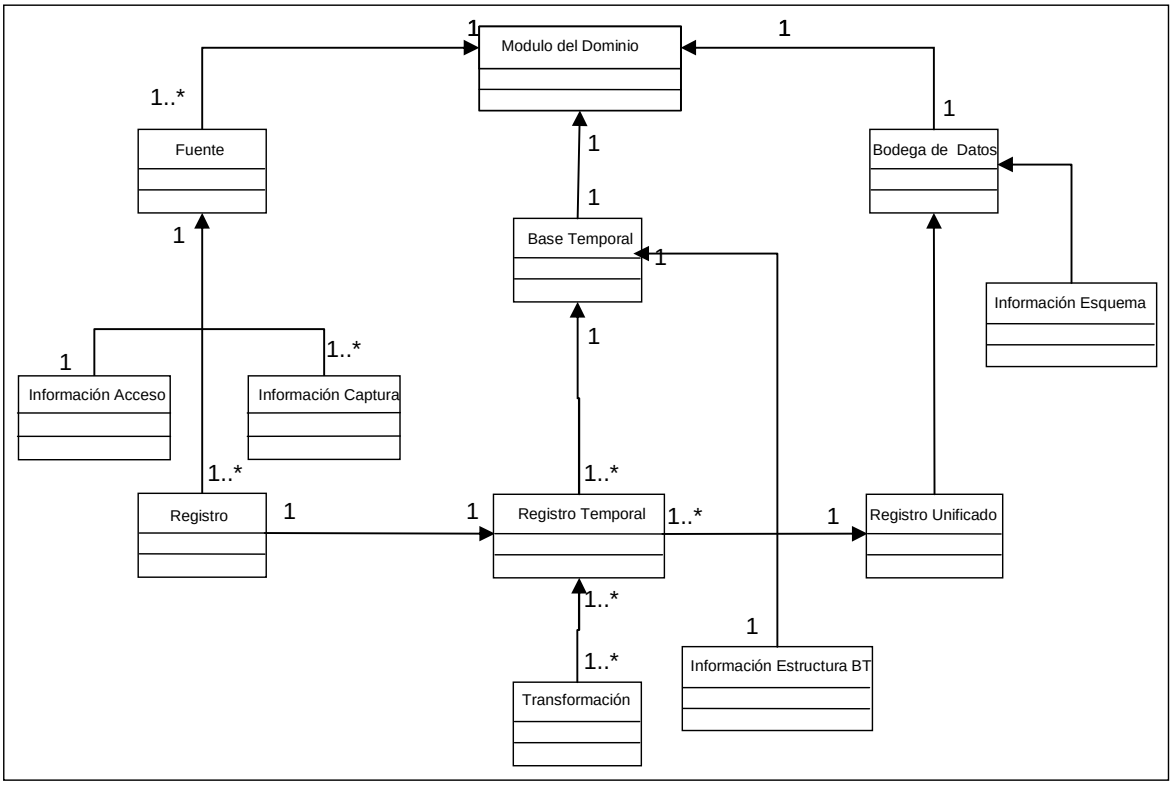


Figura 4.7 Modelo del Dominio.

4.2.4 Modelo de Coordinación y Comunicación

El principal objetivo del modelo de coordinación es modelar la interacción agente-agente. Esta interacción engloba la interacción "máquina-máquina" y "hombre-máquina", pues resulta más cómodo para el sistema homogeneizar todas las interfaces, sin embargo, el estudio de las interacciones concretas "hombre-máquina" y de los factores humanos que deben tenerse en cuenta para el diseño de tales interfaces son definidos mediante el modelo de comunicación. Seguidamente se presentan los diagramas de secuencias identificados.

Nombre:	D.S Comunicación y tratamiento a eventualidades
Agentes relacionados:	Ag Recolector, Ag Coordinador y Ag Usuario Analista.
Descripción:	Este diagrama corresponde a las tareas de comunicar y resolver una eventualidad que ocurra en el proceso de adquirir datos de las fuentes.

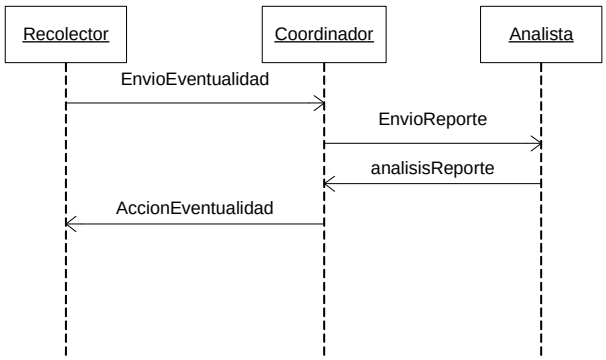


Tabla 4.36. D.S. Comunicación y tratamiento a eventualidades

Nombre:	D.S Adquisición de Información de acceso
Agentes relacionados:	Ag Recolector, Ag Coordinador
Descripción:	Muestra la comunicación que se presenta entre estos agentes con la finalidad de adquirir la información necesaria para acceder a una fuente en específico.

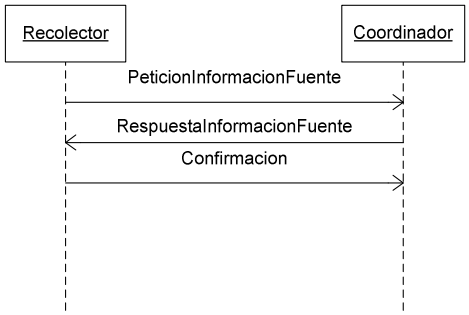


Tabla 4.37. D.S Adquisición de Información de acceso

Nombre:	D.S Asignación de una fuente
Agentes relacionados:	Ag Coordinador y Ag Recolector
Descripción:	Para la asignación de fuentes el Ag Coordinador se comunica con un Ag Recolector y verifica su disponibilidad para encargarse de una fuente

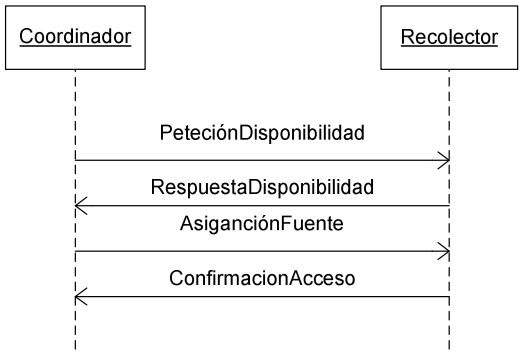


Tabla 4.38. D.S. Asignación de una fuente

Nombre:	D.S Envío de datos Fuentes - Base temporal
Agentes relacionados:	Ag Recolector y Ag Almacenista
Descripción:	Comunicación que permite el envío de todos los datos recopilados al Ag. Almacenista

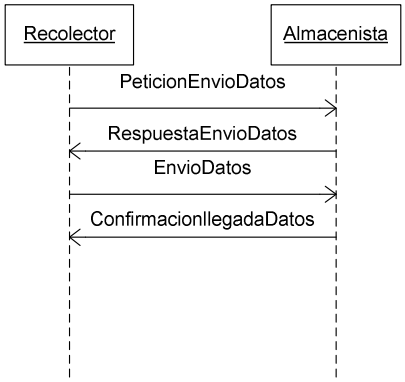


Tabla 4.39. D.S. Envío de datos Fuentes - Base temporal

Nombre:	D.S Envío datos Base temporal – Bodega de Datos
Agentes relacionados:	Ag Almacenista y Ag Integrador
Descripción:	Comunicación que permite el envío de todos los datos almacenados en la base temporal al Ag. Almacenista.

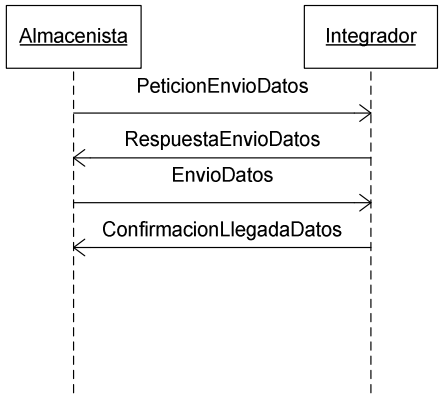


Tabla 4.40. D.S. Envío datos Base temporal – Bodega de Datos

Nombre:	D.S Avance en los procesos de Recopilación e Integración
Agentes relacionados:	Ag Analista, Ag Coordinador, Ag Recolector, Ag Almacenista y Ag Integrador
Descripción:	La construcción de un informe estado de los procesos requiere la comunicación entre todos los agentes cada uno enviando información acerca de cómo van las tareas que tiene a cargo.

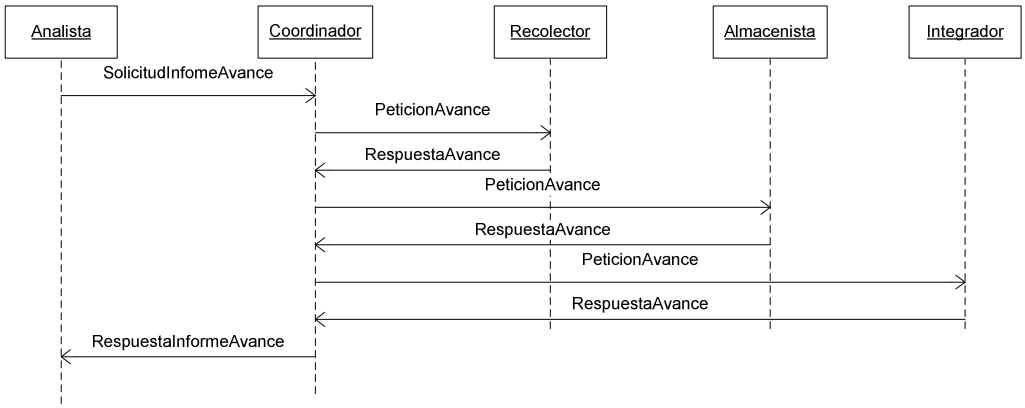


Tabla 4.41. D.S. Avance en los procesos de Recopilación e Integración

Nombre:	D.S Comunicación de Actualización Realizada
Agentes relacionados:	Ag Analista, Ag Coordinador, Ag Recolector, Ag Almacenista y Ag Integrador
Descripción:	Este proceso busca permitir que las actualizaciones que ocurran dependiendo del nivel se comuniquen a los agentes involucrados

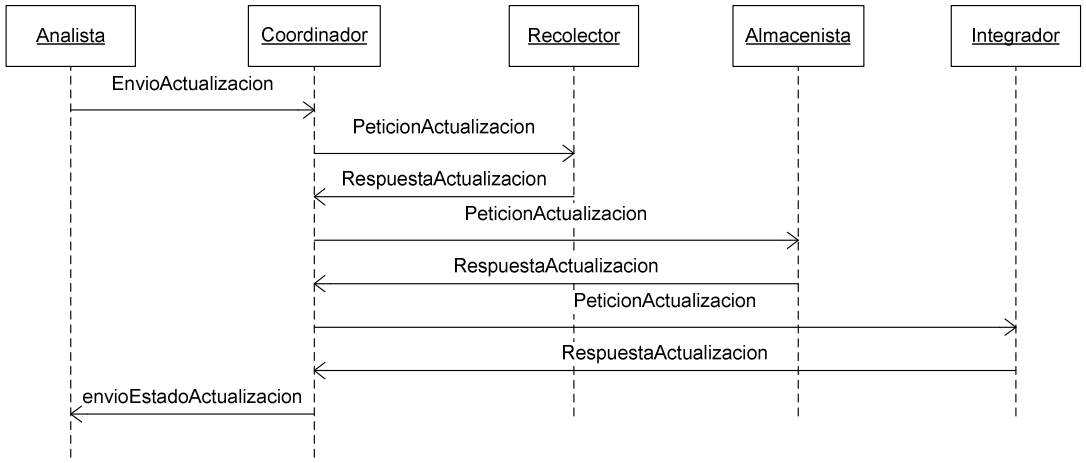


Tabla 4.42. D.S. Avance en los procesos de Recopilación e Integración

4.2.5 Modelo de la Organización

Este modelo describe tanto la organización humana en que el Sistema Multi-Agente va a ser introducido como la organización interna del mismo. El objetivo principal es analizar desde una perspectiva de grupo las relaciones entre los agentes (tanto software como humanos) que interactúan con el sistema.

La estructura organizacional del modelo propuesto se puede apreciar en la Figura 4.9

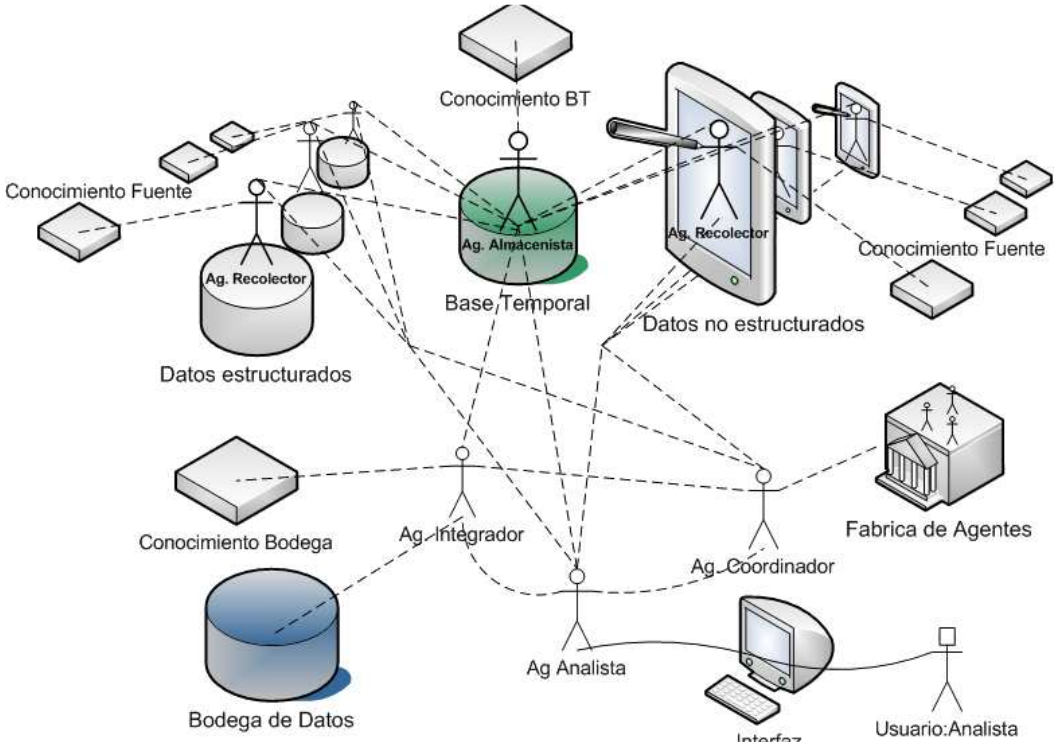


Figura 4.8 Modelo del la Organización.

4.3 Diseño

Esta fase se define como transformar las descripciones de los modelos de la fase de análisis en un sistema real. MAS-CommondKADS 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, y los protocolos de comunicación. Esta fase consiste en la construcción de:

- El diseño de la Red
- El diseño de los agentes
- El diseño de la plataforma

4.3.1 Modelo de Red

En este apartado se propone determinar las funcionalidades del modelo de red y los agentes de red encargados de llevarlas a cabo. Para el modelo propuesto no se definen agentes de red, ya que se pretende desarrollar este modelo bajo la plataforma JADE (Java Agent Develpment Framework), en la cual se presentan un conjunto de agentes por defecto, que están encargados de las diferentes tareas y

procesos de red. En el anexo 1 se presenta esta información de una manera más extensa.

4.3.2 Modelo de Agente

En este modelo se especifican los diferentes tipos de agentes que existirán en el modelo dependiendo de los roles que lleven a cabo, así como las instancias de dichos agentes durante la ejecución. La Figura 4.10 presenta el árbol de tipos de agentes identificados

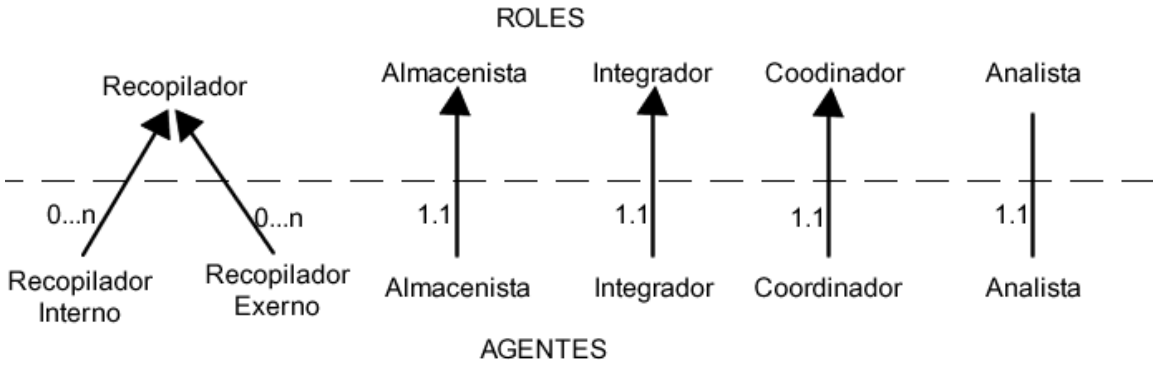


Figura 4.9 Árbol de Tipos de Agentes.

4.3.3 Modelo de Plataforma

En este modelo se documentan las características de la plataforma sobre la que se construirá el modelo. El diseño de plataforma planteado presenta un sistema distribuido desarrollado mediante el paradigma de sistemas multi-agentes el cual se instancia bajo la plataforma JADE, distribuida en una red LAN. El diagrama de despliegue correspondiente se presenta en la figura 4.10.

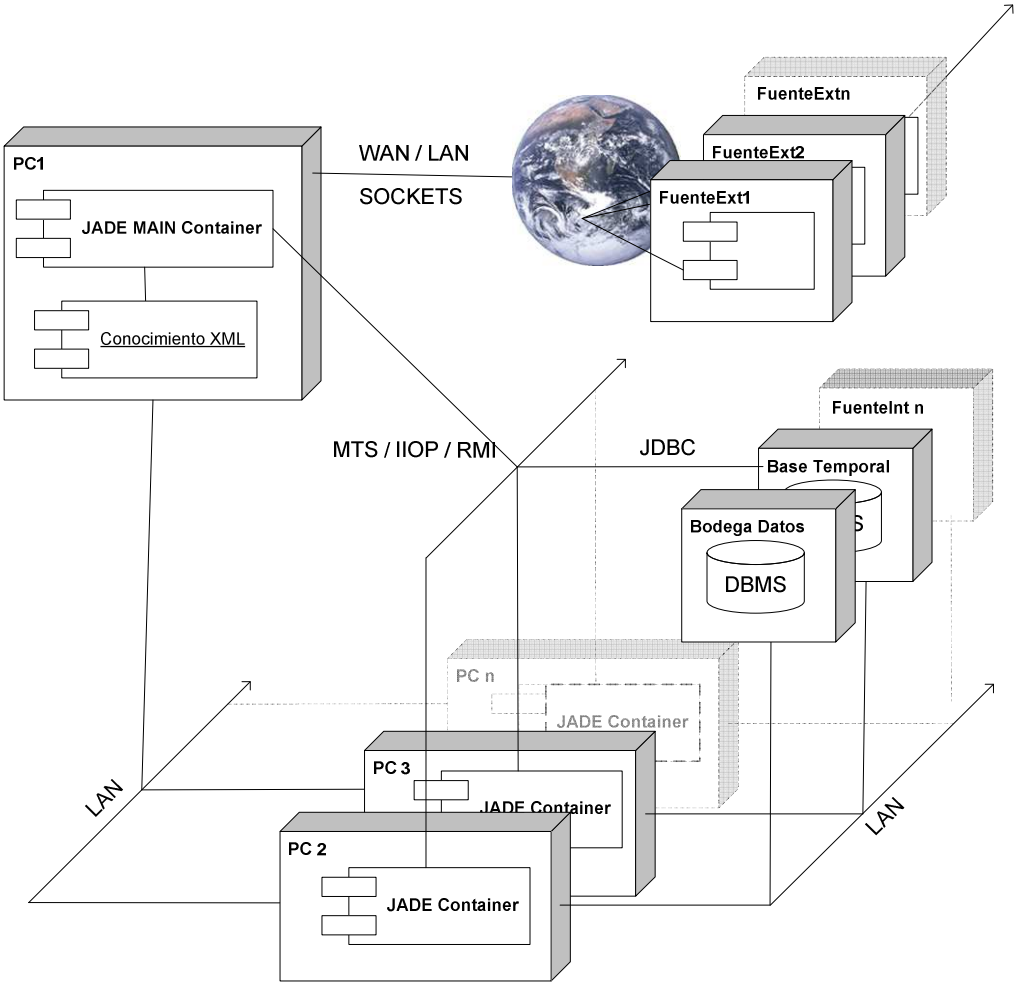


Figura 4.10 Diagrama de Despliegue.

4.4 Reflexión

En este capítulo se presentó el desarrollo de la metodología MASCommonKADS para el desarrollo de las etapas de recopilación e integración de datos, el cual es de importancia por dos razones. La primera sirve como plataforma para implementación de un prototipo de software que si bien no es la meta fundamental de esta tesis, si es de gran utilidad en el momento de validar los modelos propuestos y alcanzar el objetivo propuesto. Segundo es que proporciona un modelo formal de los agentes que intervienen el proceso de recopilación e integración de datos, así como de sus principales procesos, lo cual brinda un mayor entendimiento de dicho sistema y puede servir como punto de partida para posteriores investigaciones y desarrollos tanto conceptuales como prácticos.

CAPITULO 5

VALIDACIÓN Y ANÁLISIS DE RESULTADOS

El último objetivo de este trabajo de investigación consiste en la validación del modelo propuesto. Para cumplir tal objetivo se consideró un caso de estudio con múltiples fuentes heterogéneas, se construyó una base temporal, una bodega de datos y se implementó un prototipo considerando el modelo propuesto al que se denominó SIRD (Sistema para la Recuperación e Integración de Datos). Por medio de este caso de estudio se pretende validar por una parte el modelo en general y por otra parte analizar el desempeño de los agentes que intervienen él.

5.1 Caso de Estudio

La deserción universitaria y el retraso en los estudios son dos problemas de gran alcance nacional e Internacional. Los bajos rendimientos académicos abundan en muchos establecimientos educativos y escasean los estudiantes que cursan sus estudios y aprueban las asignaturas en los períodos establecidos. De acuerdo al estudio llevado a cabo en [Latiesa, 1996], el crecimiento de la deserción universitaria comenzó hace cuatro décadas y se fue acrecentando a finales de los noventa. Dada la importancia del tema a nivel latinoamericano, en 2005 se llevó a cabo el Seminario Internacional "Rezago y deserción en la educación superior". El informe final concluyó que los factores que inciden en la deserción estudiantil se pueden agrupar generalmente en: Condiciones socioeconómicas, aspectos de personales y aspectos académicos. La deserción académica, al igual que el retraso en la culminación a tiempo de los planes de estudios, son problemas que caracterizan en la actualidad a la mayoría de las instituciones colombianas.

Con este panorama presente, el caso de estudio ilustrado a continuación fue desarrollado para servir como base en un proyecto de Minería de Datos que buscará identificar y analizar algunos de los factores que pueden influir en la deserción y en el desempeño de estudiantes universitarios durante el transcurso de su ciclo académico.

Debido algunas restricciones en la adquisición de datos y a las ventajas en el análisis de casos de estudio controlados, se tomaron 3 fuentes de información de las cuales dos son construidas con datos simulados y una posee datos reales, tal como se muestra en la figura 5.1. Una primera fuente es un Sistema de Información

Académica (Simulada) en el cual se almacenaron datos de tipo personal y académico; una segunda fuente presenta documentos relacionados a Solicitudes de una Facultad (Simulada) como los las cancelaciones y aplazamientos de semestre; y por última un Sistema de Gestión de Cursos (Real) con datos sobre cursos virtuales activos, participación y desempeño de estudiantes.

Figura 5.1 Fuentes - Caso de Estudio

Una especificación más amplia de cada una de estas fuentes se presenta seguidamente.

5.1.1 Sistema de información Académica.

Para el diseño e implementación de esta fuente se tomó como base el Sistema SIA de la Universidad Nacional de Colombia el cual es una aplicación desarrollada con el fin de optimizar los procesos de carácter académico y administrativo en cada una de sus sedes para los docentes, estudiantes y personal administrativo. Éste también busca ampliar los mecanismos de comunicación entre los docentes y los estudiantes con el uso de herramientas de comunicación electrónica como el correo oficial.

Los servicios que conforman la aplicación se encuentran agrupados en siete categorías: Servicios de Apoyo Académico, Servicios de Apoyo a la Docencia, Servicios de Apoyo a Procesos Administrativos, Servicios del Archivo, Servicios de Información Financiera, Servicios para Búsquedas y Servicios de Libre Acceso. Para el diseño de esta fuente se tomaron en cuenta sólo dos de estos servicios: los de Apoyo Académico, en donde se pueden realizar consultas sobre horarios de clase, cursos inscritos, listas de curso, calificaciones; y los de Archivo, donde se puede realizar la consulta de datos básicos y la historia académica de un estudiante.

Con base en estos servicios se construyó el siguiente modelo de datos para esta fuente.

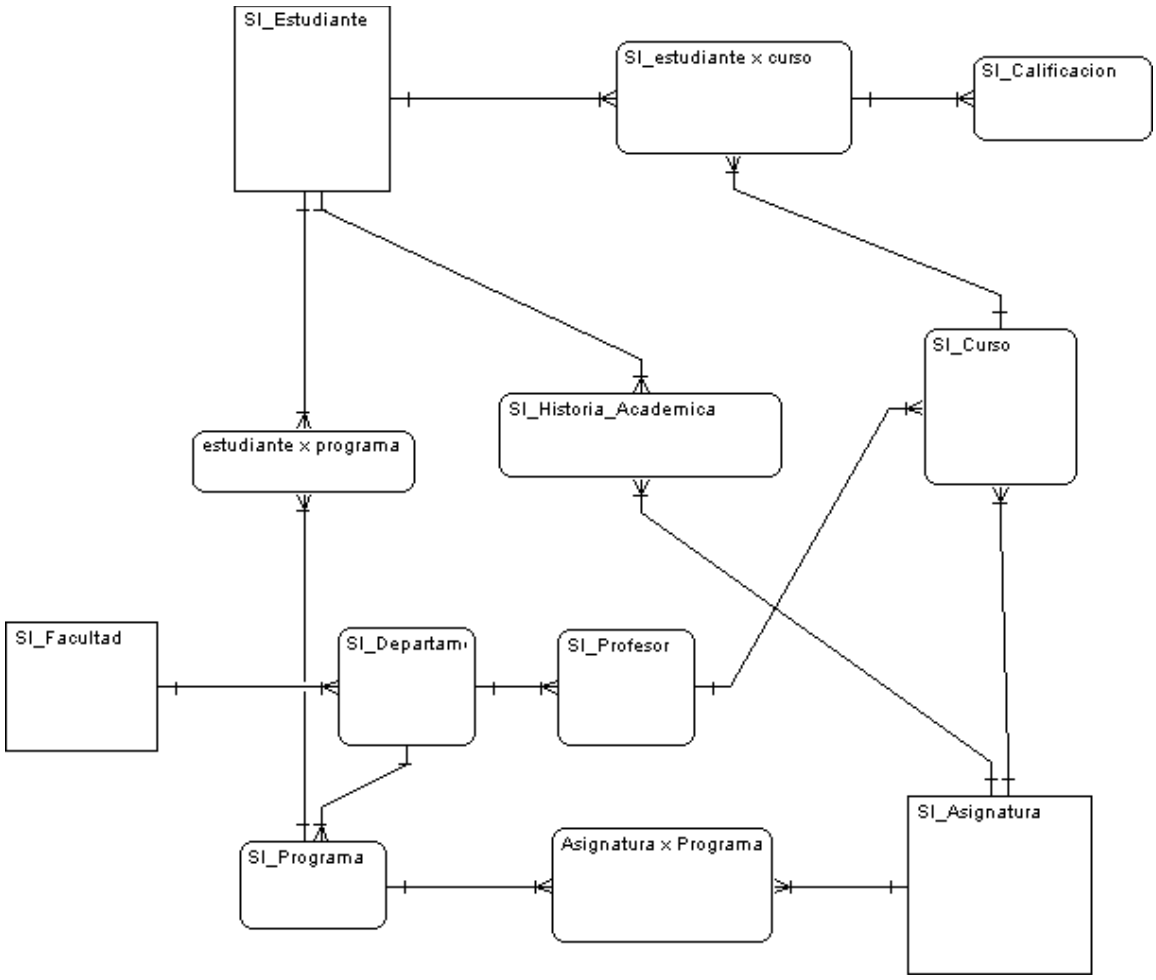


Figura 5.2 Modelo de datos Sistema de Información Académica

El Sistema de información Académica fue instanciado sobre el gestor de bases de datos PostgreSQL, presenta 12 Tablas y cerca de 79 atributos. Para esta fuente se construyeron registros para 200 estudiantes, para un total en toda en la fuente de 8327 registros.

5.1.2 Sistema de Gestión de Cursos

La fuente que representa al sistema de gestión de cursos es Moodle (Modular Object Oriented Distance Learning Enviroment). Plataforma web de distribución libre orientada a desarrollar Cursos de enseñanza virtual, apoyándose en el marco

de la teoría del constructivismo social, basado en el conocimiento sobre la teoría del aprendizaje y la colaboración.

Este sistema facilita mecanismos mediante los cuales el material de aprendizaje y las actividades de evaluación son realizados por el estudiante pero también donde los tutores o profesores pueden introducirse en el diseño y la forma de llevar el conocimiento hasta sus alumnos. Moodle está diseñada para realizar cursos interactivos virtuales que permitan:

- Presentar material didáctico para una asignatura, seminario o taller, bien en forma de lecciones, trabajos, ejercicios, cuestionarios, etc.
- Proporcionar recursos de información (en formato textual o tabular, fotografías o diagramas, audio o video, páginas web o documentos PDF entre muchos otros).
- Proporcionar diversas actividades y herramientas de comunicación para que los estudiantes interactúen entre ellos o con el profesor (foros, chats, etc.).

Aunque la versión estándar de Moodle está compuesto por un modelo de datos de alrededor de 189 tablas, para este trabajo se tomaron en cuenta 22 tablas (las más relevantes para el caso de estudio) con alrededor de 146 atributos y posee 101.956 Registros. El sistema Moodle esta instanciado sobre el gestor de base de datos MySQL y los elementos de interés del modelo de datos se presentan en la Figura 5.3.

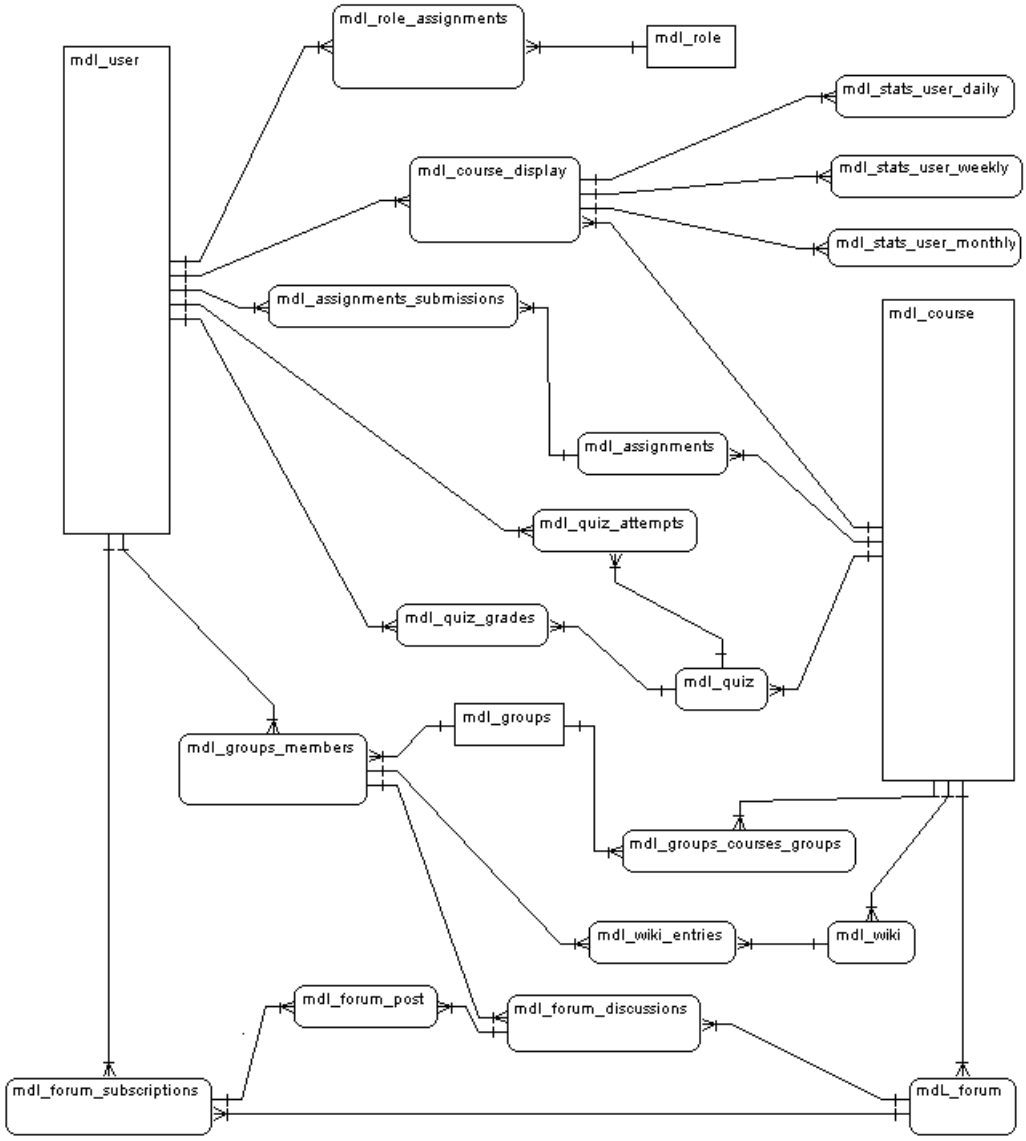


Figura 5.3 Modelo de Datos del Sistema de Gestión de Cursos Moodle.

5.1.3 Solicitudes Facultad

La fuente Solicitudes Facultad maneja registros de las solicitudes de: Inclusiones y Cancelaciones, Aplazamiento de Semestre, Validaciones, Reintegro y Cancelaciones de Semestre. Esta información es almacenada por medio de hojas de cálculo, las cuales no tienen ningún tipo de restricción de ingreso, y los formatos solo responden a un cierto tipo de plantilla definida para cada documento.

Figura 5.4 Solicitudes Facultad

Para esta fuente se construyó el siguiente número de registros para los años entre el 2005 y 2009.

Años/Documento	**2005**	**2006**	**2007**	**2008**	**2009**
Reingreso, Aplazamiento y Cancelación de Semestre	94	104	95	92	100
Inclusión y Cancelación de Asignaturas	33	42	75	50	66
Validación de Asignaturas	15	22	18	29	26

Tabla 5.1 Cantidad de datos en la fuente Solicitudes Facultad

Vistos como un conjunto de registros totales se generaron 858 registros.

5.1.4 Base Temporal y Bodega de Datos

Como se enunció en los capítulos anteriores para llevar a cabo las etapas de recopilación e integración de Datos se recomienda definir tanto una base de datos auxiliar que permita el almacenamiento temporal y un tratamiento básico de los registros; como una bodega de datos donde se integren y estructuren adecuadamente estos registros para su posterior análisis.

El modelos de datos presentado en la Figura 5.5 para la base temporal busca poder almacenar todos los registros que sean recopilados en las diferentes fuentes presentes en el caso de estudio. Por tanto este modelo se puede ver como una unificación de los modelos que se encuentran en cada una de las fuentes. Para en este caso en específico se decidió conservar todos los atributos de las fuentes estructuradas (Sistemas de Información y Académica y Moodle) y se agregaron 5 tablas adicionales que permiten el ingreso de los registros extraídos de la fuente

Solicitudes de Facultad. De este modo el modelo de datos de la base temporal posee 41 Tablas y 384 atributos. Este modelo está construido sobre el Gestor de bases de datos Oracle.

Para la construcción del modelo de datos de la Bodega de datos presentado en la Figura 5.6 se utilizó el Esquema de Estrella, el cual es un modelo basado en hechos y dimensiones, donde se poseen tablas de hechos que contienen los datos para el análisis, rodeadas de tablas de dimensiones. Se desea dejar claro que aunque el modelo de datos presentado no pretende ser el más eficiente (ya que no hace parte del alcance de esta investigación) se presta para la realización adecuada de la etapa de integración de datos. El modelo de datos de la bodega posee 5 tablas de hechos, 26 tablas de dimensiones y 209 atributos. Este modelo está construido sobre el Gestor de bases de datos Oracle.

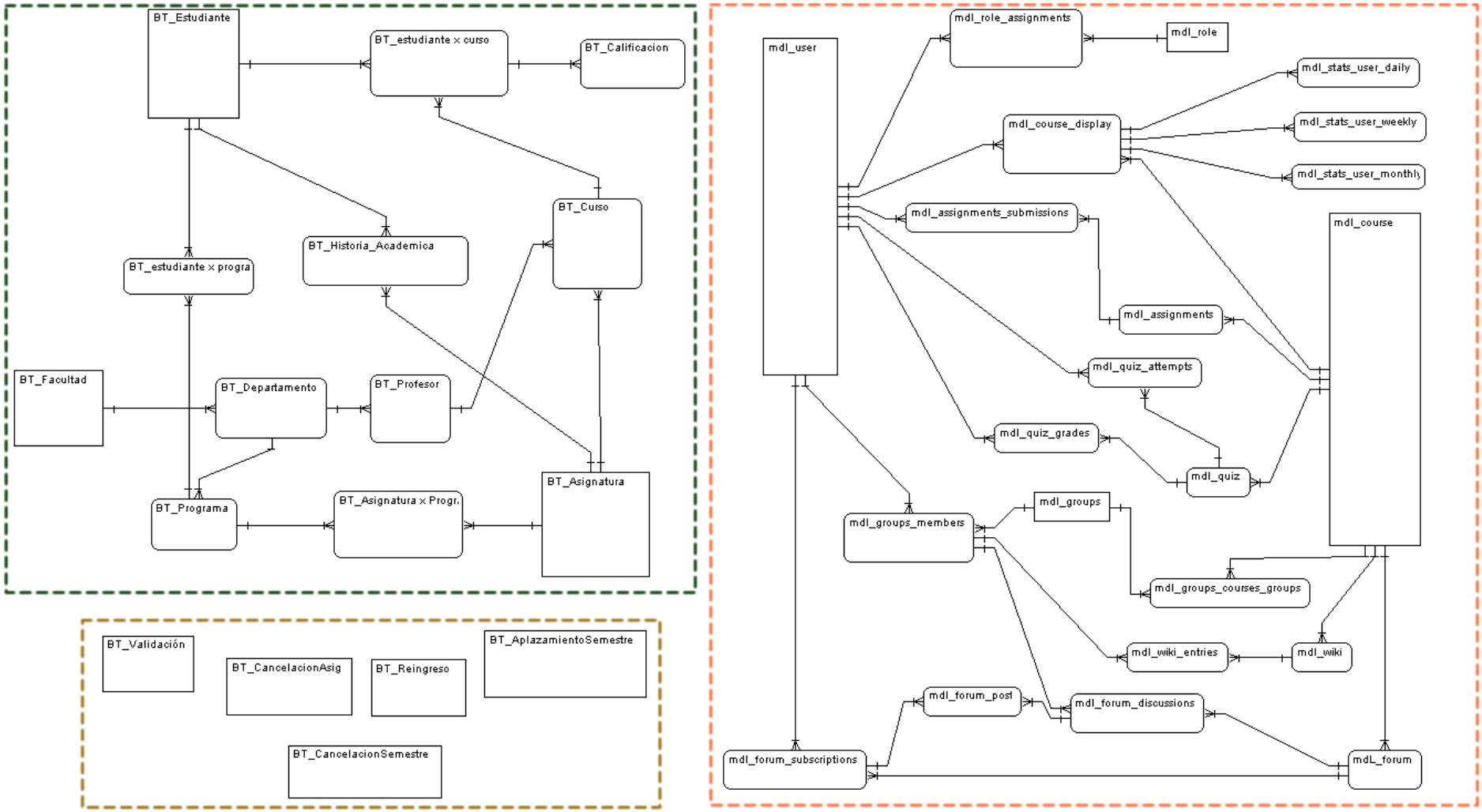


Figura 5.5 Modelo de datos Base Temporal

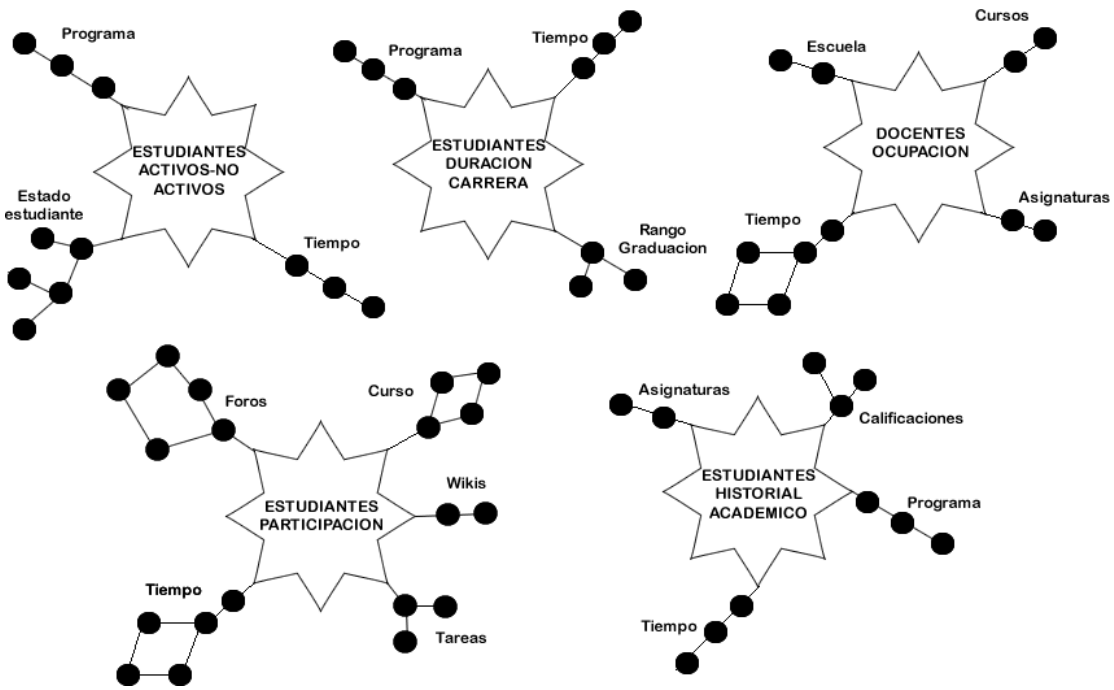


Figura 5.6 Modelo de datos Bodega de Datos (Esquema Estrella)

5.1.5 Distribución Física

Los datos del caso de estudio están distribuidos físicamente en tres servidores. En el Servidor 1 se encuentran el Sistema de Cursos Moodle y las Solicitudes de Facultad. En el Servidor 2 se encuentra el Sistema de Información Académica. Por el último en el Servidor 3 están alojadas la base temporal y la bodega de datos.

5.2 Instanciación del Modelo Propuesto

Dadas las características del caso de estudio exhibidas en el apartado anterior se presenta seguidamente la instanciación del modelo propuesto.

5.2.1 Distribución de agentes

De acuerdo a la distribución física y al número de fuentes disponibles en el caso de estudio, se presenta un Sistema Multi-Agente distribuido en una LAN, donde se encuentran 3 agentes recolectores destinados cada uno a una fuente específica, los Agentes Recolectores 1 y 2 se encuentran alojados en el Servidor 1 y el Agente Recolector 3 en el Servidor 2. Un agente almacenista encargado del manejo de la Base Temporal así como un agente Integrador responsable de los procesos de integración y manejo de la Bodega de datos se ubicaron en el Servidor 3. El Agente

Coordinador encargado de la distribución y control de los agentes recolectores así como el Agente Analista encargado de la generación y presentación de informes se alojaron en el Servidor 2.

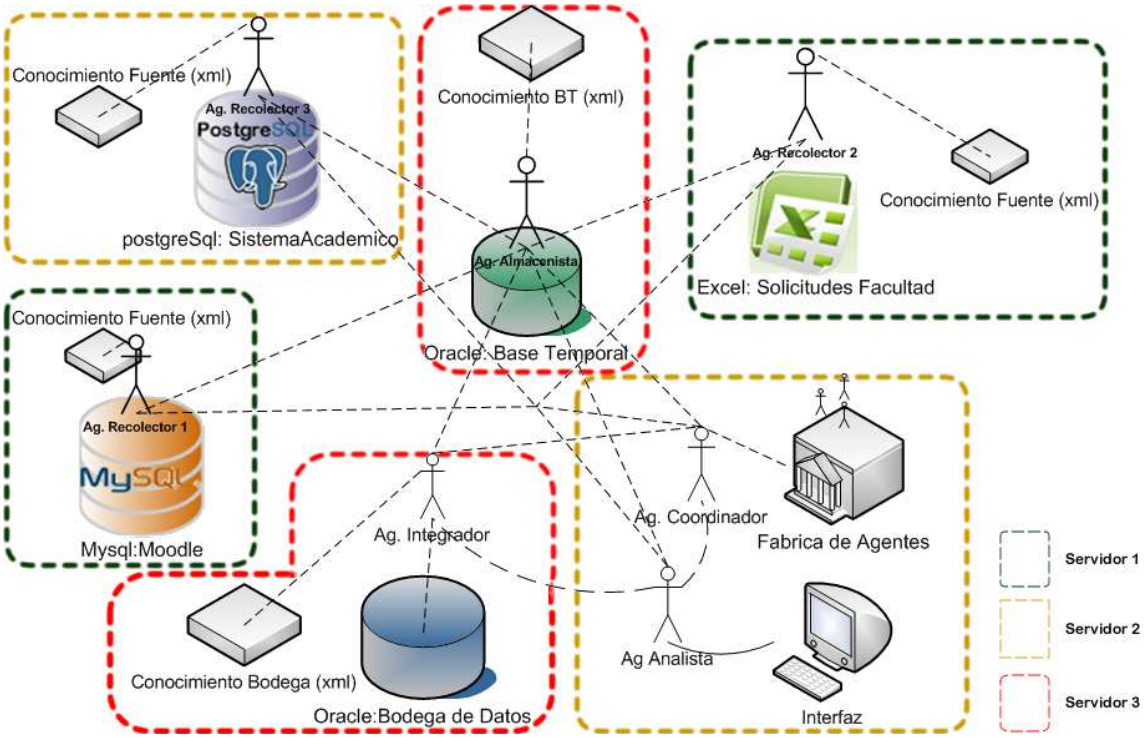


Figura 5.7 Modelo de la Organización Instanciado al Caso de Estudio

5.2.2 Definición de Conocimiento Dominio

Según lo visto en algunas secciones del capítulo 3 donde se propuso definir gran parte del conocimiento a través de plantillas en lenguaje XML, para el caso de estudio se presentan 5 Documentos XML dentro de los cuales 3 abarcan las actividades de acceso y captura de registros; otro adicional permite el mapeo de los registros capturados a la base temporal; y un último ayuda con las trasformaciones necesarias que deben sufrir los registros así como con el mapeo que va a permitir llevarlos a la bodega de datos.

5.3 Definición de Indicadores

Con la finalidad de medir y verificar el adecuado funcionamiento del modelo propuesto se definieron los siguientes indicadores de resultados.

Indicador de Efectividad en la Recopilación (IER): Indicador que pretende medir el resultado del acceso y captura de registros asociados a las fuentes de interés. La fórmula para su cálculo se presenta a continuación.

$$IER = \left(\frac{RC}{RTIF}\right) * 100\ [\%]$$

Donde:
RC=Registros Capturados de las fuentes
RTIF=Registros Totales de Interés en las Fuentes

Indicador de Efectividad en Carga Base Temporal (IECBT): Mide el resultado del proceso de carga de todos los registros recopilados en la base temporal. La fórmula para su cálculo se presenta a continuación.

$$IECBT = \left(\frac{RABT}{RC}\right) * 100\ [\%]$$

Donde:
RABT=Registros Almacenados en la Base Temporal
RC=Registros Capturados de las fuentes

Dado que la mayoría de problemas de carga de registros en base temporal se deben a la procedencia de registros de fuentes semi-estructuradas o no estructuradas, se presenta el siguiente indicador que pretende medir la efectividad en la carga de registros en la base temporal según la fuente de procedencia.

Indicador de Efectividad en Carga Base Temporal según la fuente de procedencia (IECBFi): Mide el resultado del proceso de carga de todos los registros recopilados en la base temporal, pertenecientes a una fuente i. La fórmula para su cálculo se presenta a continuación.

$$IECBF_i = \left(\frac{RABTF_i}{RCF_i}\right) * 100\ [\%]$$

Donde:
$RABTF_i$: Registros Almacenados en la Base Temporal procedentes de la F_i
RCF_i: Registros Capturados de la F_i

Indicador de desempeño del proceso de integración y carga (IDIC): Medida que permite calcular eficiencia en la realización del proceso de transformación y carga de registros en la Bodega de datos.

$$IDIC = \left(1 - \frac{(errT + errC)}{(errT + errC) + RABD}\right) * 100$$

Donde:
errT= errores cometidos en Transformaciones realizadas
errC= errores en la Carga de registros al bodega de Datos
RABD= Registros Almacenados en la Bodega de Datos.

Indicador de duración proceso de captura y carga de registros a la base temporal (IDCCT): Presenta el tiempo invertido dado una arquitectura específica en desarrollar captura de datos de las fuentes y la carga de los mismos a la base temporal.

$$IDCCT = Tcaptura + TcargaBT\ [Segundos]$$

Donde:
Tcaptura= Duración total de la actividad de capturar todos los registros de las fuentes.
TcargaBT= Duración total de la actividad de cargar todos los registros recopilados en la Base Temporal.

Indicador de duración proceso de integración (IDI): Presenta el tiempo que se tarda en desarrollar las actividades de transformación y carga de registros a la Bodega de datos.

$$IDI = Ttransformacion + TcargaBD\ [Segundos]$$

Donde:
Ttransformacion= Duración total de las operaciones de transformación de los registros que serán almacenados en la Bodega.
TcargaBD= Duración total de la actividad de cargar de registros a la Bodega.

5.4 Análisis de Resultados

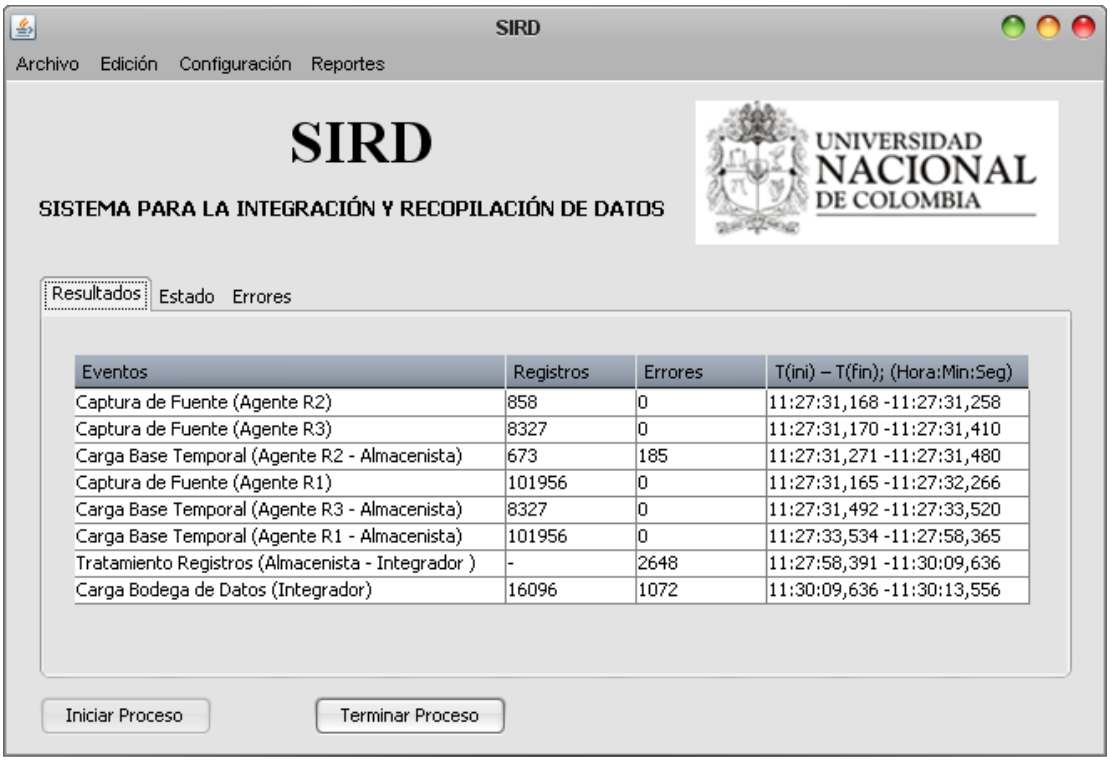
A continuación se presentan los resultados obtenidos mediante el prototipo del caso de prueba, incluyendo los valores de los indicadores definidos en el numeral anterior, sin embargo antes de ello es necesario describir las características de hardware de los servidores empleados.

Equipo	Procesador	Memoria RAM	Disco Duro
Servidor 1	Atlon XP (1.6 Ghz)	1 gb	250 Gb

Servidor 2	Atlon X2 (3 Ghz)	2 gb	500 Gb
Servidor 3	Turion X2 Ultra (2,2 Ghz)	4 gb	360 Gb

Tabla 5.2 Características de hardware Servidores

A continuación en la Figura 5.8 se presentan algunos resultados obtenidos mediante la interfaz gráfica del prototipo respecto al proceso de carga inicial de la bodega de datos.



Eventos	Registros	Errores	T(ini) – T(fin); (Hora:Min:Seg)
Captura de Fuente (Agente R2)	858	0	11:27:31,168 -11:27:31,258
Captura de Fuente (Agente R3)	8327	0	11:27:31,170 -11:27:31,410
Carga Base Temporal (Agente R2 - Almacenista)	673	185	11:27:31,271 -11:27:31,480
Captura de Fuente (Agente R1)	101956	0	11:27:31,165 -11:27:32,266
Carga Base Temporal (Agente R3 - Almacenista)	8327	0	11:27:31,492 -11:27:33,520
Carga Base Temporal (Agente R1 - Almacenista)	101956	0	11:27:33,534 -11:27:58,365
Tratamiento Registros (Almacenista - Integrador)	-	2648	11:27:58,391 -11:30:09,636
Carga Bodega de Datos (Integrador)	16096	1072	11:30:09,636 -11:30:13,556

Figura 5.8 Resultados del Sistema

Teniendo en cuenta los datos que se muestran en la Figura 5.8 se presentan a continuación los resultados para los indicadores definidos.

Para el primer indicador (*Indicador de Efectividad en la Recopilación*) tenemos el siguiente resultado

$$RC = Rc_1 + Rc_2 + Rc_3 = 101956 + 858 + 8327 = \ 111141$$

$$RTIF = \ 111141$$

$$IER = \left(\frac{RC}{RTIF}\right) * 100 = \left(\frac{111141}{111141}\right) * 100 = 100\%$$

El resultado arrojado por este indicador era de esperarse ya que las fuentes se instanciaron de forma local y no daban a lugar las restricciones de acceso, por tanto el único factor que podría afectar este indicador para el caso de estudio era una inadecuada definición del modelo del dominio, la cual no se presentó.

El segundo indicador (*Indicador de Efectividad en Carga Base Temporal*) es calculado a continuación:

$$RC = 111141$$

$$RABT = RABTF_1 + RABTF_2 + RABTF_3 = 101956 + 673 + 8327 = 110956$$

$$IECBT = \left(\frac{RABT}{RC}\right) * 100 = \left(\frac{110956}{111141}\right) * 100 = 99{,}8\%$$

En este indicador posee una alta tasa de efectividad, sin embargo denota que hubo un 0,02% de registros que no cumplieron con las restricciones de inserción presentadas por la base temporal

En el caso del tercer indicador (*Indicador de Efectividad en Carga Base Temporal según la fuente de procedencia*) se presenta el resultado para cada una de las tres fuentes:

$$Rc_1 = 101956;\ Rc_2 = 858;\ Rc_3 = 8327$$

$$RABTF_1 = 101956;\ RABTF_2 = 673;\ RABTF_3 = 8327$$

$$IECBF_1 = \left(\frac{RABTF_1}{RCF_1}\right) * 100 = \left(\frac{101956}{101956}\right) * 100 = 100\%\ (Moodle)$$

$$IECBF_2 = \left(\frac{RABTF_2}{RCF_2}\right) * 100 = \left(\frac{673}{858}\right) * 100 = 78{,}4\%\ (Solicitudes\ Facultad)$$

$$IECBF_3 = \left(\frac{RABTF_3}{RCF_3}\right) * 100 = \left(\frac{8327}{8327}\right) * 100 = 100\%\ (Sistema\ de\ Inf.Academico)$$

El resultado de las fuentes 1 y 3 es completamente satisfactorio, mientras que el de la fuente 2 (Solicitudes Facultad) presenta un valor considerablemente más bajo. Esto se debe principalmente al sistema de almacenamiento en el que se encuentra el cual no posee restricciones de inserción y por tanto algunos registros pueden no

cumplir las condiciones de las tablas en la base temporal. Esta situación no se presenta en las otras dos fuentes que son estructuradas por medio de un SGBD.

El Cuarto indicador (*Indicador de desempeño del proceso de integración y carga*) presentó el siguiente resultado.

$$errT = 2648;\ errC = \ 1072; RABD = 16096$$

$$IDIC = \left(1 - \frac{(errT + errC)}{(errT + errC) + RABD}\right) * 100$$

$$IDIC = \left(1 - \frac{(2648 + 1072)}{(2648 + 1072) + 16096}\right) * 100 = 81{,}2\%$$

El 81,2% de efectividad es este indicador es para esta investigación un gran logro dada la complejidad de esta etapa. Los registros de integración resultantes en este caso de estudio son pocos pero esto es debido principalmente a la implementación de una regla que no permite llevar registros de estudiantes a la bodega si no se encuentran dentro el Sistema de Información Académica.

Para el quinto indicador (*Indicador de duración proceso de captura y carga de registros a la base temporal*) se procedió de la siguiente manera:

Para calcular la duración total de la actividad de capturar todos los registros de las fuentes (Tcaptura) se tomó el valor inicial arrojado por el primer agente en comenzar a capturar registros y se halló la diferencia con el valor final del último agente en terminar de capturar sus respectivos registros.

$$Tcaptura = \ 11{:}27{:}31{,}165 - \ 11{:}27{:}32{,}266 = 1{,}101\ seg$$

Para calcular la duración total de la actividad de cargar todos los registros en la base temporal (TcargaBT) se realizó un procedimiento análogo tomando el valor inicial arrojado por el primer agente en comenzar a cargar registros y el valor final del último agente en terminar de capturar sus respectivos registros.

$$TcargaBT = \ 11{:}27{:}31{,}271 - \ 11{:}27{:}58{,}365 = 27{,}094\ seg$$

De esta manera el valor para este indicador es:

$$IDCCT = Tcaptura + TcargaBT = 1{,}101 + 27{,}094 = 28{,}195\ seg$$

El Sexto y último indicador (*Indicador de duración proceso de integración)* se presenta a continuación.

El valor de la duración total de las operaciones de transformación aplicadas a los registros para su posterior almacenamiento en la Bodega (Ttransformacion) se puede calcular a partir de la información presentada en la Figura 5.9. De la misma manera se puede calcular la duración de la actividad de cargar de registros a la Bodega (TcargaBD).

$$Ttransformacion = \ 11{:}27{:}58{,}391 \ - 11{:}30{:}09{,}636 = 131{,}245 \ seg$$

$$TcargaBD = \ 11{:}30{:}09{,}636 \ - 11{:}30{:}13{,}556 = 3{,}920 seg$$

$$IDI = Ttransformacion + TcargaBD = 131{,}245 + 3{,}920 = \ 135{,}165 \ seg$$

5.5 Reflexión

El caso de prueba presentado es este capítulo permitió evaluar de manera tanto cuantitativa como cualitativa de la efectividad del Modelo propuesto, respecto no sólo a la estructuración de los procesos requeridos para la Recuperación e Integración de datos (dada la efectividad de los mismos), sino también al esquema Multi-agente propuesto (dada la automatización alcanzada y la eficiencia de la misma).

Si bien las etapas posteriores del proceso de KDD no hacen parte del alcance del caso de estudio presentado, se puede decir que una vez completadas las etapas de recuperación e integración se podría proceder con dichas etapas garantizando, al menos hasta el punto donde les compete, una alta calidad de los datos almacenados en la bodega. Vale la pena agregar, de nuevo aclarando que está por fuera del alcance de este trabajo, que el caso de estudio presentado podría ser explotado en una situación real, aplicando posteriormente en la etapa de Minería de Datos análisis que podrían llegar a ser de interés como:

- ¿Qué factores parecen tener mayor incidencia con el problema de deserción estudiantil?
- ¿Existe una diferenciación de estudiantes según criterios personales (sexo, edad, raza), económicos (estrato social, valor de matrícula), familiares

(convivencia con los padres, número de hermanos), etc. respecto al desempeño académico?

- ¿Cuál es el impacto académico en los estudiantes del uso de la educación virtual como herramienta de apoyo en su proceso de aprendizaje?

Entre muchos otros.

CAPITULO 6

CONCLUSIONES Y TRABAJO FUTURO

Dada la complejidad que involucran las diferentes etapas del KDD, es clara la necesidad de desarrollar soluciones a través de enfoques innovadores que permitan incrementar la automatización y eficiencia de procesos involucrados con la finalidad reducir en la medida de lo posible el esfuerzo invertido.

Con esto en mente, esta investigación presenta una solución basada en agentes de software, y más específicamente en SMA, que pretende servir de apoyo para las etapas de Recopilación e Integración de datos dentro del proceso de KDD, esto con el fin de mejorar la calidad y velocidad de las demás etapas de este proceso como son la minería de datos y la evaluación de resultados.

Para la construcción de tal modelo se han considerado los problemas estructurales encontrados en las aproximaciones más conocidas y se ha tenido como norte lograr el mayor nivel de automatización posible en cada una de las tareas involucradas. En este sentido, más que una competencia, el modelo presentado es un complemento de dichas aproximaciones, tomando de cada uno sus principales fortalezas y logrando con esto generar una solución más completa a los diferentes problemas presentados en cada etapa.

Durante el desarrollo de esta investigación se propuso una estructuración para las etapas de interés, con la finalidad de servir de guía en el desarrollo de las etapas de interés, buscando aumentar la precisión de los datos obtenidos, optimizar el uso de recursos y mejorar la calidad en el desarrollo de las diferentes tareas involucradas.

En esta investigación se define un modelo sobre una arquitectura distribuida, escalable, basada en Sistemas Multi-Agentes que soporta el lanzamiento de agentes que integran y recopilan información en bases de datos distribuidas. Este modelo se ha implementado sobre un caso de estudio práctico con datos reales y simulados logrando resultados que demuestran la pertinencia del mismo.

A partir de los resultados obtenidos con este enfoque y considerando que las aproximaciones basadas en agentes están tornándose cada vez más importantes debido a su generalidad, flexibilidad, modularidad y su capacidad para aprovechar sistemas de recursos distribuidos, puede decirse que este paradigma merece

especial interés en procesos como el tratado en esta investigación gracias al buen desempeño para reducir trabajo y sobrecarga de información en tareas complejas, convirtiéndolo en una alternativa eficiente para la computación distribuida.

Como trabajo futuro se pretende dotar de otros mecanismos de autonomía e inteligencia al Modelo de Agentes propuesto con el propósito de avanzar hacia desarrollo cada vez más efectivo y eficiente de cada una de las tareas involucradas. También se pretende incrementar aún más la automatización, mediante modelos más parametrizables que se puedan adaptar fácilmente a diferentes entornos y cambios en los mismos.

A futuro también se tiene como objetivo expandir el modelo para que pueda abarcar otras de las etapas del proceso KDD.

REFRENCIAS BIBLIOGRAFICAS

[Bellifemine et al, 2001] Bellifemine, F., Poggi, A. & Rimassa R. "JADE: a FIPA2000 compliant agent development environment". International Conference on Autonomous Agents. Pag: 216 – 217. ISBN:1-58113-326-X. 2001

[Bernstein & Haas, 2008] Bernstein, P. A.; Haas, L. M. "Information integration in the Enterprise". Communications of the ACM, Volume 51 Issue 9, pp: 72-79. ACM 2008.

[Boussaid et al, 2003] Boussaid, O.; Bentayeb, F.; Duffoux, A.; Clerc, F. "Complex Data Integration Based on a Multi-agent System". Springer Berlin / Heidelberg 2003, pp: 201-212, ISBN: 978-3-540-40751-5

[Chawathe et al, 1994] Chawathe, S., Garcia-Molina, H., Hammer, J., Ireland, K, Papakonstantinou, Y., Ullman, J. & Widom, J. The TSIMMIS Project: Integration of Heterogeneous Information Sources. In Proceedings IPSJ, Tokio, Japan. 1994, pp 7-18

[Cios et al, 2007] Cios, K.J.; Pedrycz, W.; Swiniarski, R.W.; Kurgan, L.A. "Data Mining A Knowledge Discovery Approach". Springer Science+Business Media 2007, LLC, pp: 606, ISBN: 978-0-387-33333-5

[Codd, 1970] Codd, E. "A relational model of data for large shared data banks". Communications of the ACM. Volume 13 , Issue 6. Pag: 377 – 387. ISSN:0001-0782.

[Connolly et al, 1996] T. Connolly, C. Begg, A. Strachan. Database Systems: A Practical Approach to Design, Implementation and Management. Addison-Wesley. ISBN 10: 0201342871 / 0-201-34287-1. ISBN 13: 978020134287. 1996

[Di Fatta & Fortino, 2007] Di Fatta, G. & Fortino G." A Customizable Multi-Agent System for Distributed Data Mining". In Proceedings of the 2007 ACM symposium on Applied computing. pp: 42 - 47. ISBN:1-59593-480-4

[Fayyad et al, 1996] Fayyad, U. M.; Piatestsky-Shapiro, G.; Smith, P. "From Data Mining to Knowledge Discovery: An Overview". Advances in Knowledge Discovery and Data Mining, pp:1-34, AAAI/MIT Press 1996

[FIPA, 2000] Foundation for Intelligent Physical Agents, FIPA Abstract Architecture Specification. 2000. http://www.fipa.org/specs/fipa00001/

[Franklin et al, 1996] Franklin, S., Graesser, A. “Is it an Agent, or just a Program?: A Taxonomy for Autonomous Agents”. In: Proceedings of the Third International Workshop on Agent Theories, Architectures, and Languages. Springer-Verlag (1996).

[Goasdoué et al, 2000] Goasdoué, F.; Lattes, V.; Rousset, M. “The use of CARIN language and algorithms for information integration: the PICSEL system”. Int. Jour. of Cooperative Information Systems (IJCIS) 9 (2000). pp 383–401

[Goasdoué & Rousset, 2004] Goasdoué, F.; Rousset, M. “Information Integration using Mediation”. Plein Sud Spécial Recherche, Université Paris-Sud XI, 2004, Nº 13, pp: 1-12.

[Haas, 2007] Haas, L.M. “Beauty and the beast: The theory and practice of information integration”. In International Conference on Database Theory (2007), 28–43.

[Han & Kamber, 2006] Han, J.; Kamber. “Data Mining Concepts and Techniques, Second Edition”. The Morgan Kaufmann Series in Data Management Systems, Jim Gray, Series Editor
Morgan Kaufmann Publishers, March 2006. ISBN 1-55860-901-6.

[Hernández et al. 2004] Hernández, J.; Ramírez, M.J.; Ferri, C. “Introducción a la Minería de Datos”. Prentice Hall - Pearson Education (Madrid, España) 2004. ISBN: 84-205-4091-9

[Iglesias, 1998]. Iglesias, C. “Definición de una Metodología para el Desarrollo de Sistemas Multi-Agentes”. Universidad Politécnica de Madrid. Tesis Doctoral. 1998

[Imtiaz et al, 2005] Imtiaz, S.; Hussain, A. “Using Agents for Unification of Information
Extraction and Data Mining”. Information and Communication Technologies, 2005. ICICT 2005. First International Conference. pp: 197-200. ISBN: 0-7803-9421-6

[Inmon, 2005] Inmon, W.; Building the Data Warehouse, 4th Edition. John Wiley & Sons. 2005. ISBN: 978-0-7645-9944-6

[Jacobson et al, 2000] Jacobson I, Booch G & Rumbaugh J. “El proceso unificado de desarrollo de Software”. Editorial Addison Wesley, 2000.

[Jennings et al, 1998] Jennings, N.R., Sycara, K., Wooldridge, M. “A Roadmap of Agent Research and Development”. Autonomous Agents and Multi-Agent Systems Journal, Kluwer Academic Publishers, Boston, 1998, Volume 1, Issue 1, pages 7-38

[Kargupta et al, 1997] Kargupta, H., Hamzaoglu, I. & Stafford, B. “Scalable Distributed data mining using an agent based architecture” .In Proceedings of KDD97[C]. Menlo Park , CA:AAAI Press,1997. pp: 211-214.

[Kimball et al, 1998] Kimball, R., Reeves, L., Ross, M. & Thornthwaite, W. The Data Warehouse Lifecycle Toolkit: Expert Methods for Designing, Developing, and Deploying Data Warehouses. Wiley. ISBN: 978-0-471-25547-5. 1998

[Latiesa, 1996] Latiesa, M. “Tipología y causas de la deserción universitaria y el retraso en los estudios”. Revista Diálogo Iberoamericano. 1996.

[Lim et al, 1993]Lim, E., Srivastava, J., Prabhakar, S., & Richardson, J. “Entity identification in database integration”. In: International Conference on Data Engineering, IEEE Computer Society Press, Los Alamitos, CA, USA, 1993, pp. 294–301.

[Luján-Mora et al, 2001] Luján-Mora, S. & Palomar, M. Reducing Inconsistency in integrating data from different sources, in: Proceedings of the International Database Engineering and Applications Symposium (IDEAS 2001), IEEE Computer Society, Grenoble, France, 2001, pp. 219–228.

[Oracle, 2008] Strohm R. (Autor Principal). “Bussines Intelligence” In. Oracle® Database Concepts 11g Release 1 (11.1). pp 321-339. 2008.

[Roddick & Vries, 2006] Roddick, J.F. & Vries, D. “Reduce, reuse, recycle: Practical approaches to schema integration, evolution, and versioning”. Advances in Conceptual Modeling. Theory and Practice, Lecture Notes in Computer Science, 4231. Springer, 2006.

[Rousset, 2002] Rousset, M. “Knowledge representation for information integration”. In: XIIIth

Int. Symp. on Methodologies for Intelligent Systems (ISMIS 2002), Lyon, France. Volume 2366 of LNAI., Springer Verlag (2002) 1–3.

[Rousset, 2004] Rousset, M. “Small Can Be Beautiful in the Semantic Web”. In: Proceedings of International Semantic Web Conference (ISWC 2004), pages 6--16, 2004.

[Simitsis et al, 2004] Simitsis, A.; Vassiliadis, P.; Sellis, T. “Optimizing ETL Processes in Data Warehouses”. Data Engineering, 2005. ICDE 2005. Proceedings. 21st International Conference on Volume , Issue , 5-8 April 2005 Page(s): 564 – 575

[Simitsis et al, 2004] Simitsis, A.; Vassiliadis, P.; Sellis, T. “Optimizing ETL Processes in Data Warehouses”. Data Engineering, 2005. ICDE 2005. Proceedings. 21st International Conference on Volume , Issue , 5-8 April 2005 Page(s): 564 - 575

[Schallehn et al, 2004] Schallehn, E.; Sattler, K. & Saake G. “Efficient similarity-based operations for data integration”. Data & Knowledge Engineering. Volume 48 , Issue 3 (March 2004) . Pages: 361 – 387. ISSN:0169-023X. 2004

[Squire, 1995] Squire, C. “Data Extraction and Transformation for the Data Warehouse”. ACM SIGMOD Record ACM SIGMOD Record (1995), Pages: 446 - 447. ISSN:0163-5808

[Two Crows, 1999] Two Crows Corporation. “Introduction to Data Mining and Knowledge Discovery”, 1999, Third Edition, ISBN: 1892095025, 36 Pages.

[Vassiliadis et al, 2002] Vassiliadis, P., Simitsis, A. & Skiadopoulos, S. “Conceptual Modeling for ETL Processes”. Proceedings of the 5th ACM international workshop on Data Warehousing and OLAP. McLean, Virginia, USA (2002), pages: 14-21. ISBN:1-58113-590-4.

[Viana et al, 2005] Viana, N., Raminhos, R. & Moura-Pires J. “A Real Time Data Extraction, Transformation and Loading Solution for Semi-structured Text Files”. Chapter 6 - Extracting Knowledge from Databases and Warehouses (EKDB and W 2005). pp: 383-394. ISBN: 978-3-540-30737-2

[Xing et al 2003] Xing, Y., Madden, M., Duggan, J. & Lyons G. “A Multi-Agent System for Context-based Distributed Data Mining”. Departament of Information Technology, National University of Ireland, Galway. Technical Report, NUIG-IT-170503

[Zhang & Zhang, 2003] Zhan, K., Zhan, C. “Data preparation for data mining”. Applied Artificial Intelligence, pp:375–381, Vol. 17, 5-6: 375-382. Taylor & Francis 2003.

[Zhou et al, 1996] Zhou G., Hull, R. & King, R. “Generating Data Integration Mediators that Use Materialization”. Journal of Intelligent Information Systems, Volume 6, Numbers 2-3, June 1996, pp. 199-221(23).

ANEXO A

DETALLES DE IMPLEMENTACIÓN DEL PROTOTIPO

Para la construcción del prototipo sobre el cual se validó este trabajo de investigación se emplearon diversas herramientas para el trabajo con agentes de software. En este anexo se presenta una breve descripción, sin entrar en muchos detalles técnicos, de cada una de estas y de su funcionalidad: JADE para la creación de agentes de software y su administración, SGBD para proporcionar un entorno que sea a la vez conveniente y eficiente para ser utilizado al extraer y almacenar datos; y por último el lenguaje JAVA que aparte de ser el leguaje en el cual esta soportado JADE, brinda interoperabilidad ente sistemas operativos, dándole portabilidad al prototipo.

A.1 Plataforma JADE

JADE (Java Agent DEvalopment Framework) es un entorno de software para construir sistemas multi-agente para la gestión de recursos de información en red. El objetivo de JADE es simplificar el tiempo de desarrollo de agentes, garantizando el cumplimiento de estándares a través de un amplio conjunto de servicios del y agentes del sistema [Bellifime, 2001]. El entorno de desarrollo está formado por una serie de librerías que permiten la implementación de agentes, mientras que el entorno de ejecución brinda la capacidad de ejecución brinda la capacidad de ejecución y comunicación de los mismos.

JADE cumple con las especificaciones FIPA (Foundation for Intelligent Physical Agents) para la interoperabilidad de plataformas de sistemas multi-agente, y cumple a dos niveles: a nivel de arquitectura y a nivel de mensajes. En el nivel de arquitectura, este estándar especifica en [FIPA, 2000] que una plataforma de agentes debe estar compuesta por:

- Un AMS (Agent Management System): Este agente proporciona el servicio de nombres asegurando que cada agente en la plataforma disponga de un nombre único. También representa la autoridad, es posible crear y matar agentes en contenedores remotos requeriendoselo al agente AMS.

- Un DF (Directory Facilitator): Proporciona el servicio de Páginas Amarillas. Gracias al agente DF, un agente puede encontrar otros agentes que provean los servicios necesarios para lograr sus objetivos.

- Un ACC (Agent Comunication Channel): Agente que proporciona la ruta para el contacto básico entre agentes dentro y fuera de la plataforma. Por medio de este agente se proporciona un método de comunicación confiable, ordenado y exacto. Esto se logra gracias a que este agente soporta RMI para la comunicación dentro de la misma plataforma e IIOP para interoperabilidad entre agentes en diferentes plataformas.

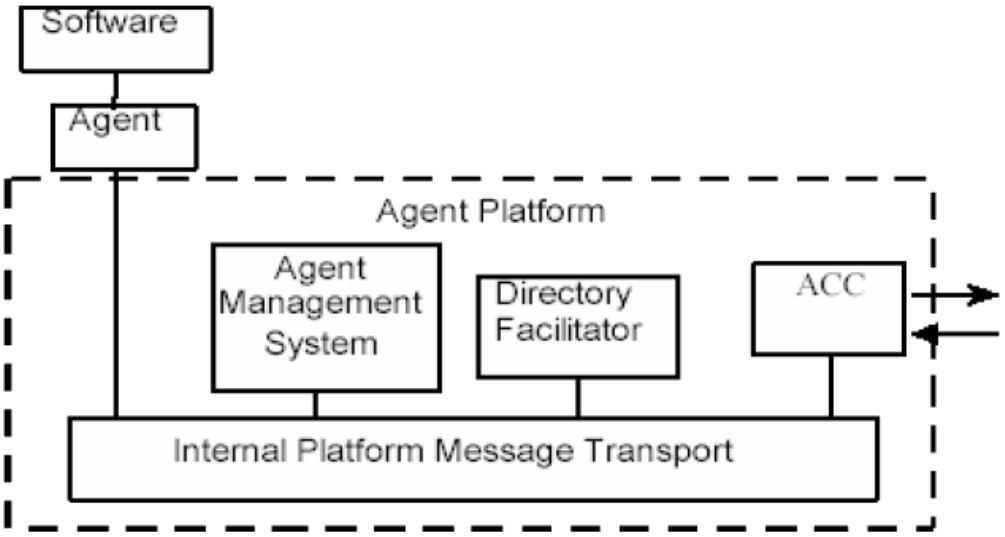


Figura A.1 Modelo de referencia FIPA para una plataforma de agentes extraída de [FIPA, 2000]

En cuanto al nivel de mensajes, FIPA propone sustituir el lenguaje KQML (Knowledge Query and Manipulation Language) tradicionalmente usado, por un nuevo ACL (Agente Comunication Language) denominado FIPA ACL. Para el transporte de este tipo de mensajes JADE proporciona un mecanismo de transporte de manera que el programador únicamente se preocupa por implementar la clase ACLMessage el cual contiene los métodos para llenar cada uno de los parámetros del mensaje.

Otra característica de JADE es que permite tener agentes distribuidos en diferentes máquinas o host. Esto permite que en cada host se encuentre la misma plataforma pero en diferentes contenedores que alojan diferentes instancias de agentes con una sola máquina virtual corriendo por host. Por otro lado, cada agente es

implementado como un hilo de JAVA lo cual permite que cada instancia pueda correr independientemente con sus propios comportamientos.

En la figura A.2 que corresponde a la interfaz provista por el agente RMA, se puede observar la forma en que JADE distribuye los agentes en la plataforma, así como los agentes AMS, DF y el ACC, los cuales se activan automáticamente cuando la plataforma es lanzada. Desde la interfaz también se pueden lanzar los agentes internos de JADE para la depuración del sistema (Sniffer, Introspector, Dummy) que proporcionas mecanismos para la monitorización y control de la plataforma y de los agentes.

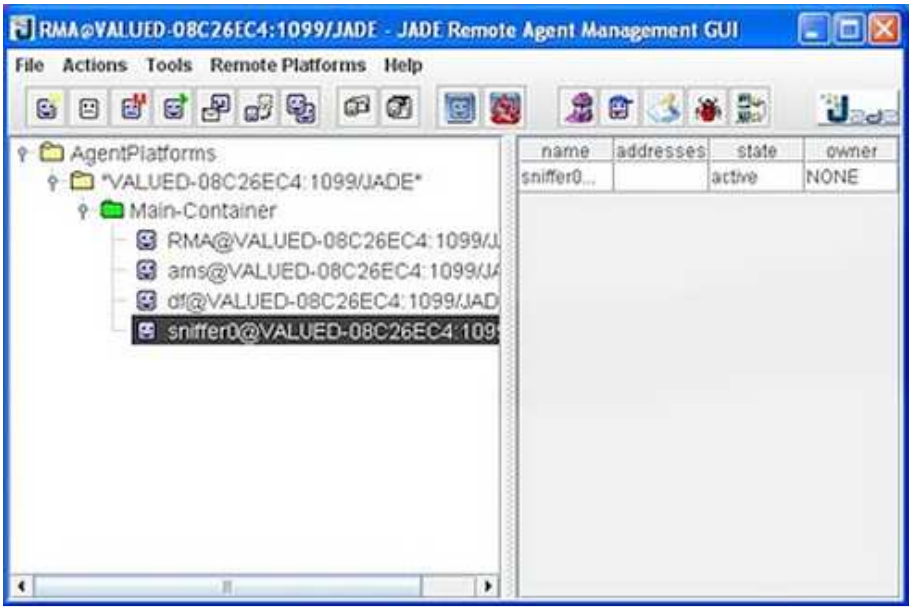


Figura A.2 Interfaz de JADE

A.2 SGBD

Los Sistemas Gestores de Bases de Datos (SGBD) son un tipo de software dedicado a servir de intermediario entre el usuario y los datos. Surgen desde mediados de los años sesenta y toman fuerzan a partir en los años setenta con la presentación del modelo relacional propuesto por Edgar Codd en [Codd, 1970]. Los SGBD tienen como principal objetivo permitir la creación y administración de bases de datos.

Algunos de los tópicos más representativos en los SGBD son:

- Independencia. La independencia de los datos consiste en la capacidad de modificar el esquema (físico o lógico) de una base de datos sin tener que realizar cambios en las aplicaciones que se sirven de ella.

- Seguridad. La información almacenada en una base de datos puede llegar a tener un gran valor. Los SGBD deben garantizar que esta información se encuentra segura de permisos a usuarios y grupos de usuarios, que permiten otorgar diversas categorías de permisos.

- Eficiencia. Lógicamente, es deseable minimizar el tiempo que el SGBD tarda en dar la información que se solicitada y en almacenar los cambios realizados.

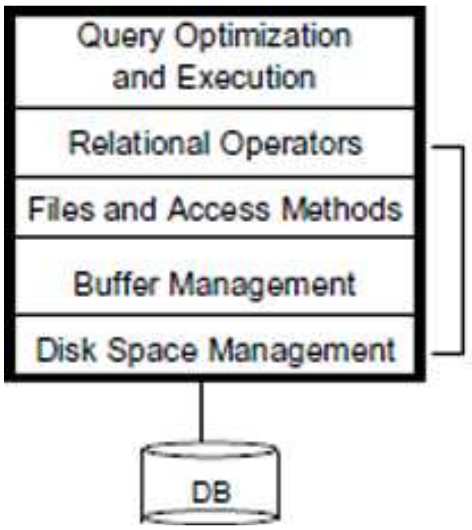


Figura A.3 Estructura de un SGBD

A.3 Arquitectura técnica del prototipo

La arquitectura empleada para la implementación de un prototipo consecuencia de esta investigación es coherente con los lineamientos de la fase de diseño presentados en el capítulo 5, según el cual, cada agente es una instancia de la clase a la que pertenece, y todos están contenidos dentro del entorno de JADE. El esqueleto de los agentes desarrollados se codificó empleando las librerías provistas por JADE, mientras que la codificación de los comportamientos específicos se realizó en JAVA mediante la plataforma NetBeans. La interfaz del prototipo se implementó mediante las librerías grafica disponibles en NetBeans y una imagen real de dicha interfaz es presentada en la figura A.4.

Figura A.4 Resultados del Sistema

El prototipo se diseñó para trabajar sobre una red LAN con tres servidores y se encuentra distribuido de la siguiente forma:

Un servidor 1 en que se tiene el SGBD MySQL con la base de datos de sistema Moodle , los archivos de la fuente Solicitudes Facultad y un contenedor secundario en donde se encuentra alojado un agente Recolector.

Un servidor 2 donde se tiene el SGBD PostgreSQL con la base de datos del sistema de información académica y toda la capa de vista del sistema, presentando las diferentes interfaces al usuario analista. También en este servidor se instancia el contenedor principal, en el cual se ejecutaran los agentes provistos por JADE, como son: AMS, DF, y RMA, además de algunos agentes del sistema, como son: Analista, Coordinador.

Y por último, el servidor 3 donde se tiene el SGBD Oracle con la base temporal y la bodega datos y un contenedor secundario en el cual se alojan los agentes integrador y almacenista.