Centro de Investigación y de Estudios
Avanzados
del Instituto Politecnico Nacional

Unidad Zacatenco
Departamento de Computación

Marco de trabajo basado en ontologías para el proceso ETL

Tesis que presenta

Joel Villanueva Chávez

Para obtener el Grado de

Maestro en Ciencias de la Computación

Directora: **Dra. Xiaoou Li Zhang**

México, D.F. **Noviembre de 2011**

Agradecimientos

Resumen

En sus inicios, los sistemas de información capturaban y almacenaban información sin un propósito especifico bajo diversos medios como: archivos de texto, binarios o XML y Bases de datos entre otros. Esto propicio la aparición los sistemas OLTP (Procesamiento de Transacciones en Línea), los cuales están orientados al uso de transacciones de inserción, modificación, y recuperación rápida de información.

Recientemente los investigadores propusieron enfoques para analizar y extraer conocimiento e información de los datos almacenados por sistemas OLTP, dando origen a los sistemas OLAP (procesamiento analítico en línea), estos están orientados al análisis de grandes cantidades de datos contenidos en un Data warehouse (almacén de datos o DWH).

La construcción de un data warehouse se realiza siguiendo el proceso ETL (Extracción Transformación y Carga). El cual comienza con la extracción de información de los sistemas OLTP, después esta es transformarla y finalmente depositada en el almacén de datos.

El problema central del proceso ETL es la interoperabilidad provocada por la diversidad léxico-sintáctica de las fuentes de información. Los enfoques actuales hacen la integración hasta un nivel léxico dejando el semántico (el mas complejo) en manos de las personas. Este tipo de problemas eran difíciles de afrontar, pero hoy en día se cuenta con herramientas como las ontologías con las cuales es posible combatir la interoperabilidad a un nivel semántico.

En este trabajo de tesis presentamos un marco de trabajo basado en ontologías para mitigar la interoperabilidad de el proceso ETL. Proponemos una metodología para capturar reglas de negocio con ayuda de las ontologías y gestión de conocimiento; modelar el contenido y estructura del almacén de datos y realizar el proceso ETL basado en el uso del conocimiento de las ontologías para lograr la integración de información a nivel semántico.

Abstract

In the beginnings, the information systems were used to capture and store information without a specific propose and under different media such as: data, binary and XML files and databases. These systems have evolved and lead to OLTP (On-Line Transaction Processing) systems, which are oriented to insert, modify and retrieve information transactions.

In recent time the researches have proposed different approaches in order to analyze and extract knowledge and information from the OLTP systems stored data. It produced the OLAP (On-Line Analytical Processing) systems origins. They are oriented to analyze huge amount of information in a Data warehouse (DWH).

The ETL process (Extraction Transformation and Loading) is followed to build a data warehouse. It begins with the Extraction of information from the OLTP systems, then it is transformed and finally it is loaded in the data warehouse.

The main problem of ETL process is the interoperability due to the lexical-syntactic diversity from the data sources. Current approaches make the integration up to lexical level and leaves the semantic level (the hardest) in the user´s duty. These problems used to be hard to face however today we have tools like ontologies, they allow to deal with the interoperability even to a semantic level.

In this thesis we present a framework ontology-based to mitigate the interoperability problems in ETL process. We propose a methodology to capture the business rules through ontologies and knowledge management, modeling the content and structure of data ware house and making the ETL process based in the use of ontologies' knowledge to get the information integration to a semantic level.

Índice general

Índice de figuras

Índice de cuadros

Capítulo 1

Introducción

Por mucho tiempo la información almacenada por sistemas OLTP (Procesamiento de transacciones en linea por sus siglas en ingles) servía únicamente para elaborar reportes a veces complejos y difíciles de entender. Con el devenir de la minería de datos, surgieron enfoques para analizar información histórica almacenada y extraer conocimiento o información no trivial que fuera útil a los propietarios de dicha información; lo anterior dio pie a los sistemas OLAP (Procesamiento Analítico en Linea), sistemas pensados para el análisis de grandes volúmenes de datos.

Los sistemas OLAP trajeron consigo un nuevo concepto, el Data Warehouse. Un Data Warehouse es un repositorio que guarda los datos históricos de una organización y se construye a partir de la integración de las diversas fuentes de información existentes[1].

La construcción y almacenamiento de información de un Data Warehouse se hace a través del proceso denominado ETL (extracción, transformación y carga). El proceso ETL tiene como función, extraer los datos de los diferentes sistemas OLTP de una organización, transformarlos y unificarlos bajo un esquema y estructura, para finalmente depositarlos dentro del Data Warehouse.

Las fuentes de información que intervienen en el proceso ETL difieren unos de otros en algunos elementos como: mecanismos de acceso, estructura y esquemas de datos; algunos de ellos pueden ser archivos XML, archivos CSV, archivos de texto plano o bases de datos relacionales. Es por ello que el proceso ETL tiene que enfrentar problemas de interoperabilidad a diversos niveles: técnico, sintáctico y semántico. Para el tratamiento de los dos primeros hay líneas de investigación y enfoques ya consolidados [2, 3, 4], pero hasta hace poco no se contaba con herramientas para poder mitigar el último.

De los enfoques actuales para abordar la interoperabilidad a nivel semántico el más ampliamente usado es la aplicación de las ontologías. Las ontologías ayudan a definir los elementos, relaciones y reglas de un dominio de conocimiento específico y favorecen el intercambio y compresión de datos e información entre sistemas heterogéneos.

En general, las ontologías han sido principalmente empleadas en problemas relacionados con la web semántica, pero analizando la similitud de la problemática de la web semántica y el proceso ETL, es posible proponer enfoques basados en ontologías

para ayudar al proceso ETL.

Las ontologías almacenan conocimiento un dominio de conocimiento, es por ello que es posible capturar el conocimiento de expertos en un dominio específico dentro de una ontología para posteriormente ser utilizado para resolver problemas concretos. En este caso se observa que la información contenida dentro de la ontología será la pieza clave para la integración de las diversas fuentes de información dentro del proceso ETL.

Para precisar más el contexto de investigación en el que estará inmerso este trabajo, presentamos las áreas en las que esta involucrado este trabajo de investigación según la clasificación de la ACM [5]:

- H. Information Systems
  - H.2 DATABASE MANAGEMENT
    - H.2.8 Database Applications
      - Data mining
  - H.3 INFORMATION STORAGE AND RETRIEVAL
    - H.3.2 Information Storage
      - Record classification
- I. Computing Methodologies
  - I.2 ARTIFICIAL INTELLIGENCE
    - I.2.5 Programming Languages and Software
      - Expert system tools and techniques
  - I.2.6 Learning
    - Knowledge acquisition

1.1. Antecedentes generales

El proceso ETL fue propuesto inicialmente como un proceso altamente dependiente de la intervención de los administradores de los almacenes de datos, comprendía tareas engorrosas como lo son: análisis de los esquemas de datos, programación de rutinas extracción de cada uno de ellos, análisis de las reglas y transformaciones para cada una de las diferentes fuentes, por decir algunos.

Los primeras implementaciones ETL datan de finales de la década de los 90, era bastante rudimentarios y precarios [6]. La información recibía un tratamiento a nivel léxico, sintáctico. El tratamiento semántico, es decir las reglas de negocio y propias del área que servían para ejecutar los cambios y transformaciones eran definidas en conjunción con los expertos de dominio, esto implicaba pérdida de tiempo al momento de coordinar actividades [7].

Debido a la similitud de la problemática que enfrenta la web semántica y el proceso ETL, se han propuesto algunos enfoques basados en ontologías [8, 9]. De manera errónea algunos de estos enfoques han dividido el diseño conceptual del proceso ETL de las reglas de negocio que gobiernan los datos involucrados en el mismo, como pasa en [9, 10]. Es por esto que es determinante considerar el conocimiento que ayudará a definir o a asistir un proceso ETL si se opta por enfoques basados en ontologías.

La captura del conocimiento de los expertos dentro de una organización, para ayudar a la mejora y resolución de problemas existentes dentro de la misma, es parte de una disciplina denominada gestión del conocimiento (knowledge management) [11]. Se han propuesto enfoques, para poder realizar la captura de dicho conocimiento por medio de ontologías.

1.2. Motivación

Las herramientas y enfoques actuales para llevar a cabo el proceso ETL tienen algunas desventajas como son: alta dependencia de la intervención humana debido la integración de la información solo hasta un nivel léxico sintáctico [1], otra desventaja es la dificultad de instalar, configurar e implementar dichas herramientas [12].

Al analizar enfoques actuales que abordan el proceso ETL tales como [3], se sigue observando que el problema central es la interoperabilidad. La interoperabilidad se define según la IEEE como «la capacidad de dos o mas sistemas o componentes para intercámbiar información y usarla una vez que esta ha sido intercambiada» [13]. Y podemos encontrar diversos niveles de interoperabilidad como:

Interoperabilidad a nivel técnico: permite a los sistemas de información el intercambio de señales a través de una conexión física y un conjunto de protocolos de comunicaciones (como TXP/IP).

Interoperabilidad a nivel sintáctico: posibilita a los sistemas de información leer datos entre si y obtener una representación sobre la cual pueden operar.

Interoperabilidad a nivel semántico: permite a los sistemas de información el intercambio de información basado en un significado común de los términos y relaciones que ésta usa.

Los niveles de interoperabilidad se esquematizan en la figura 1.1:

De los enfoques actuales con los que la web semántica ha atacado los problemas de interoperabilidad, las ontologías parecen brindar una buena alternativa para ayudar al problema en todos y cada uno de sus niveles, prueba de ello son la creación de herramientas como buscadores, integradores de servicios web, etc. Es por ello que ha surgido la idea de la aplicación de las ontologías para combatir la problemática de interoperabilidad en el proceso ETL, e ir mas allá de la integración de información a niveles técnico-sintácticos.

Pero aún en el caso de algunos enfoques que consideran realizar el proceso ETL a niveles semánticos con ayuda de ontologías, se observa que la base de conocimiento que éstos emplean por lo general no es definida a partir del conocimiento de personas especializadas en el dominio de conocimientos, sino por personas especializadas

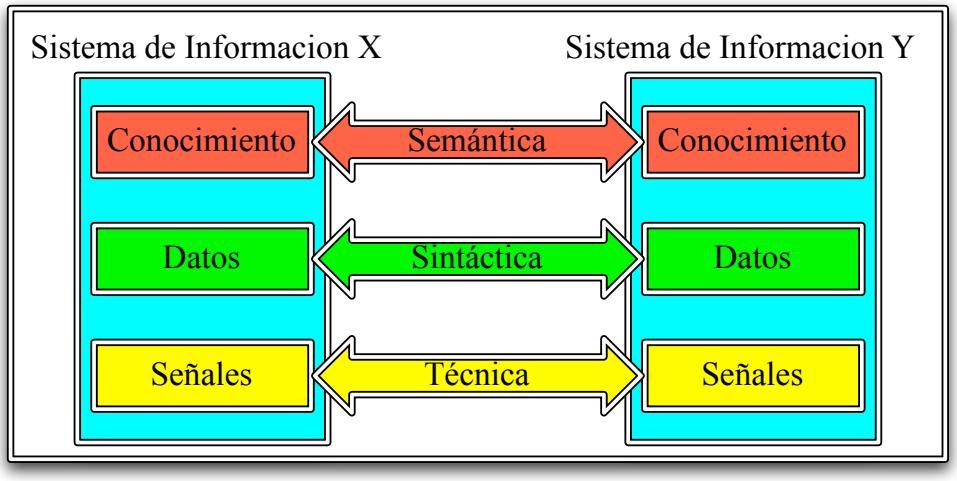

Figura 1.1: Los niveles de Interoperabilidad

en la definición de ontologías. Esto pudiese parecer incongruente pero, si lo que se está definiendo es un dominio de conocimiento, como lo sostiene [14], no existe mejor conocimiento que el obtenido de una persona especializada en el dominio de conocimiento que se está definiendo. Solo de esta forma los elementos, relaciones y reglas del dominio pueden ser definidas de manera adecuada.

La disciplina especializada en la captura del conocimiento de los expertos dentro de una organización es denomina como gestión del conocimiento. A pesar de ser un enfoque bastante atractivo, actualmente existen muy pocas metodologías y herramientas para poder llevar a cabo la captura y gestión del conocimiento, entre ellas encontramos: manuales de políticas y procedimientos, metodologías, etc. y en tiempos recientes ontologías. No obstante se han propuesto esquemas basados en ontologías como [15], para poder realizar una correcta gestión del conocimiento, es decir la correcta captura, gestión y administración del conocimiento de una organización.

Si encontramos un número reducido de enfoques para realizar la gestión del conocimiento con ayuda de ontologías [16, 11, 15], mucho menos es el número de éstos que plantean la aplicación de este conocimiento para resolver problemas concretos dentro de la organización. Si en algún momento se aprovechase el potencial de la gestión conocimiento, las organizaciones podrían obtener de forma más sencilla información a partir de sus datos y conocimiento a partir de la información.

Dado el panorama anterior, surge la motivación de este trabajo de tesis, dada la problemática de interoperabilidad existente dentro del proceso ETL y dado que las ontologías han demostrado ser elementos que combaten eficazmente este problema en todos sus niveles, resulta interesante poder plantear una solución que en principio defina adecuadamente un dominio de conocimiento con ayuda de la gestión del conocimiento, extrayendo el conocimiento de los expertos y almacenándolo en una ontología, para que después este conocimiento pueda ayudar a mitigar el problema de la interoperabilidad dentro del proceso ETL.

1.3. El proyecto de tesis

El proyecto define, a nivel conceptual y de diseño, un marco de referencia para poder llevar a cabo tres actividades centrales:

- La captura del conocimiento en ontologías con ayuda de la gestión del conocimiento.
- La administración de dicho conocimiento.
- La recuperación y uso del conocimiento para realizar el proceso ETL.

El proceso ETL basado en ontologías, considerando como potenciales fuentes de información: XML, bases de datos relaciona y archivos CSV. Nuestro proceso ETL trabajará con base en a la extracción de meta información de las fuentes de información. Posteriormente el proceso se auxiliará de las ontologías para la categorizar y etiquetar semánticamente la meta información. Ademas de proponer ajustes a niveles léxicos y semánticos para finalmente tener un esquema de datos unificado y así poder realizar los procesos de extracción y carga de forma autónoma.

La principal diferencia del trabajo de tesis con enfoques similares como [10], radica en que nuestro trabajo presenta la definición del conocimiento de la ontología por medio de un procedimiento especializado como lo es la gestión del conocimiento, además de proponer una arquitectura en capa para así atacar de forma dedicada los diversos niveles de interoperabilidad.

Para la realización de este trabajo de tesis se hace uso de diversas tecnologías. En el caso de la primera parte se analizaron y evaluaron algunas metodología para realizar la gestión del conocimiento como [17, 18, 19], posteriormente se estudiaron algunas propuestas para su implementación con ayuda de ontologías, de esta forma se realizó el modelado y diseño conceptual del marco de trabajo.

En lo referente a la implementación se consideraron herramientas que posibilitaran la construcción de ontologías bajo metodologías como las que se presentan en [19]; como ejemplos podemos citar algunas APIs (Interfaz de programación de aplicaciones) como JENA [20], JADE [21] .

La idea central del proceso ETL propuesto descansa sobre la manipulación de meta-datos, su categorización, etiquetado semántico, encapsulamiento, transformación y la concepción de estos como agentes centrales para la extracción de información. Para tal propósito se investigaron y evaluaron herramientas que permitieran extraer meta-datos como: Xerces[22] y XSom [23] para el trabajo con archivos XML; el API JDBC para el trabajo con bases de datos y java SE 1.5 y algunas de sus utilidades para el trabajo con archivos CSV. Para el etiquetado, encapsulación y transformación de meta-datos se propone un esquema conceptual, mapeos y matrices de transformación basados en objetos java. Estos interactúan con las ontologías por medio de APIs especializadas: protege [24] y JENA [20].

La arquitectura y procesos de extracción de los datos a nivel técnico y léxico están definidos a nivel conceptual sobre una jerarquía de objetos java complementadas con

algunas APIS ya mencionadas. El marco de trabajo encapsula la funcionalidad y tratamiento específico de los diversos tipo de fuentes de información de para que las tareas específicas sean transparentes al usuario.

1.4. Organización del documento

El presente documento de tesis presenta el marco de trabajo (ONTO-ETL), conformado tres elementos centrales.

En primer lugar dos metodologías, una para realizar la gestión del conocimiento basado en ontologías que modelan los elementos, relaciones centrales y reglas de negocio de un dominio específico; la segunda de ellas define la realización del proceso ETL con ayuda del conocimiento adquirido. El tercer elemento del marco de trabajo es el diseño conceptual de una biblioteca de componentes software, que asiste a las metodologías anteriormente descritas.

Para poder mostrar la factibilidad del diseño y metodología propuestas, se desarrolló una aplicación que utiliza componentes de la biblioteca elaborada, además de ello se implementó un caso de estudio teórico aplicando las metodología propuestas y la aplicación desarrollada.

El presente documento de tesis se estructura de la siguiente manera:

En el capítulo 2 se presentan los fundamentos de las ontologías y la gestión del conocimiento, se habla de la evolución histórica del concepto de ontología y sus fundamentos teóricos. Posteriormente se exponen las clasificaciones hechas sobre ontologías, para dar pie a la ingeniería ontológica, donde se abordan los elementos de las ontologías, las metodologías para la construcción de las mismas y finalmente se presentan los principales lenguajes ontológicos. La segunda parte de este capítulo aborda definiciones acerca del conocimiento, para después definir los componentes, orígenes y metodologías para la implementación de la gestión del conocimiento. Finalmente se listan algunos enfoques para la implantación del proceso basados en ontologías.

En el capítulo 3 se describe el proceso ETL, se da un contexto general de los almacenes de datos, su relación con los sistemas OLAP y el uso de los mismos para soluciones de inteligencia de negocio. Posteriormente se describen los fundamentos, elementos, técnicas, procesos, consideraciones y problemáticas actuales del proceso ETL, para finalmente dar una semblanza de los enfoques actuales para atacar el problema, desde un punto de vista ontológico, como también fuera del mismo.

En el capítulo 4 se expone el framework Onto-ETL, primeramente se introduce la arquitectura general del marco de trabajo y una descripción de los tres elementos centrales. Posteriormente se detalla la metodología propuesta para la gestión del conocimiento, la metodología para realizar el proceso ETL a través del uso de ontologías y finalmente se presenta el diseño conceptual y la descripción de los componentes de la biblioteca de software elaborada.

En el capítulo 5 se presenta el caso de estudio donde se evaluó la biblioteca de componentes, la descripción general del problema, las características de las fuentes de información que se consideraron y las aplicaciones que se desarrollaron sobre el

marco de trabajo para atacar el problema. Posteriormente se muestran las pruebas realizadas, los objetos, productos y resultados obtenidos.

En el capítulo 6 se presentan las conclusiones, limitaciones y el trabajo a futuro que se puede realizar sobre el trabajo de investigación realizado.

Capítulo 2

Las ontologías y la Gestión de Conocimiento

El concepto de ontología tiene un origen filosófico y fue adoptado hasta hace pocos años dentro del ámbito de la ciencias de la computación. Desde su adopción dentro del ámbito de la computación ha sufrido diversos cambios, dado que es un concepto que intenta definir, modelar y formalizar dominios de conocimiento reales. Dentro de la comunidad de inteligencia artificial han surgido diversas corrientes que han definido el termino de ontología. Por una parte las tendencia formales y apegadas a las concepciones originales del término pero a veces alejadas de la vida real. Por el contrario se han propuesto definiciones más flexibles y apegadas a la realidad pero sacrificando cierto grado de formalidad.

Puesto que las ontologías son entidades que permiten definir dominios de conocimiento de manera formal y «entendible» de un sistema computacional, han posibilitado la interoperabilidad entre sistemas heterogéneos. Esto posibilitó, en un principio la creación de aplicaciones como son: intercambio de información automática entre sistemas web (Web Semántica), buscadores semánticos, recuperación y clasificación de información y meta información. En los últimos años se ha propuesto la idea de realizar la captura del conocimiento organizacional por medio de ontologías.

La idea de contar con una biblioteca o memoria corporativa dentro de una organización, en la cual esté contenida el conocimiento, experiencia y capital intelectual de los miembros de una organización, con el objetivo de que este conocimiento ayude a solucionar problemas y agilizar el flujo de información y conocimiento dentro de la organización, recibió el nombre de Gestión de Conocimiento (knowledge management).

En años recientes se han propuesto enfoques que señalan cómo realizar la Gestión de Conocimiento por medio de ontologías [18]. Estos enfoques señalan las fases principales que involucra la Gestión de Conocimiento, tales como son: la captura del conocimiento, su administración, recuperación y uso concreto para la resolución de problemas concretos.

2.1. Definiciones

En el contexto filosófico

El término de ontología se ha ido modificando a lo largo de su historia. En esta sección abordaremos diversas definiciones del concepto desde dos principales perspectivas, una de ellas filosófica, como la que aborda Kant [25]:

"*La filosofía trascendental es el sistema de todas nuestras cogniciones puras a priori, que podemos llamar ontología. Así, ontología trata con cosas en general, desde abstractas hasta particulares. Abarca todos los conceptos puros de la comprensión y todos los principios de la razón. Las ciencias principales que pertenecen a la metafísica son: ontología, cosmología, y teología. Ontología es una pura doctrina de elemento de toda nuestra cognición al completo, o: contiene la suma de todos nuestros conceptos puros que podemos tener a priori sobre la cosa.*"

Y la otra relacionada con el campo de la inteligencia Artificial, la primera definición venida de este campo fue dada por Neches [26] y dice lo siguiente:

"*Una ontología define los términos básicos y relaciones que conforman el vocabulario de un área específica, así como las reglas para combinar dichos términos y las relaciones para definir extensiones de vocabularios.*"

En el contexto de la inteligencia artificial

El concepto de ontología comenzó a convertirse en un tema de interés para algunas comunidades de Inteligencia Artificial hacia comienzo de los años noventa, entre ellas podemos citar: Ingeniería del Conocimiento, Procesamiento del Lenguaje Natural o Representación del Conocimiento. Recientemente la noción de ontología ha tomado fuerza en áreas como integración inteligente de información, sistemas cooperativos de información, recuperación de información, comercio electrónico y Gestión de Conocimiento. Lo anterior dio origen a definiciones como la de Gruber [27]:

«Una ontología es una especificación explícita de una conceptualización. El término proviene de la filosofía, donde una ontología es un recuento sistemático de la existencia. En sistemas de Inteligencia Artificial, lo que existe es lo que puede ser representado. Cuando el conocimiento de un dominio se representa mediante un formalismo declarativo, el conjunto de objetos que puede ser representado se llama universo del discurso. Esos conjuntos de objetos, y las relaciones que se establecen entre ellos, son reflejados en un vocabulario con el cual representamos el conocimiento en un sistema basado en conocimiento. Así, en el contexto de Inteligencia Artificial, podemos describir la ontología de un programa como un conjunto de términos. En tal ontología, las definiciones asocian nombres de entidades del universo del discurso con textos comprensibles por los humanos que describen el significado de los nombres, y axiomas formales que limitan la interpretación y buen uso de dichos términos. Formalmente, una ontología es una teoría lógica».

Quizás el auge e interés que han tomado las ontologías se debe, en gran medida, a lo que estas prometen: un entendimiento compartido y común de un área o domi-

nio de conocimiento el cual posteriormente podrá ser comunicado entre individuos o aplicaciones.

2.2. Tipos de ontologías

Dentro de la literatura podemos encontrar diversos criterios para clasificar las ontologías, a continuación presentaremos los dos más ampliamente usados:

2.2.1. Clasificaciones por el tipo de conocimiento almacenado

- Con base al tipo de conocimiento que contienen: una clasificación con base a este criterio fue dada en [17], y consta de las siguientes categorías:

  - **Ontologías de dominio**: contienen todos los conceptos asociados a un dominio particular.
  - **Ontologías de tarea**: dictan la forma en la cual se puede usar el conocimiento de un dominio para poder realizar tareas específicas.
  - **Ontologías generales**: guardan descripciones generales acerca de objetos, eventos, relaciones de tiempo y causa, modelos de comportamiento y funcionalidades.

2.2.2. Clasificaciones por la motivación de la ontología

- Con base a la motivación de la ontología: para atender este criterio se propuso la siguiente clasificación:

  - **Ontología para la representación del conocimiento**: permiten explicar conceptualizaciones que tienen como base formalismos para la representación del conocimiento.
  - **Ontologías genéricas**: definen términos que son considerados generales en diversas áreas. Se llaman también ontologías abstractas por que permiten definir conceptos abstractos.
  - **Ontologías del dominio**: definen la conceptualización especifica de un dominio.
  - **Ontología de aplicación**: este tipo de ontologías están directamente relacionadas con el desarrollo de una aplicación concreta.

2.3. Ingeniería Ontológica

2.3.1. Elementos de las ontologías

Las ontologías nos proporcionan un vocabulario común acerca de un área y definen, de maneras diversas en diversos grados de formalismo, el significado de los términos y las relaciones entre los mismos. Para poder formalizar el conocimiento dentro de las ontologías, éstas se auxilia de cinco componentes básicos: clases, relaciones, atributos, funciones, axiomas e instancias [27].

- **Clases**: dentro de este ámbito, suele utilizarse el término clase o concepto de manera indistinta. Una clase puede ser algo sobre lo que se dice alguna cosa y, por lo tanto, ésta podría ser una tarea, una función, acción, estrategia, etc.
- **Relaciones**: estas nos representan algún tipo de interacción entre los conceptos del dominio.
- **Atributos**: son propiedades que pueden tener las clases y por tanto también sus individuos.
- **Funciones**: son un caso especial de relaciones en las cuales el elemento n de la relación es único para los n-1 elementos precedentes.
- **Axiomas**: son expresiones que son siempre ciertas. Se incluyen dentro de una ontología para diversos propósitos, como pueden ser, la definición de restricciones sobre valores de atributos o los argumentos de las relaciones.
- **Instancias**: se usan para representar elementos específicos del dominio.

2.3.2. Metodologías para la construcción de ontologías

En la literatura se encontraron diversas metodologías para la construcción de ontologías. Dichas metodologías se pueden clasificar por diversos criterios. Uno de ellos es con base a la infraestructura que se tiene para su elaboración. Bajo este parámetro encontramos metodologías para construir ontologías "partiendo de cero" y por otra parte tenemos aquellas que nos ofrecen hacer la construcción de ontologías a partir de proceso de reingeniería.

El estudio y análisis de las metodologías queda fuera del alcance de este documento, por lo que se mencionara algunas sin entrar en más detalle:

- Metodología Cyc [28].
- Metodología de Construcción de Ontologías de Uschold y King [16].
- Metodología de Construcción de Ontologías de Grüninger y Fox [29].
- Metodología KACTUS [30].

- Metodología SENSUS [31].
- Metodología On-To-Knowledge [15].
- TERMINAE [32].

2.4. Lenguajes Ontológicos

Un lenguaje ontológico es un medio para poder expresar las ontologías de forma que éstas puedan ser comprendidas por las máquinas. Una gran parte de ellos ha surgido a la par de la llamada Web Semántica. Sin embargo, algunos otros han sido propuestos por desarrolladores de herramientas ontológicas puras.

En este apartado presentamos algunos de los lenguajes ontológicos más empleados al día de hoy.

- **SHOE**: el denominado Simple HTML Ontology Extension [33], desarrollado en la Universidad de Maryland como una extensión de HTML con la inclusión de conocimiento semántico en documentos Web.

  Ofrece soporte para poder modelar Ontologías, clasificación de instancias, relaciones entre clases y aplicación de reglas.

- **RDF(s)**: por sus siglas en inglés Resource Description Framework [34], fue desarrollado por el W3C con la finalidad de poder especificar contenido semántico, estandarizado, interoperable y basado en XML. Basa su modelo de datos en tres representaciones distintas: como tripletas, grafos y en XML.

- **OML**: Ontology Markup Languaje [35], fue desarrollado en la Universidad de Washington y esta basado parcialmente en SHOE. Las ontologías se representan a través de un conjunto de entidades de todos los tipos.

- **XOL**: XML-based Ontology Exchange Language [36]; desarrollado por la comunidad bioinformática de EEUU con la finalidad de intercambiar información en sistemas heterogéneos.

- **OIL**: Ontology Interchange Languge [37], fue desarrollado como una parte del proyecto OntoKnowledge, su sintaxis está basada fuertemente en lenguajes como XOL y RDF. Trabaja las ontologías en tres capas diferentes:

  1. El nivel de objeto, donde se describen instancias concretas de la ontología
  2. El primer metanivel, donde se proporcionan las definiciones ontológicas actuales; y
  3. El segundo metanivel, relacionado con la descripción de características de la ontología como autor, nombre, etc.

2.5. La Gestión de Conocimiento

La Gestión de Conocimiento es una disciplina de reciente creación. En particular el término Gestión de Conocimiento fue usado por primera vez en una conferencia por Karl Wiig [11]. Desde entonces se han dado definiciones como la siguiente:

"La Gestión de Conocimiento es el proceso de capturar experiencia colectiva organizacional donde ésta resida (por ejemplo, bases de datos, documentos, mentes humanas) y su distribución allá donde pueda ayudar a mejorar los resultados"[38].

En las siguientes secciones ahondaremos más en este término y los conceptos que lo envuelven.

2.5.1. Conceptos importantes

El conocimiento

Como primer punto abordaremos el concepto de conocimiento. Tomado desde la perspectiva de la Gestión de Conocimiento, podemos definir al conocimiento como: "El conocimiento es información organizada y analizada para hacerla comprensible y aplicable a la resolución de problemas y toma de decisiones" [39].

Tipos de conocimiento.

Con la finalidad de poder categorizar los tipos de conocimiento existentes. Se deben considerar diversos factores, uno de ellos puede ser el medio donde sé almacena, como pueden ser la mente humana, documentos escritos, sistemas de representación de conocimiento, etc.

Otro factor a considerar puede ser la forma de acceder al conocimiento. Dados los dos puntos anteriores, en [40] se propusieron dos tipos fundamentales de conocimiento:

- **Conocimiento tácito**: este tipo de conocimiento es todo aquel adquirido a través de la experiencia, conocimiento simultáneo (relativo al aquí y ahora) y conocimiento análogo (asuntos prácticos).
- **Conocimiento explícito**: corresponde a todo aquel conocimiento racional (en la mente), conocimiento secuencial (relativo al ahí y al entonces), y conocimiento digital (aspectos teóricos).

Debemos mencionar que, dados estos dos tipos de conocimientos, hay formas de poder transformar, compartir y difundir entre los dos tipos. En particular dentro de una organización se busca la transformación del conocimiento tácito en implícito, es decir de formalizar todo el conocimiento ganado a través de la experiencia.

Sistemas de Gestión de Conocimiento

Como parte de la naturaleza misma del conocimiento, encontramos según [41] dos propiedades "no deseadas del mismo":

- Si no se explica se hace tácito.
- Si no se comunica y comparte se pierde.

Con la finalidad de atacar estas dos problemáticas, dentro del área de Inteligencia Artificial aparecieron dos áreas tecnológicas: la Ingeniería del Conocimiento y la Gestión de Conocimiento, ambas comparte una misma tecnología medular, la tecnología del conocimiento.

La Inteligencia Artificial al aplicar las tecnologías del conocimiento, pudo generar como resultados bases de conocimiento, que podían ser consumidas por sistemas expertos y basados en conocimiento. Aunque en los desarrollos más recientes de Gestión de Conocimiento, el conocimiento queda totalmente disponible para su asimilación por seres humanos.

Ahora bien, de acuerdo con [18], un sistema de Gestión de Conocimiento debe facilitar:

- La conversión de datos y texto en conocimiento.
- La conversión de conocimiento individual y de grupo en conocimiento accesible.
- La conexión de individuos y conocimiento a otros individuos y otro conocimiento.
- La comunicación de información entre diferentes grupos.
- La creación de nuevo conocimiento útil para la organización.

Desde esta perspectiva, el presente trabajo de tesis contempla algunos de los puntos señalados anteriormente. Por tal motivo, podemos afirmar que se estará trabajando con aspectos de la Gestión de Conocimiento.

Capítulo 3

Data Warehouse y el proceso ETL

El concepto de Data Warehouse fue propuesto inicialmente por Codd en [6], dicho término cambiaba de forma radical el concepto del almacenamiento de la información, ya que proponía la integración y unificación de las fuentes de información heterogéneas surgidas de los sistemas transnacionales dentro de una organización, para así proporcionar información histórica, consolidada y fiable a soluciones analíticas mas complejas.

En este capítulo exploraremos el contexto general en el cual están inmersos los almacenes de datos, su importancia y el rol que desempeñan como elementos centrales de soluciones más complejas como los sistemas de soporte a la toma de decisiones EIS (Sistemas de Información Ejecutiva) y soluciones BI (Business intelligence).

Dada la importancia de los almacenes de datos, es de vital importancia la correcta construcción y diseño de los mismos. Es por ello que hay que garantizar que los mecanismos y procesos para la elaboración y construcción de un Data Warehouse, en concreto el proceso ETL se realice con base a las mejores prácticas. Por tal razón analizaremos los elementos, tareas, problemáticas y técnicas que sigue el actual proceso ETL.

3.1. El Data Warehouse

El objetivo de un Data Warehouse es ofrecer los datos de una organización para una mejor toma de decisiones. El éxito de un Data Warehouse radica en entregar la información de la mejor manera a los usuarios finales.

La construcción de un Data Warehouse involucra el proceso de tomar datos de sistemas legados y transaccionales y transformarlos en información organizada en un formato amigable a los usuarios para facilitar el análisis y el soporte a la toma de decisiones basado en hechos.

Según kimbal un Data Warehouse se puede definir de la siguiente manera [4]:

«*Un Data Warehouse es un sistema que extrae, limpia, ajusta y entrega las fuentes de información y solo entonces soporta e implementa herramientas de consulta y análisis con el propósito de tener una correcta toma de decisiones.*»

A menudo suele confundirse el término de Data Warehouse, definiéndolo erróneamente como:

- Un producto: un Data Warehouse involucra muchísimos elementos como para ser considerado un producto como son: análisis del sistema, manipulaciones de datos, movimiento de datos y el modelado dimensional y el acceso a datos.

- Un lenguaje: dada la diversidad de los componentes de un Data Warehouse, se requiere el dominio de uno o más lenguajes de programación y de la comprensión de múltiples especificaciones de datos.

- Un proyecto: el correcto desarrollo de un Data Warehouse involucra muchos proyectos y diversas fases en cada uno de ellos, esto debido al grado de especialización de las tareas que se tiene que realizar.

- Un modelo de datos: un modelo de datos por si mismo no hace al Data Warehouse. Sin datos, ni el mejor modelo de datos es funcional.

- Una copia de un sistema transaccional: un Data Warehouse tiene un modelo de datos propios diferente al de los sistemas transaccionales de los cuales se alimenta, puesto que debe de responder a otro tipo de problemas.

3.1.1. Desarrollo histórico de los almacenes de datos

La problemática principal de los almacenes de datos surgió desde el momento en que múltiples medios de almacenamiento estuvieron a disposición dentro de las organizaciones. Bajo los primeros esquemas de almacenamiento como las cintas magnéticas, se tenían problemas como la sincronización de archivos y la complejidad de mantener aplicaciones desarrolladas y de desarrollar nuevas.

Al surgir los actuales medios de almacenamiento como los dispositivos de almacenamiento de acceso directo, se dio pie al desarrollo de los sistemas gestores de bases de datos. Esto dio lugar al paradigma «una sola fuente de datos para todos tipo de procesamiento (transaccional y analíticos)».

Posterior al desarrollo de los sistemas gestores de bases de datos, se desarrollaron los lenguajes 4G, éstos permitieron a los usuarios finales el control directo de los datos y los sistemas. Esto provoco que el paradigma de «la base de datos única» se viniera abajo, puesto que una sola base de datos no servía para el procesamiento de operaciones transaccionales y el procesamiento analítico al mismo tiempo.

Con la caída del paradigma de «la base de datos única», se dio la división de los sistemas de información en OLTP y OLAP, el primero especializado en procesamiento de transacciones y el segundo en procesamiento analítico.

La evolución de los sistemas de almacenamiento se puede ver en la figura 3.1:

A lo largo de la evolución de los sistemas de almacenamiento, un sin fin de fuentes de datos fueron desarrolladas sobre los esquemas de datos iniciales y posterior a la

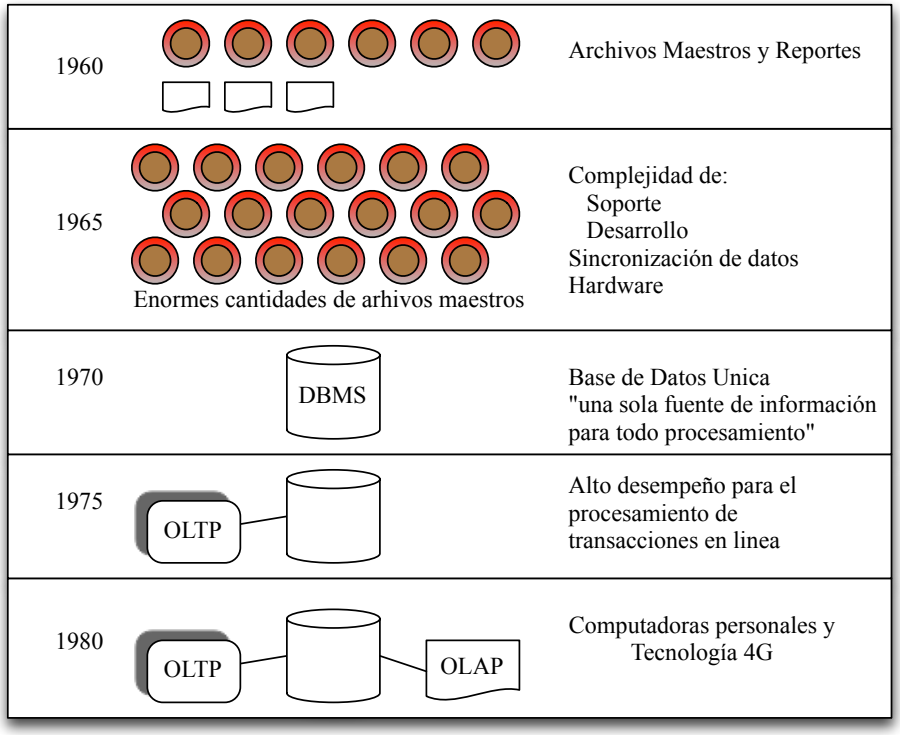

Figura 3.1: Evolución de los sistemas de almacenamiento

división de los sistemas en OLTP y OLAP se han desarrollado nuevos sistemas bajo los esquemas actuales.

Lo anterior dio lugar al surgimiento de la elaboración de aplicaciones de extracción, con el objetivo de alimentar sistemas OLAP con datos provenientes de sistemas OLTP. Un programa extractor como también se les denomina, es un programa que busca y analiza a través de un archivo o bases de datos, usando algún criterio para seleccionar datos y transportar dichos datos a otro archivo o bases de datos (ver figura 3.2).

Debido al dinamismo que presentan los datos de las aplicaciones OLTP, el uso de programas extractores enfrentaba serios desafíos, entre los cuales están:

- **Falta de credibilidad de los datos**: cuando se debía efectuar análisis sobre algún aspecto específico de una organización, se podía llegar a resultados diferentes debido a que las diversas fuentes de información no estaban consolidadas ni sincronizadas. Los motivos de la falta de credibilidad son los siguientes:

  - No existe una base de tiempo para realizar las extracción y sincronización de los datos.
  - A pesar de usar los mismos algoritmos para el análisis de información no se hace sobre el mismo tipo de entradas.
  - El nivel de detalle de las extracciones no está definido de la misma forma para todos los que acceden a la información.

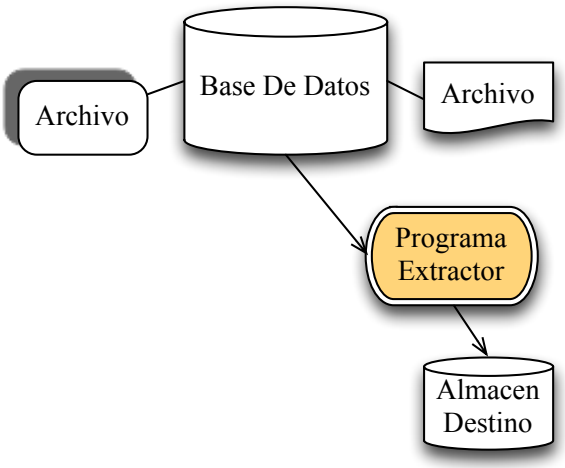

Figura 3.2: Programa Extractor

- Los datos sobre los cuales se efectúa el análisis no están consolidados.

- **Productividad**: cuando se tiene un gran numero de fuentes de información dentro de una organización, tareas como las siguientes se vuelven muy complicadas y difíciles de hacer:

  - Localizar y analizar los datos para analizar.
  - Compilar los datos para el análisis.
  - Entregar al programador o analista los recursos para poder hacer análisis de forma fácil.

- **Imposibilidad de transformar los datos en información:** cuando se intenta hacer análisis sobre información que no está correctamente unificada, es imposible obtener información acerca de aspectos como: un mismo elemento que está en diversas fuentes de información con nombres diversos, un elemento distinto que tiene nombres diversos en las fuentes de información o de elementos que existen sólo en una fuente de información. De esta forma resulta complicado afirmar que nuestras fuentes de datos nos dan información y no sólo datos, debido a que no se tienen unificadas bajo un mismo contexto.

Los sistemas OLTP cambiaron de enfoque, dejando atrás las aplicaciones extractores, ya que era imposible solventar los retos actuales son ese tipo de enfoque. Esto dio pie a una nueva arquitectura para la construcción de nuevos sistemas de almacenamiento masivo, denominados Data Warehouse (almacenes de datos).

3.1.2. Elementos principales de los almacenes de datos

Un Data Warehouse concentra información proveniente de sistemas operacionales. Para hacer una diferencia más clara entre la información que guarda un Data Wa-

rehouse, la arquitectura bajo la cual se desarrollaron los almacenes de datos, define dos tipos fundamentales de datos: los *datos primitivos* y los *datos derivados*. En la tabla 3.1 se muestran las principales diferencia entre estos dos tipos de datos.

Datos Primitivos (Operacionales)	Datos Derivados (Analíticos)
Orientados a la aplicación	Orientados a aspectos o temas
Detallados	Resumidos, de otra manera redefinidos
Precisión en el momento del acceso	Representan valores a través del tiempo, instantáneos
Sirve a los perfiles operativos	Sirve a los perfiles de gestión y toma de decisiones
Puede ser actualizado	Es de sólo lectura
Se ejecuta reiteradamente	Se ejecuta de forma heurística
Los requerimientos para el procesamiento son entendidos a priori	Los requerimientos para el procesamiento no son entendidos a priori
Sensibles al desempeño computacional	No importa el desempeño computacional
Compatibles con ciclos de desarrollo clásicos	Completamente diferentes a ciclos de desarrollo clásicos
Son accedidos de forma unitaria	Son accedidos de forma grupal
Manejados con base a transacciones	Manejados con base a análisis
El control de actualizaciones es una preocupación central	El control de las actualizaciones no es problema
Un elemento claves es una alta disponibilidad	La disponibilidad no es elemento crucial
Administrado de forma integral	Administrado a través de subconjuntos
No hay redundancia	La redundancia es un factor primordial
Estructura estática, contenido variable	Estructura flexible
Pequeñas cantidades de datos usadas en un proceso	Grandes cantidades de datos usadas en un proceso.
Soporta operaciones del día a día	Soporta necesidades gerenciales
Alta probabilidad de acceso	Baja o modesta probabilidad de acceso

Cuadro 3.1: Características de los datos primitivos y derivados

Después de analizar las características de los datos que componen un Data Warehouse, describiremos algunos de los aspectos más importantes de los almacenes de datos. Un Data Warehouse es orientado a un tema específico, integrado, no volátil y

una colección de datos variante en el tiempo. A continuación detallaremos cada uno de estos aspectos.

- **Orientado a un tema específico**: los sistemas operacionales están siempre organizados en torno a aplicaciones funcionales de la compañía, por ejemplo en una compañía de seguros. Los sistemas operacionales están enfocados en aplicaciones para tratar aspectos relacionados con los autos, pólizas de personas o elementos materiales y agentes de seguros. Sin embargo las temáticas principales para la compañía desde un punto de vista corporativo son el cliente, las primas e intereses y las demandas. De igual forma para otro tipo de compañía encontramos un conjunto único de temas específicos y relevantes. En la figura 3.3 encontramos algunas aplicaciones de los sistemas operacionales y los temas específicos que interesan y derivan en la construcción de almacenes de datos.

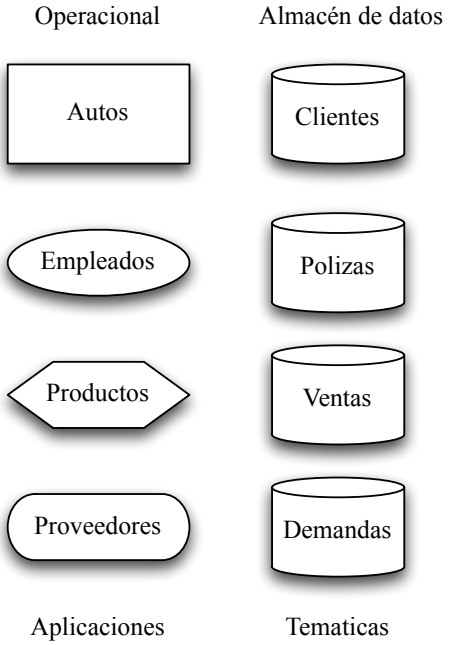

Figura 3.3: La orientación a temas específicos de un Data Warehouse

- **Integrado**: de todos los aspectos de un Data Warehouse, el ser integrado es el más importante de ellos. Un Data Warehouse es alimentado por diversas fuentes de datos, conforme la información es depositada en el almacén, ésta es convertida, reformateada y resumida gradualmente. El resultado final son los datos (contenidos dentro del almacén), con un único esquema e imagen colectiva. La figura 3.4 ilustra algunos ejemplos de transformaciones que ocurren al integrar las fuentes de información de los sistemas operacionales.

Los desarrolladores encargados de elaborar sistemas transaccionales nunca toman en cuenta que los datos que son capturados y almacenados por este tipo de sistemas

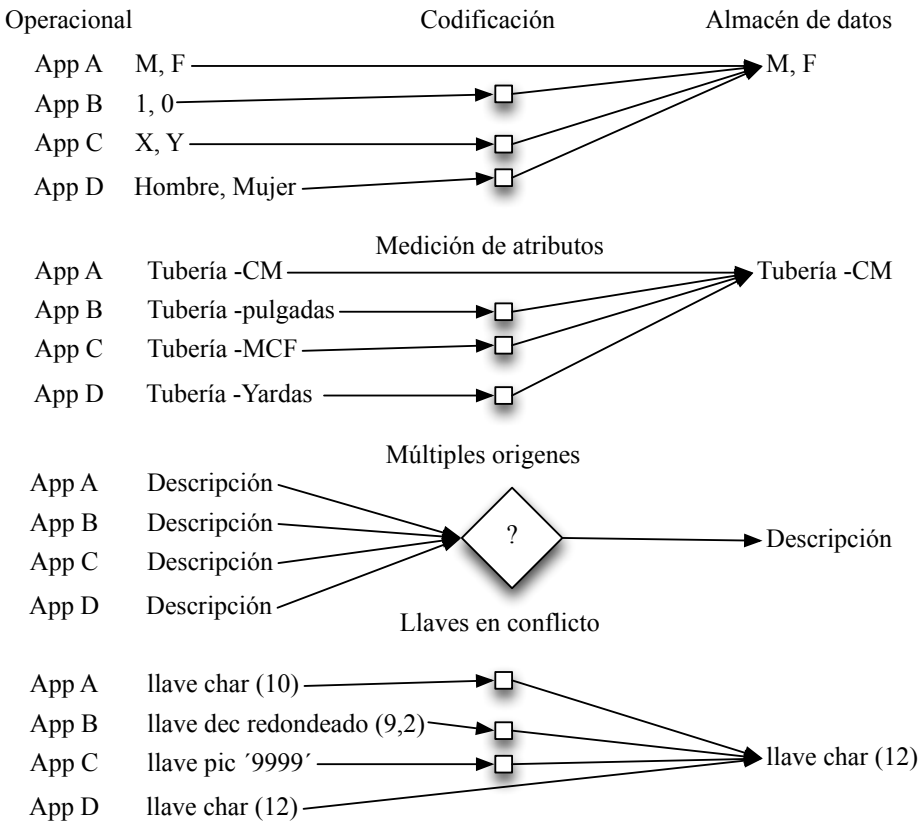

Figura 3.4: Integración de fuentes de datos

pasaran después por un proceso de integración y consolidación. Sin estas consideraciones se pasan por alto aspectos de codificación consistente como: convenciones de nombres, características de los atributos físicos, unidades de medias sobre atributos, por mencionar algunos. El resultado es que los datos generados por las aplicaciones son de características muy diferentes aunque se trabaje con aspectos similares.

- **No volátil** (No se modifica constantemente): a diferencia de lo que ocurre en un sistema transaccional donde la información es insertada y modificada de forma constante, en un Data Warehouse los datos son cargados y accedidos por lo general de forma masiva, pero no son modificados. En lugar de ello cuando los datos son cargados dentro del almacén se crea una imagen consistente del mismo.

- **Variante en el tiempo**: puesto que un Data Warehouse guarda la información histórica de los sistemas transaccionales, esto implica que cada unidad de los datos del almacén sea precisa solo en algunos momentos en el tiempo.

3.1.3. Aplicaciones y uso de los almacenes de datos

Los primeros sectores que adoptaron arquitecturas basadas en almacenes de datos incluían organizaciones de telecomunicaciones, compañías de seguros, bancos y tiendas de autoservicio. Poco después los almacenes de datos fueron considerados por un sin fin de organizaciones de diversos giros.

Las compañías cuyo perfil es adecuado para la implementación de almacenes de datos se caracterizan por:

- Ejecutar una gran cantidad de transacciones.
- Tener una gran diversidad de usuarios.
- Almacenar grandes volúmenes de datos en repositorios diversos; y sobre todo.
- Formar parte de un mercado muy competitivo.

Dentro de la arquitectura de los almacenes de datos es inconcebible el concepto de un solo usuario final. Por el contrario, los usuarios finales de un Data Warehouse suelen ser una comunidad entera y de gran diversidad. Esta comunidad se puede agrupar en cinco tipos de perfiles:

- **Granjeros**: son aquellos usuarios predecibles, hacen actividades y consultas rutinarias sobre el Data Warehouse, sus consultas suelen ser cortas, dado que los granjeros en todo momento saben la información que quieren, pueden ir y obtener la información de manera rápida.
- **Exploradores**: son aquellos usuarios que actúan de una forma totalmente impredecible debido a que no conocen con certeza la información que desean. Por lo general explora grandes volúmenes de datos de forma heurística. En muchas ocasiones el explorador busca algo que nunca encuentra pero en otras encuentra resultados sumamente relevantes.
- **Minadores**: son aquellos usuarios que escarban en grandes volúmenes de información y determinan qué datos aportan conocimiento relevante y cuáles no. Por lo general se apoyan de herramientas estadísticas. Los minadores determinan la fuerza de hipótesis y afirmaciones formuladas por exploradores.
- **Turistas**: son aquellos usuarios con basta experiencia y conocimiento, siempre saben donde encontrar la información que requieren.

3.2. El proceso ETL

El proceso ETL es el fundamento de un Data Warehouse. Un proceso ETL diseñado y llevado a cabo de forma apropiada, extrae datos de las fuentes de información, refuerza la calidad y consistencia de los mismos y finalmente entrega los datos en una

presentación y formato listo para ser consumidos por aplicaciones para la toma de decisiones.

El proceso ETL determina el éxito o fracaso de la implementación de un Data Warehouse. A pesar de que la construcción del proceso ETL es una actividad que no es visible a usuarios finales, esta tarea consume casi el 70 por ciento de los recursos necesarios para la implementación y mantenimiento de un Data Warehouse convencional

El proceso ETL agrega un valor significativo a los datos. Este va más allá de la transportación de los datos de las fuentes orígenes a la carga dentro del Data Warehouse. En especifico el proceso ETL se encarga de:

- Remover errores y corregir datos faltantes.
- Proporcionar medidas documentadas de la calidad de los datos.
- Supervisar el flujo de los datos transaccionales.
- Ajustar y transformar los datos de múltiples fuentes para poder unificarlos.
- Estructurar los datos para ser usados por las herramientas y usuarios finales.

El proceso ETL es intuitivo y fácil de comprender. La idea básica del proceso ETL es: tomar los datos de las fuentes de información y depositarla en el Data Warehouse; sin embargo la limpieza y transformación de la información son procesos mucho más complicados de lo que se puede apreciar a simple vista. De hecho, estos procesos generales suelen dividirse en un sin fin de tareas específicas, dependiendo de las características de las fuentes de datos, las reglas de negocios, las herramientas existentes y las características del Data Warehouse final.

El reto para un correcto desarrollo del proceso ETL es planificar adecuadamente la gran cantidad de tareas, para lo cual es indispensable conservar la perspectiva simple e intuitiva de la misión del proceso.

3.2.1. Requerimientos y consideraciones generales

El diseño y alcances del proceso ETL están siempre en función de las necesidades de negocio, es decir, los propósitos específicos por los que se quiere consolidar la información dentro de el Data Warehouse. Puesto que un Data Warehouse es para propósitos meramente de análisis, se debe tener en cuenta el tipo de información que se quiere obtener, y con base a ello determinar los elementos y consideraciones de las fuentes de información que estarán involucradas en el análisis.

Resulta indispensable antes de comenzar con el proceso ETL, comprender las necesidades de las personas involucradas en el análisis que se realizará sobre el Data Warehouse, principalmente para saber qué información va a formar parte del proceso, qué transformaciones se tendrán que hacer sobre la misma y en qué forma será consultada y entregada la información final.

Una vez conocidas las necesidades que dan pie al proceso ETL, se puede entonces determinar los elementos de las fuentes de información que participaran en el proceso, los mecanismos de acceso y extracción de los datos, las transformaciones y políticas de negocio que habrá que aplicar sobre los mismos y determinar así la forma, presentación y estructura en que finalmente serán almacenados para su consumo final.

Otros aspectos a considerar posteriores a la realización del proceso ETL son:

- **Grado de integración de los datos**: cual será el volumen o unidad mínima de información sobre la cual se harán las cargas de información.
- **Latencia de los datos**: con que frecuencia se harán adiciones de información al Data Warehouse.
- **Requerimientos de seguridad**: cuales son las políticas de seguridad que se aplicaran para la carga de información.
- **Perfil de los datos**: cual es el tipo y naturaleza de las fuentes de información que intervienen.
- **Linaje y archivado**: cual será el mecanismos para dar seguimiento a los datos y sus transformaciones, desde su sistema origen hasta su destino final dentro del Data Warehouse.

3.2.2. La arquitectura general

La arquitectura general sobre la cual será desarrollado el proceso ETL es un elemento primordial para garantizar el éxito de la implementación. Un mal diseño y elaboración de la arquitectura involucraría la nueva implementación total del proceso.

Algunos aspectos a considerar para el desarrollo de la arquitectura del proceso son los siguientes:

- La arquitectura se implementara sobre una herramienta ETL o se desarrollará codificando los módulos del proceso ETL que sean requeridos de forma manual. Elaborar la implementación sobre una herramienta ETL hace el desarrollo más rápido pero requiere de un perfecto entendimiento de la política de negocio y de los objetivos que se persiguen; de manera adicional, resuelve problemas tediosos que se tendrían en una implementación manual. Entre estos podemos citar: la gestión de meta-datos, la sincronización de los repositorios de información y los mapeos entre fuentes de datos origen y destino. Por el contrario si se decide realizar la codificación de forma manual, se puede tener un control más específico sobre las transformaciones y los meta-datos.
- La carga de información será a través de procesos batch o sobre un flujo de datos. La arquitectura estándar del proceso ETL esta basada en cargas batch de información periódicas, éstas suelen ser lentas debido al gran volumen de

información que tienen que transportar. Sin embargo si en la práctica se requiere que un Data Warehouse sea actualizado de forma constante y rápida, el esquema de actualización por batch no es funcional y se puede recurrir a la carga por flujo de datos constante. Dado que la naturaleza de estos dos esquemas son muy diferentes, su impacto en la arquitectura general del proceso es muy grande, puesto que todas rutinas de extracción, limpieza, integración y entregas se implementarían de maneras radicalmente opuestas.

- La dependencia entre tareas será vertical u horizontal. Si se opta por hacer que las tareas sean independientes sobre un flujo de trabajo horizontal, implica el hecho de que los datos de dos fuentes de información pueden ser procesados de forma independiente. Por el contrario si se hacen tareas independientes sobre un flujo vertical, los trabajos de extracción, limpieza y carga de las diversas fuentes de información estarán sincronizados y la carga de información se hará de manera simultánea.

La arquitectura general de un Data Warehouse se puede apreciar en la figura 3.5. Podemos observar de la figura que el Data Warehouse se divide en dos grandes secciones, la trasera y la delantera.

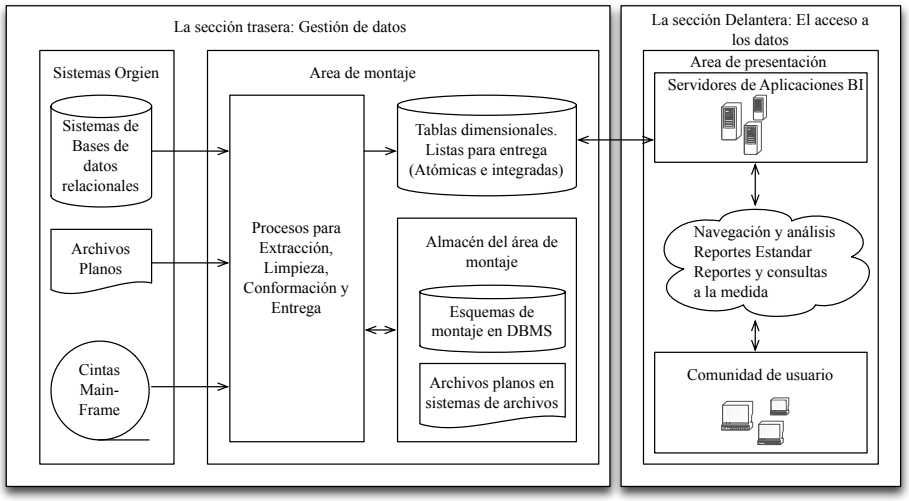

Figura 3.5: Arquitectura general de un Data Warehouse

Las secciones del Data Warehouse están física, lógica y administrativamente separados, esto implica que dependen de estructuras de datos y administraciones diferentes.

En la sección trasera podemos ver en principio las diversas fuentes de información de los sistemas. En el área de montaje se procede con los proceso de extracción, limpieza, conformación y entrega. El resultado final es puesto en dos almacenes diferentes, uno con tablas dimensionales para poder ser consumidos por aplicaciones BI residentes en la sección delantera y el segundo un almacén interno del área de montaje.

En la sección delantera tenemos tres elementos principales: los servidores de aplicaciones BI encargados de consumir la información del Data Warehouse, un conjunto de servicios remotos de consulta y demanda de información que son atendidos por los servidores BI y que son consumidos por una comunidad de usuarios.

3.2.3. Estructuras de datos del proceso ETL

La sección trasera del Data Warehouse es comúnmente llamado el área de «*staging*», que en este contexto tiene el significado de *escritura a disco*. El proceso ETL involucra la interacción con diversas estructuras de datos que se leen y escriben en diversos dispositivos de almacenamiento, por tal motivo revisamos las estructuras de datos más frecuentes que se presentan a lo largo del proceso. Revisaremos aspectos puntuales como: estructura, métodos de extracción e inserción.

Archivos de texto plano

Los archivos de texto plano son aquellos que almacenan la información en filas y columnas para emular la estructura de una tabla de una base de datos. Si se trabaja bajo ambientes Windows o UNIX, los archivos están codificados en el estándar ASCII (Amercian Standar Code for Information Interchange). Los archivos planos pueden ser manipulados y procesados por algunas herramientas ETL o por lenguajes de secuencias de comandos como si se tratasen de tablas de bases de datos, solo que en algunas ocasiones mucho más rápido que ellas.

Las operaciones de ordenado, mezcla, eliminado, reemplazo y muchas otras funciones de migración de datos se ejecutan mucho más rápido sobre archivos de texto plano que sobre sistemas DBMS.

Una seria consideración que se debe tomar en cuenta cuando se trabaja con archivos de texto plano es que se debe de tener un correcto seguimiento y gestión de los meta-datos. Este trabajo algunas veces se puede evitar si se emplea una herramienta ETL, de lo contrario es algo que tiene que llevarse a cabo puesto que resulta primordial en fases del proceso como en la transformación, mapeo y carga de la información.

En la fase de entrega, los archivos plano son una excelente alternativa a las bases de datos relacionales, puesto que tiene un mejor desempeño y facilitan tareas como:

- Escritura en disco de los datos para su monitoreo y seguimiento.
- Ordenación de la información.
- Filtrado de los datos.
- Remplazo y substitución de cadenas de texto.
- Aplicación de operaciones de agregación.
- Referenciación de fuentes de información.

Un ejemplo de archivo de plano puede ser visto en la figura 3.6.

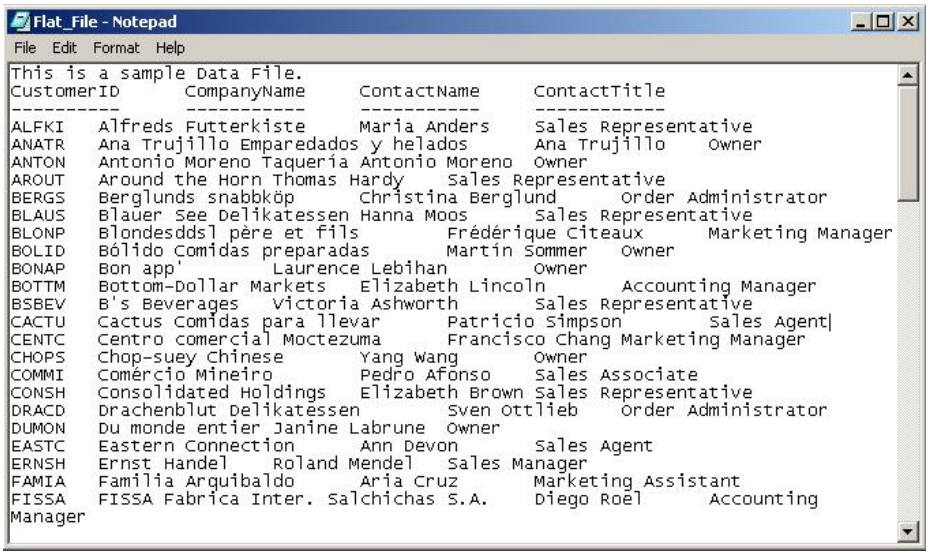

Figura 3.6: Archivo plano

Archivos XML

XML (Lenguaje extensible de marcado), es un meta lenguaje que permite definir la gramática de lenguajes específicos. Fue diseñado para describir datos, lo que le permite la lectura de datos a través de aplicaciones; actualmente es ampliamente usado para la definición nuevos lenguajes y para el almacenamiento e intercambio de información entre sistemas de información.

Un documento XML se estructura de forma jerárquica con base a etiquetas. Los documentos XML están compuestos por los siguientes elementos:

- Prólogo: es el primer elemento de todo documento XML, contiene los siguientes elementos:
  - Declaración XML: sentencia que especifica que el documento es de tipo XML.
  - Declaración de tipo de documento: sentencia que vincula el documento actual con una definición de tipo de documento.
  - Comentarios: opcionales y con fines por lo general informativos, no tienen valides sintáctica para el documento.
- Cuerpo: es un elemento no opcional, debe de contener un solo elemento raíz y dentro de él se define el contenido y componentes del documento los elementos que lo conforman son:
  - Etiquetas: son los elementos base de los documentos XML, éstas pueden definir dentro de sí: otras etiquetas, caracteres o ambos; o pueden estar vacíos.

```
<?xml version="1.0" ?>
- <clientes>
  - <registro>
      <fecha>01/01/2009</fecha>
      <codigocliente>0001</codigocliente>
      <telefono>555555</telefono>
      <direccion>calle x No yyy Boston</direccion>
    </registro>
  - <registro>
      <fecha>01/01/2009</fecha>
      <codigocliente>0002</codigocliente>
      <telefono>55555</telefono>
      <direccion>calle x No yyy Boston</direccion>
    </registro>
  - <registro>
      <fecha>01/01/2009</fecha>
      <codigocliente>0003</codigocliente>
      <telefono>55555</telefono>
      <direccion>calle x No yyy Boston</direccion>
    </registro>
  </clientes>
```

Figura 3.7: Un archivo XML

- Atributos: son elementos adicionales a las etiquetas, incorporan características y propiedades a los elementos, tienen la forma clave=valor.
- Entidades predefinidas: elementos propios del lenguaje para declarar caracteres especiales.
- Secciones CDATA : es una construcción para especificar datos utilizando cualquier caracter (inclusive especiales y reservados) sin que sean interpretados como elementos XML.
- Comentarios: elementos que se agregan en el documento pero que no son tomados en cuenta al procesar el documento.

Un ejemplo pequeño de un documento XML es mostrado en la figura 3.7:

Los documentos XML se dividen en dos tipos:

- Bien formados: son aquellos que cumplen íntegramente con la sintaxis del lenguaje, es decir que está escrito conforme a las reglas de especificación del lenguaje.
- Válidos: son aquellos documentos que además de ser bien formados, basan las relaciones de todos sus elementos bajo un esquema de definición único. Este esquema de definición puede ser dado por un DTD (definición de tipo de documento) o por elementos denominados XSchema. Estos últimos basan su sintaxis

en XML, permiten especificar nuevos tipos de datos y son extensibles, por lo que la mayoría de los documentos XML actuales basan su definición en este estándar.

Los conjuntos de datos definidos en XML pueden ser tratados de forma adecuada si se conocen las definiciones del documento, es decir si se trabaja con documentos válidos y se conocen los elementos que lo definen (DTD o XSchema). Con esta información es fácil realizar la extracción y categorización al igual que las transformaciones y carga; de no contar con esta información el procesamiento de los archivos se complica.

Bases de datos relacionales

El modelo de base de datos relaciona fue propuesto por E. F. Codd en su famoso articulo «*A relational model of data for large shared data banks*» [42]. Este artículo fue el origen de los sistemas de bases de datos relacionales actuales; y el documento fundamental de la teoría de las bases de datos relacionales.

El fundamento del modelo relacional es el concepto de relación, el resto del fundamento del modelo relacional está basado en la teoría de conjuntos y en la lógica de predicados.

Los elementos principales del modelo relacional son los siguientes:

- **Relación**: podemos ver a una relación como una tabla que guarda información a través de columnas y filas.
- **Atributo**: es una columna con denominación dentro de una relación.
- **Dominio**: es el conjunto de valores válidos que puede tomar un atributo.
- **Tupla**: es una fila dentro de una relación.
- **Grado**: es el número de atributos de una relación.
- **Cardinalidad**: es el número de tuplas contenidas dentro de una relación.
- **Base de datos relacional**: es una colección de relaciones normalizadas, en la que cada relación tiene un nombre único.

En el proceso ETL es importante detectar los elementos anteriormente mencionados, ya que esta información representa los meta-datos de la base datos y con base a esta se definirán puntos mas avanzados como mapeos, transformaciones y mecanismos de extracción e inserciones.

El lenguaje de manipulación de los datos sobre las bases de datos relacionales es SQL, por tal motivo es importante tener un adecuado manejo del lenguaje SQL para facilitar la elaboración de mecanismos de extracción e inserción de información.

Algunos puntos a favor que tiene las bases de datos dentro del proceso ETL son los siguientes:

- **Presencia de meta-datos**: el contar con meta-datos hace mucho más fácil la manipulación y mapeo de la información a lo largo del proceso ETL, además de que elimina el problema de gestión de forma independiente al ser parte del modelo mismo modelo.

- **Basadas en un modelo matemático**: dado que tiene un fundamento matemático es fácil con base a este poder, detectar inconsistencias o errores en los modelos de datos.

- **Fácil recuperación e inserción de la información**: debido a que definen un lenguaje único de consulta e inserción (SQL), es fácil poder recuperar e insertar información dentro de la base de datos.

- **Soporte y administración**: por lo general dentro de las organizaciones hay personal encargado del mantenimiento y soporte de las bases de datos, es por ello que suelen ser un medio más fiable que los descritos anteriormente.

Si se analiza a las bases de datos como medios de almacenamiento final del Data Warehouse, suelen ser la mejor alternativa debido a las bondades que ya presentamos anteriormente.

Fuentes de datos no relacionales

La integración de fuentes de datos heterogéneas es un reto que los desarrolladores ETL deben constantemente afrontar, además de integrar nuevas dominios de aplicación dependiendo del crecimiento y expansión del dominio del Data Warehouse.

Muchas de las fuentes de información que intervienen en los procesos ETL provienen de fuentes de datos no relacionales o en su defecto son de bases de datos relacionales que no necesariamente tienen relación entre sí. Algunas fuentes de información no relacionales son: archivos VSAM (Virtual Storage Access Method), archivos planos, hojas de cálculo, etc.

Traer fuentes de datos tan dispersas y diversas a un único modelo relacional es una práctica común dentro del proceso ETL. Sin embargo, analizando el problema suele ser algo no necesario. La correcta definición de un proceso ETL puede gestionar adecuadamente las fuentes de datos heterogéneas, minimizando la necesidad de almacenar toda la información dentro de una misma base de datos.

La integración de fuentes de datos no relacionales involucra el esfuerzo extra de la revisión de la integridad de la información. Esto debido a que este tipo de fuentes de información no posee elementos como la integridad referencial.

3.2.4. Meta-datos

Los meta-datos son un tópico fundamental en el proceso ETL, debido a que todas y cada una de las aplicaciones que interactúan con el Data Warehouse hacen uso

de los meta-datos para poder lograr la interoperabilidad, es decir, la comprensión e intercambio de información.

Es imposible idear una solución que gestione de principio a fin los meta-datos de las fuentes de información que involucran en el proceso ETL. En lugar de ello, los enfoques actuales implementan componentes para la gestión de meta-datos entre los elementos del proceso que intercambian información.

El proceso ETL es el centro que manipula y gestiona la información que se depositará dentro del Data Warehouse. Debido a ello es intuitivo pensar que este proceso tiene la responsabilidad de gestionar y almacenar los meta-datos que darán soporte al Data Warehouse.

A lo largo del proceso ETL se obtendrán, usarán y generarán la mayoría de los meta-datos del Data Warehouse; así mismo el proceso ETL definirá el *linaje de los datos*. *El linaje de los datos* rastrea los datos desde su la ubicación donde fueron extraídos y documenta de forma precisa que transformaciones son hechas sobre ellas hasta que son finalmente depositados en el almacén.

Una de las principales razones por las cuales es difícil administrar los meta-datos, es que su definición es ambigua ya que es difícil definir que comprende el termino de meta-datos.

Los meta-datos pueden ser definidos como entidades que describen y definen datos. Dentro del contexto ETL podemos encontrar dos clases de meta-datos, los meta-datos de la sección trasera y lo meta-datos de la sección delantera.

Lo meta-datos la sección delantera sirven para guiar el proceso de extracción, limpieza y carga de información.

En la sección delantera su función es más descriptiva y ayuda al desarrollo de las consultas y reportes. Ayuda al administrador ETL a transportar la información al Data Warehouse y muestra a los usuarios de negocio el origen de los datos.

Para poder comprender mejor el concepto de los meta-datos y su importancia dentro del proceso ETL, a continuación presentamos una clasificación de los tipos de meta-datos que intervienen en el proceso:

- **Meta-datos de los sistemas origen**: éstos definen la estructura interna de los sistemas de las fuentes de información, entre ellas encontramos: repositorios, esquemas origen, copias de libros, tablas de sistemas relacionales origen y DDL, hojas de calculo, especificaciones URL, información descriptiva de las fuentes de información y procesos de información internos como bitácoras.

- **Meta-datos del almacenamiento temporal y procesamiento**: almacenan información acerca de, la adquisición de los datos, gestión de las tablas de dimensiones, relaciones de agregación y transformación, seguimiento de flujo de datos, bitácoras y documentación

- **Meta-datos de los DBMS**: una vez cargada la información en los Data Marts (subconjunto de datos que almacena información de una área específica del negocio) o en el Data Warehouse, los meta-datos ofrecen información acerca del contenido de las tablas, índices, seguimiento del procesamiento, elementos

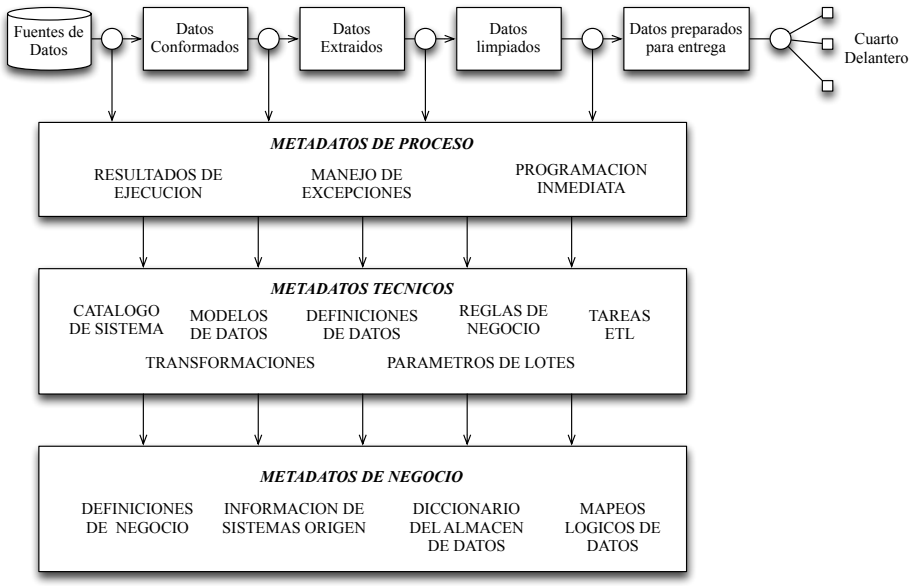

Figura 3.8: Los meta-datos en el proceso ETL

de seguridad del almacén, procedimientos almacenados y scripts SQL así como elementos de respaldo.

- **Meta-datos de la sección delantera**: llegando a este nivel los meta-datos ayudan más a describir las reglas y nombres de negocios que operan sobre la información, definen elementos como: descripciones y nombres de negocio, especificaciones de reportes, instrucciones de impresión y entrega de datos, perfiles de acceso a información, perfiles de acceso a información, acceso a mapas de elementos diversos, etc.

Una clasificación más general de los meta-datos, con base al perfil de los usuarios que los consumen es el siguiente:

- **Meta-datos de negocio**: describen el significado de los datos en desde el punto de vista de las reglas de negocio.

- **Meta-datos técnicos**: representan los aspectos técnicos de los datos, incluidos los atributos, tipos de datos, longitudes, linaje. Son resultado de la elaboración de perfiles de datos.

- **Meta-datos generados por procesos de ejecución**: guardan resultados estadísticos y de seguimiento del proceso ETL, elementos tales como: mapas de transformación entre fuentes de información, cantidad de registros extraídos y almacenados, registros no clasificados, tiempos de carga y extracción por mencionar algunos.

En la figura 3.8 se muestran los diversos tipos de meta-datos distribuidos en los tres niveles de abstracción. Como base de todos ellos tenemos los meta-datos de negocio, posteriormente los meta-datos técnicos y por último los de proceso, observamos que los meta-datos están presentes en las transiciones de las diversas fases del proceso ETL, puesto que definen la configuración origen y destino de cada una de las transformaciones que van sufriendo los datos a lo largo del proceso.

3.2.5. La extracción, limpieza y conformación

El primer paso para la integración de las fuentes de información es la extracción exitosa de los datos de los sistemas origen.

La mejor forma de empezar el proceso de integración es la definición de las interfaces y canales de comunicación de los sistemas orígenes con el proceso ETL. Puesto que cada fuente de información tiene un conjunto propio de características, es necesario dar un tratamiento específico a cada una de ellas para poder hacer una correcta extracción y transformación de la información.

Las organizaciones evolucionan y conforme eso pasa se desarrollan sistemas de información para ayudar las labores de la empresa. Entre ellos podemos encontrar: puntos de venta, gestión de inventarios, control de producción y en una gran cantidad de sistemas legados; mas allá de que estos sistemas de información trabajen de forma independiente y hayan sido adquiridos en tiempos diversos. Generalmente son incompatibles desde el punto de vista lógico y físico. El proceso ETL generalmente se encuentra con fuentes de información con diferentes objetos:

- Sistemas gestores de bases de datos
- Sistemas operativos
- Hardware
- Protocolos de comunicación

El mapeo lógico de datos es la parte medular de la extracción de información, en sus manos está el identificar las características y propiedades de las fuentes de información, determinar los que serán extraídos y de qué forma. Asimismo determina la forma en que serán transportados y cómo serán considerados para tratamientos posteriores. La implementación física del proceso ETL puede ser catastrófica si no se hace una correcta orquestación de los datos de los fuentes orígenes para poder hacer una correcta interfaz con el resto del proceso ETL.

Una excelente práctica para la implementación del proceso de extracción elaborar el diseño lógico de datos antes de la implementación física, por tal motivo es trascendental elaborar las siguientes rutinas antes de comenzar con la implementación física del proceso ETL:

- **Elaborar un plan**: se debe tener un plan con el diseño lógico del proceso, acorde a las especificaciones del arquitecto del Data Warehouse, a fin de identificar

tanto los sistemas origen como el repositorio destino y poder así idear la ruta ideal que ha de seguir el proceso para poder llevar a cabo la integración.

- **Identificar las fuentes de datos candidatas**: dado que el Data Warehouse almacena datos sobre los cuales se hacen análisis de temas concretos, se debe de elegir aquellas fuentes de información que aporte información para el tópico que se este analizando.
- **Analizar los sistemas origen con herramientas de análisis de datos**: se debe tener un análisis de los sistemas origen para poder determinar la calidad de los datos, su integridad y su aptitud para el propósito deseado.
- **Recibir un «*paseo*» a través del linaje de datos y las reglas de negocio**: una ves hecho el análisis sobre los datos, se debe inspeccionar la arquitectura del proceso ETL, al igual que el linaje y las reglas de negocio de extracción que seguirán los datos a lo largo del proceso. De esta manera se podrán identificar las transformaciones y modificaciones que sufrirán los datos.
- **Recibir un *«paseo»* a través del modelo de datos del Data Warehouse**: se debe de comprender a la totalidad el modelo físico de datos del Data Warehouse destino, a fin visualizar el destino final que tendrán las diversas fuentes de información e idear así los mapeos y transformaciones adecuados.
- **Validar los cálculos y fórmulas**: una vez definidas las transformaciones y mapeos y habiendo definido el linaje de datos, es necesario cotejar esta información a fin de verificar que se están llevando por buen camino los datos.

Una vez hecho el análisis anterior se puede a elaborar el mapa lógico de datos. Un mapa lógico de datos es una estructura, por lo general una tabla con las características de las fuentes de información, algunos elementos que tiene esta tabla son:

- Nombre de la tabla destino
- Nombre de la columna destino
- Tipo de tabla
- SCD (Grado del cambio de dimensión)
- Base de datos fuente
- Nombre de la tabla origen
- Nombre de la columna destino
- Transformaciones

Para elaborar el mapa lógico de datos, se procede por diversas fases:

- La fase de exploración y descubrimiento de información, donde se colecta y documenta la información de los diversos sistemas fuentes de datos, además de ello se comienza con el seguimiento de los mismos.
- La fase de análisis del contenido de datos, donde se navega a través de la fuentes de datos y se recaba información acerca de valorares nulos, codificación y estructura de los esquemas de datos.
- La fase de recolección de reglas de negocio
- La fase de integración de fuentes de datos heterogéneas

Posterior al análisis de las fuentes de información se procede con la implantación de los mecanismos de extracción de información sobre las diversas fuentes de datos. Las fases por las que se pasa en este proceso son:

- Conexión a diversas fuentes de información
  - A través de ODBC
  - Lectura de archivos contenidos en diversos esquemas
  - Lectura y parseo de archivos XML
- Extracción sobre datos que se han modificado:
  - Detección de cambios
  - Detección de registros eliminados o sobre escritos

Posterior a la extracción los datos deben pasar por un proceso de limpieza, con la finalidad de que los datos que se depositen en el almacén estén libres de errores, permitiendo así análisis más fiables sobre los mismo. Adicional a esta fase, se procede a la conformación de los datos para poder agrupar y entregar en el formato indicado la información final al Data Warehouse.

El objetivo de la fase de limpieza es mejorar la calidad de los datos que son extraídos para entregarlos mejor al Data Warehouse. Para que los datos sean considerados precisos y de calidad, deben de cumplir las siguientes características: ser correctos, no ambiguos, consistentes y completos.

Para poder cumplir con los objetivos de la fase de limpieza se definen roles y se les asignan actividades específicas, algunas de ellas son: gestión del Data Warehouse, administración de la información, supervisión de la calidad de la información, gestión de dimensiones, recuperación de tablas de hechos. Por lo general estas actividades suelen tener objetivos en conflicto; por lo que parte del proceso es definir en qué grado se han de cumplir las expectativas de cada actividad para poder generar un proceso que tenga calidad en cada uno de los aspectos involucrados.

Una ves que se recuperó y limpió la información de las fuentes de información por separado; la segunda parte del proceso hace una limpieza pero esta vez sobre los datos conjuntados. Los aspectos a considerar en esta fase son:

- **Estandarización y consolidación de los diversos perfiles de datos**: de acuerdo al resultado del análisis de los perfiles de datos se tienen que homogenizar aspectos como: definiciones de esquemas, objetos de negocio, dominios, fuentes de datos, definiciones de tablas, sinónimos, reglas de datos y valores de datos.

- **Manejo de errores**: para poder gestionar los errores que se suscitan en esta fase se hace uso de una tabla donde se lleva el registro de los eventos que generan errores y de las fuentes involucradas en dichos errores.

- **Supervisión de dimensiones**: es el proceso mediante el cual se evalúa si la información extraída puede ser correctamente almacenada en las tablas de dimensión del Data Warehouse.

La última fase de éste proceso es la conformación de la información para poder ser cargada. Esto involucra el cotejar la información, agruparla y construir las tablas de hechos y dimensiones que serán parte del Data Warehouse final. Para poder llegar a este punto, la información necesita ser estructuralmente idéntica, libre de registros inválidos, estandarizada en términos de su contenido, libre de registros duplicados y sólo entonces puede ser destilada dentro Data Warehouse.

Cuando se pasa por el proceso de conformación, también los códigos y datos son unificados a un nivel semántico. Por ejemplo, los códigos de genero (M,F), (H,M) y (Hombre y Mujer) provenientes de tres fuentes de datos diversas, son consolidadas dentro de una sola convención.

Las tareas principales de esta fase de conformación son:

- **Conformación de información para las tablas de dimensiones**: la identificación y agrupación de información de las diversas fuentes de datos. Para poder poblar las tablas del Data Warehouse se hace a través del análisis de dimensiones compatibles, para que dos tablas sean afines para crear una dimensión. Estas deben compartir uno o más atributos cuyos valores caigan sobre los mismos dominios.

- **Conformación de información para tablas de hechos**: después de identificar las información que será parte de las tablas de dimensión, ahora queda definir las tablas de hechos que formaran parte restante del Data Warehouse. Gran parte de este trabajo se hace al identificar las dimensiones, ya que las tablas de hechos son los componentes que agrupan y definen los aspectos con base a los cuales se quiere hacer el análisis dimensional.

- **Gestión de las dimensiones**: una vez definidos los datos que serán parte de las tablas de hechos y dimensiones del almacén datos, se tiene que supervisar que una vez preparada la información para poder ser insertada dentro de una dimensión, ésta sea depositada y se notifique a la respectiva tabla de hechos que dicha dimensión ha sido dada de alta.

- **Entrega final de los datos**: esta es la etapa final de todo el proceso ETL, en este paso los datos limpiados y conformados son escritos dentro de las estructuras dimensionales bajo las rutinas y arquitecturas que se definieron.

3.2.6. La estructura del Data Warehouse

Uno de los elementos más importantes a considerar en el proceso ETL es la estructura que tiene el Data Warehouse donde se guardará la información. En la actualidad la mayoría de los almacenes de datos adoptan estructuras basadas con bases de datos relacionales, en específico dos modelos; **el modelo relacional,** también llamado enfoque «Inmon» y **el modelo multidimensional,** denominado enfoque «Kimball».

En esta ultima sección del proceso ETL hablaremos de las bondades y desventajas de ambos modelos, ninguno de los dos resuelve en su totalidad los problemas de almacenamiento y recuperación de la información de un Data Warehouse. Dado que las implementaciones de almacenes de datos tienen sus particularidades, es de suma importancia tener argumentos sólidos para determinar bajo qué circunstancias es mejor aplicar cualquiera de los dos enfoques.

El modelo relacional

Este enfoque propone la aplicación del modelo relacional clásico para la definición del Data Warehouse. Agrupa la información en tablas, con atributos, tuplas y relaciones. La estructura del modelo relacional es definida a través de un lenguaje de definición de datos (DDL). Debido a la madurez del modelo, existen múltiples implementaciones comerciales de sistemas administradores de bases de datos con este modelo, algunos son: DB de IBM, Oracle DBMS y el DBMS de Teradata.

El modelo relacional emplea llaves y llaves foráneas para establecer relaciones entre sus tablas de información; emplea el lenguaje SQL para poder recuperar e insertar información. La figura 3.9 muestra un modelo clásico de base de datos relacional, con su respectivas relaciones R1,R3... R7.

Los datos dentro del modelo relacional son normalizados, lo que hace al modelo consistente y bien fundamentado, además de poder controlar la información a un nivel de granularidad muy bajo.

La gran ventaja del modelo relacional es la flexibilidad que posee, debido en gran medida a la disciplina de construcción que caracteriza este modelo. La versatilidad es la segunda gran ventaja del modelo relacional, debido a que un buen modelo relacional permite reunir la información a detalle y combinarla para poderla visualizarla de diversas maneras.

El modelo multidimensional

En el centro del enfoque multidimensional está una unión de estrella como base del modelo de datos. Puesto que el modelo emplea el enfoque de tener una tabla central y uniones de estrella, es también conocido como modelo de estrella.

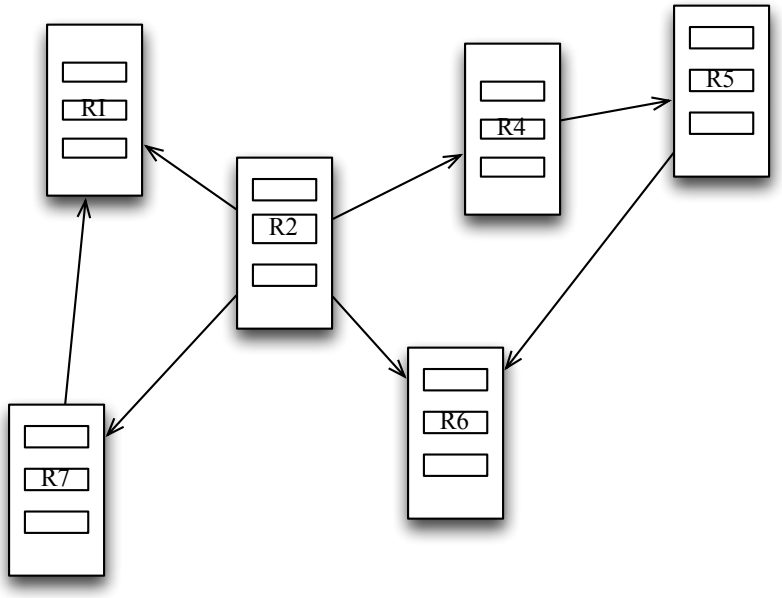

Figura 3.9: El modelo relacional

Los elementos centrales del modelo de estrella se pueden ver en la figura 3.10. En el centro del modelo está una tabla de hechos. Una tabla de hecho es una estructura que contiene múltiples ocurrencias de información. Rodeando a la tabla de hechos, tenemos tablas de dimensiones. Éstas describen aspectos importantes de la tabla de hechos, se caracterizan por tener un número reducido de ocurrencias en comparación con las tablas de hechos.

Por lo general un modelo multidimensional presenta una sola tabla de hechos rodeada de dimensiones, pero puede darse una variante del modelo, por tal motivo los sistemas multidimensionales se dividen en:

- **Estructura de estrella**: es el modelo multidimensional clásico, con una tabla de hechos rodeada por dos o más tablas de dimensiones (figura 3.10).

- **Copo de nieve**: esta variante del modelo presenta varias tablas de hechos que comparten algunas dimensiones entre sí; las dimensiones compartidas algunas veces son denominadas conformadas. En la figura 3.11 podemos observar la estructura de un modelo de copo de nieve.

La gran ventaja de los modelos multidimensionales es su eficiencia para el acceso y recuperación de información. Se especializan en resolver consultas multi tabla de forma rápida. Una correcta implementación del modelo multidimensional implica una buena comprensión de los requerimientos debido a que la estructura final es rígida y no permite hacer correcciones una vez definida.

Podemos decir que el modelo relacional tiene la ventaja de ser flexible y versátil frente a lo que ofrecen lo modelos multidimensionales, lo que les permite adaptarse a

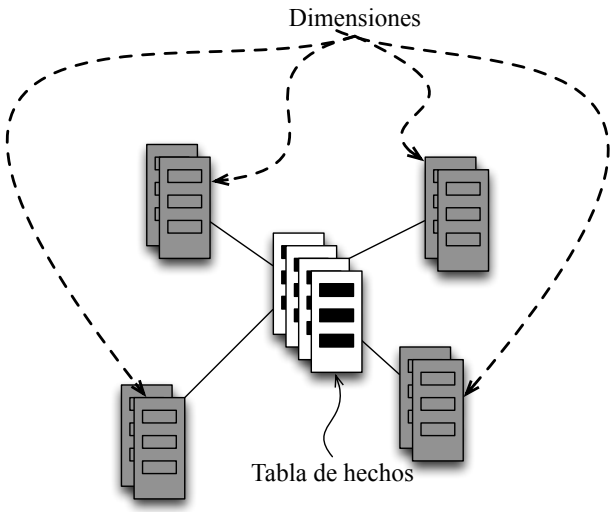

Figura 3.10: El modelo de unión de estrella

cualquier modelo de negocio y sistemas tanto transaccionales como analíticos, ademas de tener un fundamento sólido y manejar información a niveles de granularidad alta.

En contraste, los modelos multidimensionales son rígidos y cuesta trabajo idear la forma de modelar el negocio a través de ellos. Pero a diferencia del modelo relacional, se especializan en atacar problemas concretos de análisis de información, además de tener un mejor desempeño para recuperar grandes volúmenes de información. Otra característica de este tipo de modelo es que la información que almacena va dirigida a un nicho específico de una organización, a diferencia de un modelo relacional que es de un alcance más general.

3.3. Herramientas y enfoques existentes

Como pudimos observar, el proceso ETL involucra una gran variedad de elementos, proceso y tareas. Debido a la complejidad del proceso se han propuesto herramientas y enfoques tanto académicos como comerciales para poder mitigar los problemas del proceso ETL.

A continuación hacemos una reseña de los trabajos y herramientas que en últimos años se han propuesto para atacar el proceso ETL.

3.3.1. Trabajos basados en enfoques no ontológicos

En el ámbito privado, la mayoría de las herramientas ayudan en todo el ciclo de desarrollo del proceso ETL. Comienzan con la selección de las fuentes de información, el análisis de perfil de datos, la limpieza, la conformación; en algunos casos

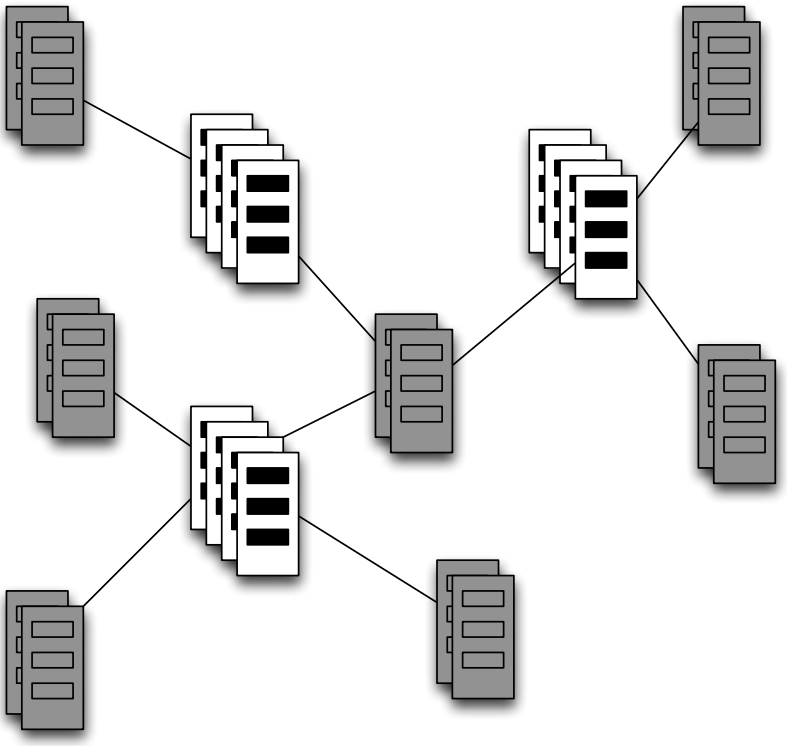

Figura 3.11: El modelo de copo de nieve

proponen mapeos y reglas de transformación para los datos extraídos y con ayuda de administradores ETL definen el linaje final de los datos.

Algunas herramientas que abordan el problema ETL son las siguientes:

Oracle Warehouse builder [43]

Es un producto que forma parte de Oracle 11g. Permite el modelado y la integración de datos bajo diversos entornos, permite realizar extracciones de alto desempeño y grandes cantidades de reutilización, además de tener características avanzadas para la administración de meta-datos. Adicional a lo anterior, esta suite ETL integra la parte de calidad de datos que permite la creación de perfiles de datos, reglas de datos (reglas de negocio) y cumplimiento de la información permitiendo generar correcciones de datos de manera automática.

También define conectores que permiten un mejor acceso e integración con aplicaciones de ERP y CRM. El producto promete convertir los datos en resultados tangibles de manera rápida y con calidad.

Pentaho [44]

Pentaho es un proyecto Open Source para la inteligencia de negocios, agrega como uno de sus módulos centrales, una herramienta para llevar el proceso ETL denominada

Pentaho Data Integration (Kettle). Pentaho Data Integration está formado por un conjunto de herramientas, cada una con un propósito específico.

- **Spoon**: es la herramienta gráfica que nos permite el diseño de las transformaciones y trabajos. Incluye opciones para pre-visualizar y probar los elementos desarrollados. Es la principal herramienta de trabajo de Pentaho Data Integration y con la que se construyen y validan procesos ETL.

- **Pan**: es la herramienta que nos permite la ejecución de las transformaciones diseñadas en spoon (bien desde un archivo o desde el repositorio). Nos permite desde la línea de comandos preparar la ejecución mediante scripts.

- **Kitchen**: similar a Pan, pero para ejecutar las tareas.

- **Carte**: es un pequeño servidor web que permite la ejecución remota de transformaciones y tareas.

SQLServer [45]

Es un componente que Microsoft incorpora dentro de su motor de base de datos SQL Server, permite realizar el proceso ETL mediante la definición de flujos de trabajo. Define tareas para procesamiento de cubos de análisis de servicios.

Ofrece una herramienta visual en forma de asistente para definir datos desde diversos orígenes sin que se permita la modificación de los mismos. Comparada con las herramientas anteriores no tiene compatibilidad con muchas fuentes de información, no es tan flexible para poder soportar transformaciones o procesos de calidad de información.

Adicional a las herramientas que se presentaron brevemente, tenemos algunas otras suites especializadas en ETL como las siguientes: DataStage [46], Informática [47], SAS ETL[48].

Dentro del ámbito privado, algunas veces se lleva a cabo el proceso sin hacer uso de soluciones especializadas en ETL. Cuando se hace el proceso de esta forma, por lo general se definen rutinas SQL de carga, parseo de archivos y elaboración y ejecución de secuencias de instrucciones (SCRIPTS). Esto hace el proceso caótico y muy desorganizado debido a que no hay control de los meta-datos, de los mapeos, de las relaciones y del linaje de datos; lo que produce que muchos proyectos de esa naturaleza no entreguen buenos resultados

3.3.2. Trabajos basados en enfoques ontológicos

El problema central del proceso ETL es el de interoperabilidad a diversos niveles, este misma problemática es la que presenta la Web Semántica y es debido a esto que se han propuesto enfoques basados en herramientas que combaten problemas de la Web Semántica para poder solucionar problemáticas propias del proceso ETL.

En [49] se propone una forma de abordar diversos problemas que surgen en la construcción de un Data Warehouse con la ayuda de ontologías.

Longbing Cao[50] descompone el proceso de construcción de un Data Warehouse en diversas etapas, cada una de ellas con problemáticas propias; para así atacar los problemas de forma concreta.

Por otra parte propone una arquitectura en capas con diversos elementos definidos en su ingeniería ontológica.

El autor propone los siguientes mecanismos para poder hacer un correcto proceso para la elaboración de un Data Warehouse basado en ontologías:

- Construcción de perfiles ontológicos para problemas de un dominio muy específico.
- Definición de compromisos ontológicos y relaciones semánticas.
- Representación ontológica de los diversos actores involucrados en el proceso de construcción del Data Warehouse.
- Agregación y transformación de elementos de un dominio y en dominios diversos.

Por otra parte [50], pone de manifiesto algunas de las problemáticas más importantes del proceso de ETL: lo inflexible y poco adaptable que es el proceso y el problema de interoperabilidad entre aplicaciones software que se deriva del mismo. Para poder combatir la problemática, el autor propone una solución híbrida entre el enfoque clásico o estructurado y un enfoque basado en ontologías que aborda los siguientes puntos:

- Definir de manera formal los objetivos que se persiguen al comenzar el proceso.
- Integración semántica de las diversas fuentes de información, tomando como base esquemas ontológicos.
- Interacción a nivel semántico entre las diversas capas que componen el proceso de BI; logrando esto a través del manejo de esquemas ontológicos.

Una propuesta novedosa se puede encontrar en [51], donde trata la extracción de información de diversas fuentes de datos y su posterior carga en un almacén unificando.

Para cumplir sus objetivos se auxilia de herramientas especializadas en la extracción de información de cada una de las diversas fuentes de información (PLN, OCR, etc). Cabe resaltar que este enfoque define el uso de ontologías definidas para lograr una correcta integración de la información. De esta forma el proceso seguido es el siguiente:

- Creación de la ontología.
- Extracción de información de las diversas fuentes de datos.

- Con ayuda de la ontología e intervención humana y la información extraída se va poblando la ontología.

Como último trabajo hablaremos del enfoque presentado por Hilary Cheng [52], nos da una panorámica general de algunos de los problemas que podemos encontrar en el proceso de ETL. Se centra en resolver un caso de estudio muy específico, por lo que el modelo pierde un poco de generalidad, sin embargo se pueden tomar partes a nivel arquitectural que pueden ayudar, con respecto al uso de ontologías. Este trabajo de investigación las utiliza para tres puntos clave:

- En la capa de transformación de datos, para poder agrupar la información con base a su significado.
- Al término del proceso para poder documentar el proceso que se lleva a cabo
- Para poder administrar el conocimiento aprendido.

Existen más enfoques dentro del área de la investigación que atacan el proceso ETL pero queda fuera del alcance de esta tesis su explicación a detalle, para un mayor detalle se recomienda ir a las referencias.

Capítulo 4

El marco de trabajo Onto-ETL

Un marco de trabajo o framework es definido como una arquitectura conceptual y tecnológica que posibilita la realización de proyectos de software de manera más simple y ágil. Un marco de trabajo abstrae los procesos y componentes generales de una serie de problemas semejantes, para con base a ello proponer una metodología y herramientas que hagan mucho más fácil el desarrollo de aplicaciones que resuelvan dichos problemas.

Onto-ETL es producto del análisis de los problemas que enfrenta el proceso ETL actual [3], las debilidades y deficiencias de los enfoques basados en ontologías para la realización del proceso ETL [52] y de las mejores prácticas que implementan algunos enfoques para mitigar la interoperabilidad del proceso ETL [49].

A diferencia de muchos enfoques basados en ontologías para asistir el proceso ETL, Onto-ETL aborda la construcción y gestión de la ontología que dará soporte al proceso ETL. con base a la experiencia adquirida durante la investigación, pudimos observar que el fracaso de muchos enfoques basados en ontologías es producido por dejar de lado la correcta construcción de la ontología

Un Data Warehouse contiene la información de toda una organización, por tal motivo hay un sin numero de elementos: términos, relaciones, reglas de negocio, etc que debemos de comprender para poder integrar la información y transformarla cuando se construye y deposita la información dentro del mismo.

Es imposible para el cuerpo técnico encargado de la elaboración de un Data Warehouse dominar los conceptos, relaciones y reglas de gobiernan la información de dicho almacén. Pero por otra parte hay gente dentro de la organización con el conocimiento especifico y experiencia del domino de conocimiento de la información que se trata durante el proceso ETL. Ese conocimiento y experiencia como observaremos es susceptible de ser capturado y aprovechado para poder construir un repositorio de conocimiento (ontología maestra). Ahora que el conocimiento plasmado en la ontología proviene de la gente con experiencia en el dominio, podemos afirmar que es fiable y de calidad y sin duda alguna podrá ahora ser usado como base para la solución de muchos problemas. En nuestro caso específico, este conocimiento nos servirá para poder realizar un proceso ETL con un mínimo de intervención humana y con una gestión adecuada de los datos y meta-datos que intervienen.

4.1. Arquitectura general del Marco de trabajo propuesto

El principio de funcionamiento de Onto-ETL descansa sobre cuatro principios básicos:

- El proceso ETL se realiza sobre un dominio de conocimiento claramente delimitado.
- El dominio de conocimiento asociado al proceso ETL queda perfectamente modelado mediante reglas de negocio.
- La Gestión de Conocimiento permite modelar reglas de negocio mediante ontologías.
- El proceso ETL puede ser automatizado con ayuda de ontologías de dominio.

Onto-ETL aborda la problemática con dos enfoques principales: la Gestión de Conocimiento con la ayuda de ontologías y la implementación de un nuevo proceso ETL con la colaboración y reutilización del conocimiento obtenido.

La solución a la problemática se logró desarrollando los tres elementos principales del marco de trabajo:

- Una metodología para realizar la Gestión de Conocimiento que ayudara a la captura y Gestión de Conocimiento de expertos.
- Una metodología para realizar el proceso ETL con ayuda del conocimiento almacenado.
- Una biblioteca de componentes software que ayudara a las metodologías propuestas a realizar sus diferentes tareas.

En las metodologías se definirán los actores, herramientas, componentes software, procesos y tareas a seguir para resolver las dos problemáticas.

En cuanto a la biblioteca de Software se expondrán los elementos de la biblioteca de software desarrollada; su estructura, funcionalidad y la participación de los mismos en las metodologías anteriormente señaladas.

En la figura 4.1 mostramos las dos problemáticas que ataca Onto-ETL. La gestión de conocimiento para la captura de las reglas de negocio y la generación una ontología que es el modelo del conocimiento de los elementos que intervienen en le proceso ETL y por otra parte la interoperabilidad que impera entre las fuentes de datos origen del proceso ETL.

En la figura 4.2, mostramos en un diagrama de casos de uso los actores que intervienen en Onto-ETL y las principales funciones que desempeñan dentro del marco de trabajo, el administrador ETL es el encargado de llevar el proceso ETL, sus labores principales son la configuración inicial del proceso ETL, así como la supervisión de la creación del modelo lógico y físico de datos.

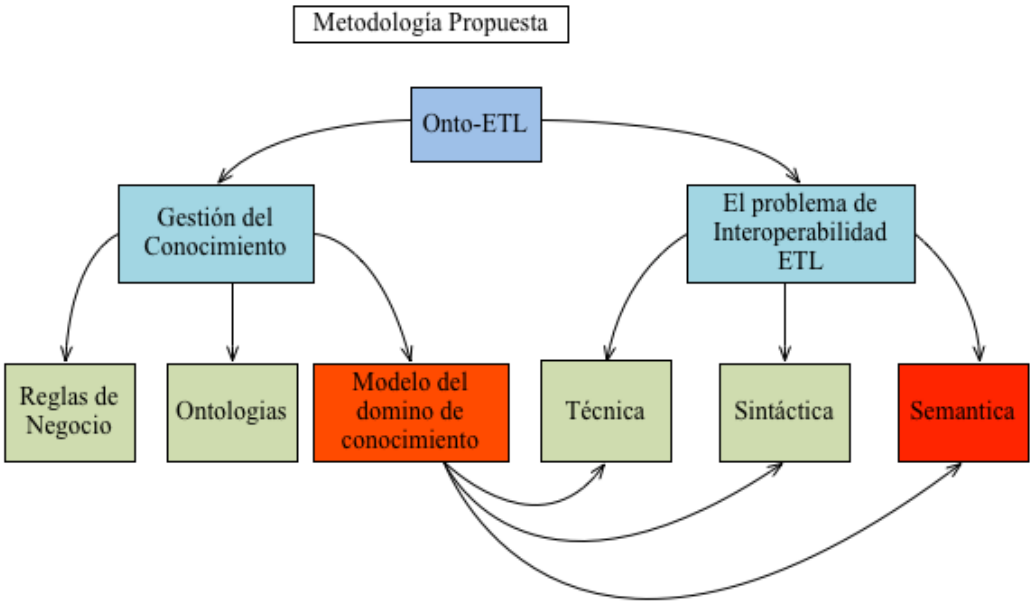

Figura 4.1: Elementos centrales Onto-ETL

El administrador o gestor del conocimiento tiene como responsabilidades, la definición del modelo ontológico, la codificación correcta del conocimiento capturado, la construcción de la ontología y la administración de la ontología.

Por su parte el experto del dominio, representa para Onto-ETL el facilitador del conocimiento, ademas de alguien que colabora con la administración de la ontología, aportando su conocimiento cuando ese sea requerido.

En las siguientes secciones ampliaremos la descripción de cada uno de los elementos que describimos de forma general en esta vista general del marco de trabajo.

4.2. La Gestión de Conocimiento

El conocimiento dentro de una organización reside tanto en documentos como en la mente del capital humano de la misma y en los sistemas de información de la misma. El objetivo de la Gestión de Conocimiento es hacer fluir y favorecer el intercambio de conocimiento entre estos elementos, para enriquecer el potencial de una organización y resolver de una mejor forma los problemas.

Desde sus orígenes, la Gestión de Conocimiento ha buscado agrupar información relevante del capital humano dentro de una organización, para crear repositorios de conocimiento que puedan ser consultados para poder apoyar a la organización en la toma de decisiones.

En el caso de Onto-ETL, la Gestión de Conocimiento tiene por tareas principales:

- La conversión de datos en conocimiento.
- La conversión de conocimiento individual en conocimiento accesible.

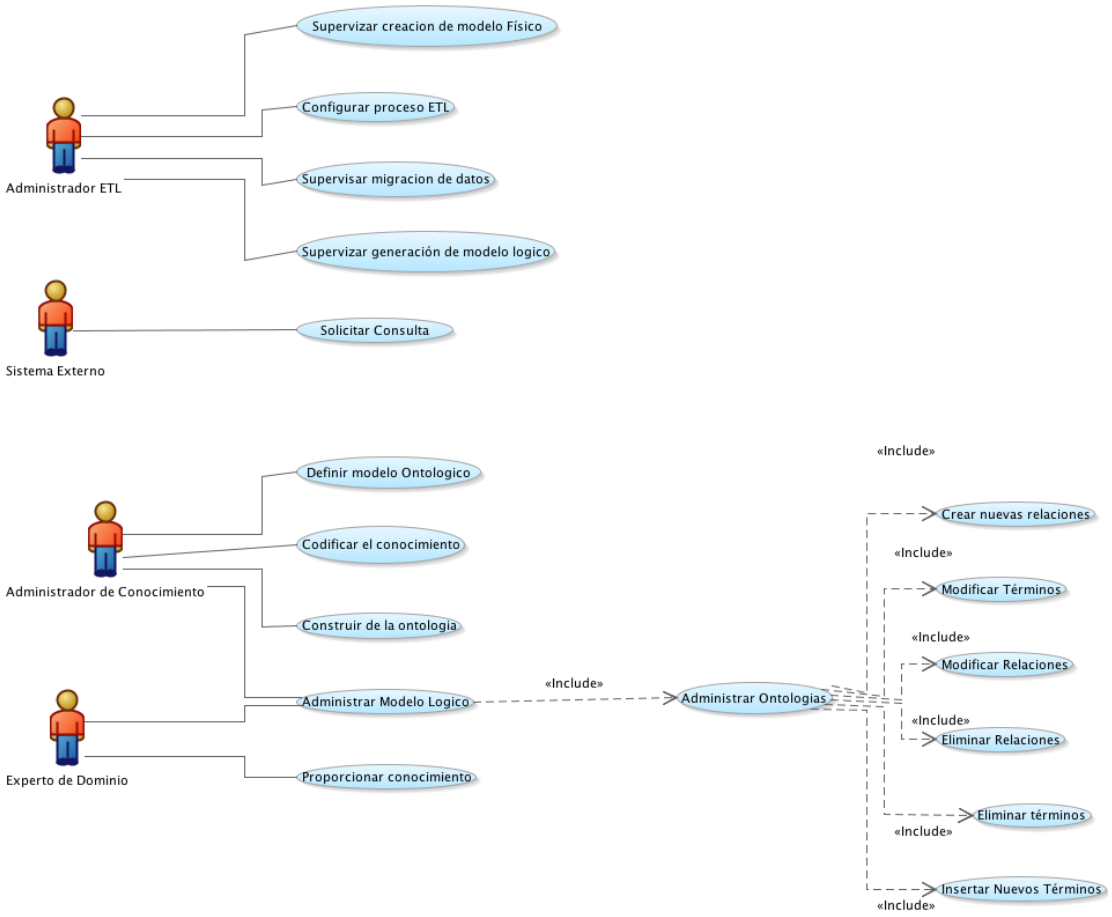

Figura 4.2: Casos de uso Onto-ETL

- La comunicación de información entre individuos y sistemas de datos.
- Almacenamiento de conocimiento útil para la organización.

Dentro de la Gestión de Conocimiento existen dos actores principales, los agentes que generan el conocimiento a ser almacenado y los que lo consumirán. Onto-ETL tiene como agentes generadores de conocimiento, expertos en el área sobre la cual se implementa el proceso ETL, por ejemplo si el proceso se va a efectuar sobre un esquema financiero, los actores serán personas de la organización expertas en ese dominio, de esta forma se puede asegurar que el conocimiento adquirido es el adecuado; en cuanto al agente que realizara el consumo de información, podemos citar al proceso ETL mismo ya que el tomara el conocimiento guardado en las ontologías para poder automatizar algunas partes del proceso y simplificar el trabajo de las personas encargadas del proceso.

En lo concerniente a las tecnologías del conocimiento, las ontologías han ganado la reputación de ser el mejor medio para poder representar el conocimiento [53, 54, 32].

Por tal motivo Onto-ETL define como medio de representación del conocimiento a las ontologías; dada la relevancia que han adquirido las ontologías, se han propuesto una gran cantidad de metodología para la creación de las mismas, algunos enfoques proponen la construcción de forma individual colectiva, bien comenzando sin tener elemento alguno o con ingeniería inversa.

En posteriores secciones definiremos el modelo de conocimiento y sus elementos al igual que las metodologías sobre las cuales Onto-ETL se apoya para la creación y mantenimiento de la ontología.

4.2.1. Descripción y elementos principales

La presente metodología tiene como objetivo central definir la forma los elementos, tareas y elementos necesarios para poder realizar una adecuada Gestión de Conocimiento, esto incluye la captura del conocimiento de los expertos, su representación en ontologías, su gestión y su consulta para resolución de problemas futuros.

De manera especifica se trabajara almacenando el conocimiento de tipo factual, es decir relaciones de hechos, donde lo importante será definir un vocabularios de dominós específicos: términos y sus relaciones, dejando de lado elementos como axiomas e inferencias, el objetivo es almacenar conocimiento que ayude a clasificar los meta-datos del proceso ETL forma autónoma.

Como objetivos específicos tenemos:

- Definir el modelo ontológico, entorno de trabajo y herramientas sobre las que descansara la Gestión de Conocimiento.
- Definir un mecanismo para una realizar la captura del conocimiento de forma sencilla.
- Definir el proceso para transformar el conocimiento a fin de organizarlo y almacenarlo en ontologías.
- Definir mecanismos de mantenimiento, acceso y recuperación del conocimiento a fin de que este pueda brindar beneficios y se pueda compartir.

El siguiente punto a definir en la metodología son los roles que tendrán relación directa con las tareas; Onto-ETL define los siguientes roles:

- **Expertos en el dominio**: son las personas de la organización de las cuales se ha de extraer el conocimiento que formara parte de la ontología, preferentemente debe de ser persona con facilidades para expresar de forma simple sus ideas y conceptos, para así poder hacer el proceso mas fácil.
- **Administradores del conocimiento**: son las personas encargadas de verificar que el conocimiento capturado queda correctamente modelado en las ontologías, así mismo supervisan los proceso de mantenimiento de la ontología (inserción, eliminación y consulta de información).

Formalización del modelo ontológico

Para hacer una correcta selección del tipo de ontología que se va a implementar, se tienen que conocer lo objetivos y alcances de la aplicación de las misma. En el caso en particular de Onto-ETL, el conocimiento que se quiere modelar corresponde únicamente a la definición de términos y las reglas que los asocian dentro de un dominio especifico, es decir la ontología será usada con fines taxonómicos solamente.

Como primer punto de la metodología definiremos los supuestos y bases que tomamos en cuenta para la definición del modelo ontológico, ya que en la actualidad existen diversos enfoques de la definición de ontología y de sus diversos elementos y relaciones.

Onto-ETL toma como base el concepto de ontología definido por [27], esta sostiene que «*Una ontología es un especificación de una conceptualización de un dominio de conocimiento*»; por lo anterior el esquema o modelo ontológico tendrá las siguientes características:

- Las ontologías son representadas mediante jerarquías múltiples de dominio restringido (MHRD), por lo que los conceptos tendrán las siguientes características:
  - Están definidos por un conjunto de atributos, pero no se considera la presencia de axiomas entre atributos.
  - Existen relaciones taxonómicas entre los conceptos. ie existe herencia múltiple.
  - Existen relaciones mereológicas entre conceptos
- El esquema usado en este trabajo incluye axiomas estructurales, que se manifiestan en las relaciones *concepto tiene atributo*, *concepto A es una parte del concepto B*.
- Existirán relaciones mereológicas entre conceptos, esta se define a través de la relación part-of y sus propiedades. Hay diversas teorías para implementar esta relación dentro de un esquema ontológico, pero para el alcances de este trabajo, esta relación se acota de tal forma que no permite la transitividad, de igual forma la relación será irreflexiva (nada es parte de sí mismo) y asimétricas (si X es parte de Y, entonces Y no es parte de X). Existen algunas otras propiedades que puede tener esta relación (p. ej., consistencia, superposición y disyunción), pero nos concretaremos a implementar las que señalamos.
- Para poder garantizar que los elementos de la estructura ontológica están organizados en jerarquías múltiples, con base a relaciones es-un (is-a) o parte-de (part-of) se emplearon las funciones ilustradas en el cuadro 4.1:

Complementario a la definición del modelo ontológico que hemos tratado, Onto-ETL define una ontología y sus conceptos se pueden expresar de la siguiente forma:

Una ontología O(t) puede ser vista como una tercia <C,R,P>, donde:

Función	Descripción informal	Descripción Formal
TMHRD(t)	Calcula el conjunto de conceptos relacionados taxonómicamente.	[Cardinal(TMHRD(t)) ≥ 2] $\Longleftrightarrow$ [Para todo $c_i(t) \in$ TMHRD(t) existe $c_j(t)$ $\in$TMHRD(t) tal que IS-A($c_i(t)$, $c_j(t)$) $\vee$ IS-A($c_j(t)$, $c_i(t)$)] Donde IS-A(a, b) significa que '*a es un tipo de b*'.
M^2HRD(t)	Calcula el conjunto de conceptos relacionados mereológicamente.	[Cardinal(M^2HRD(t)) ≥ 2] $\Longleftrightarrow$ [para todo $c_i(t) \in M^2$HRD(t) existe $c_j(t) \in$ M^2HRD(t) tal que PART-OF($c_i(t)$, $c_j(t)$) $\vee$PART-OF($c_j(t)$, $c_i(t)$)] Donde PART-OF(a, b) significa que '*a es una parte de b*'.
PMHRD(t)	Calcula el conjunto de conceptos relacionados taxonómica o mereológicamente.	TMHRD(t) $\cup$ M^2HRD(t).
T-Padres(t)	Calcula los padres taxonómicos de un concepto.	$\{c_j(t) \in$ TMHRD(t) tal que IS-A($c_i(t)$, $c_j(t)$)$\}$. Siendo $c_i(t)$ un concepto miembro de TMHRD(t)
T-Hijos(t)	Calcula los hijos taxonómicos de un concepto.	$\{ck(t) \in$ TMHRD(t) tal que IS-A($c_k(t)$, $c_i(t)$)$\}$ Siendo $c_i(t)$ un concepto miembro de TMHRD(t)
M-Padres($c_i(t)$)	Calcula los padres mereológicos de un concepto.	$\{c_j(t) \in$ M2HRD(t) tal que PART-OF($c_i(t)$, $c_j(t)$)$\}$. Siendo $c_i(t)$ miembro de M^2HRD(t)
M-Hijos($c_i(t)$)	Calcula los hijos mereológicos de un concepto.	$\{ck(t) \in$ M2HRD(t) tal que PART-OF($c_k(t)$, $c_i(t)$)$\}$ Siendo $c_i(t)$ miembro de M^2HRD(t)
SPE($c_i(t)$)	Calcula los atributos específicos de un concepto.	
INH-T($c_i(t)$,$C_{ta}(t)$)	Calcula los atributos heredados de un concepto $c_i(t)$ de $C_{ta}(t)$	INH-T($c_i(t)$,$C_{ta}(t)$) = $\cup$SPE($c_j(t)$) Siendo $c_i(t)$ miembro de PMHRD(t) y $C_{ta}(t)$ un subconjunto de T-padres($c_i(t)$).
ATT($c_i(t)$)	Calcula todos los atributos (heredados y específicos) de un concepto.	ATT($c_i(t)$) = INH-T($c_i(t)$,T-Padres($c_i(t)$))$\cup$SPE($c_i(t)$) Siendo $c_i(t)$ un concepto miembro de PMHRD(t)

Cuadro 4.1: Funciones del modelo ontológico de Onto-ETL

- C = PHMRD(t), i.e. un conjunto no vacío de conceptos.
- R = {⊘, IS-A, PART-OF}, i.e., el conjunto de relaciones que se pueden establecer entre los conceptos.
- P = CxC $\Longrightarrow$R. i.e., una función que establece las relaciones entre dos conceptos.

Para fines de comprensión, el modelo ontológico empleado por Onto-ETL puede ser visualizado de cualquiera de las dos formas, en secciones posteriores definiremos como se implementan el modelo ontológico, como se administra y finalmente como se emplea durante el proceso ETL.

El ciclo de desarrollo de las ontologías

Hemos definido el modelo ontológico, sus elementos y relaciones, ahora tocaremos el punto del ciclo de desarrollo que seguirá la construcción de la ontología, ya en el capitulo 2 expusimos algunas metodología y ciclos para el desarrollo de ontologías, el ciclo que propone Onto-ETL esta basado en las mejores practicas y pasos de las metodologías revisadas [28, 16, 29, 30, 31, 15, 32].

Los puntos que ya se han definido por si mismos constituyen parte del ciclo de desarrollo, ya que fueron propuestos pensando en la metodología que habría de seguirse para la construcción de la ontología, hay que señalar que la ontología se construirá desde cero, aunque posteriormente se podrá adicionar información. Aclarados los supuestos iniciales, describimos a continuación las fases del ciclo de desarrollo de la ontología:

1. **Identificar el propósito y dominio de conocimiento de la ontología**: como ya definimos, las ontologías que define Onto-ETL son de dominio, por tal motivo el primer paso es identificar el dominio de conocimiento que se va a modelar con la ontología y el propósito concreto que ha de cumplir. Como consecuencia de esto, se deben de elegir los expertos del dominio adecuados, para poder extraer información y conocimiento que colabore con el propósito de la ontología. Hasta este momento hemos ya pasado esta fase de la metodología, con la definición del esquema ontológico y el propósito que persiguen las ontologías dentro de Onto-ETL.

2. **Capturar los términos y relaciones con ayuda de herramientas software**: el siguiente punto consiste en capturar los términos y relaciones (el conocimiento del experto) con la ayuda de herramientas software, en el caso concreto Onto-ETL, propone un enfoque basado en capturar el conocimiento por medio de su representación visual en grafos, donde los vértices representaran conceptos con sus respectivas propiedades y las aristas las relaciones que guardan dichos ejes. En el siguiente apartado de reglas de negocio y su captura ahondaremos mas sobre este modelo propuesto.

3. **Codificar los elementos capturados**: una ves capturado el conocimiento a través de los grafos, se procederá a codificar, es decir transformar los elementos en términos que formaran parte de la ontología, como objetos, propiedades y relaciones. Para este punto entra en juego el apoyo de un gestor del conocimiento a fin de que valide y corrobore las transformaciones que la herramienta produce como resultado; de esta forma se garantiza que el conocimiento capturado es correcto y de calidad.

4. **Construir la ontología sobre un lenguaje ontológico**: con la codificación de los elementos se construirá la ontología sobre un lenguaje ontológico, a fin de poder guardar de forma permanente el conocimiento codificado, en la sección de la construcción de la ontología, definiremos el lenguaje especifico de construcción y las herramientas necesarias para poder llevarlo a cabo.

5. **Administrar la ontología construida**: una ves construida la ontología pasa ahora a ser administrada, para poder realizar esta tarea, se propone la intervención de un gestor del conocimiento, el cual será el encargado de adicionar nuevo conocimiento, eliminar conocimiento y modificar el conocimiento contenido dentro de la ontología.

En los siguientes apartados de esta sección detallaremos las fases de captura, codificación, construcción y administración de la ontología.

4.2.2. Las reglas de negocio (conocimiento) y su captura

Las reglas de negocio son la materia prima que Onto-ETL necesita para la captura de conocimiento, puesto que dentro de una organización las reglas de negocio son las responsables de modelar los dominios de conocimiento de una organización y es precisamente este conocimiento que se quiere plasmar dentro de la ontología.

Una regla de negocio es una descripción de políticas, normas, operaciones, definiciones y restricciones presentes dentro de una organización.

El dominio de aplicación de una organización queda definido a través de las reglas de negocio que esta define [18], Onto-ETL se basa en este principio fundamental, y propone capturar el conocimiento contenido en las reglas de negocio, para poder almacenar el conocimiento de dominio de una organización y poder con ayuda de este resolver problemas.

Onto-ETL se basa en técnicas de ontología de conocimiento modernas, para poder capturar el conocimiento contenido en las reglas de negocio y guardarlo dentro de una ontología. Dentro de una organización, las únicas personas con el manejo y conocimiento de las reglas de negocio, son personas expertas de dominio. Por tal motivo, Onto-ETL define que la información contenida dentro de la ontología, será resultado directo de la captura del conocimiento de los expertos del dominio; algunas características de las reglas de negocio que favorecen su captura y almacenamiento dentro de una ontología son las siguientes:

- Declarativas.
- Atómicas.
- Construidas de manera independiente y distinta.
- Expresadas en lenguaje natural.
- Orientadas al negocio.
- Pueden ser expresadas en un lenguaje formal.

Como se puede observar, el ultimo punto señala que las reglas de negocio pueden ser expresadas en un lenguaje forma, al ser las ontologías una herramienta para definir especificaciones formales, tienen la capacidad de poder modelar y almacenar las reglas de negocio.

Algunos ejemplos de reglas de negocio pueden ser los siguientes:

- Las facturas serán pagadas en dólares
- Los usuarios tipo «A» tienen egresos mayores a X
- Las medidas de las piezas están en CM
- Un producto se define por las propiedades
- Un usuario tiene asociado un numero de seguridad único
- Un usuario trabaja de x a y
- El almacén se surte de productos cada 20 días.

Como podemos ver las reglas de negocio definen relaciones entre elementos de un dominio especifico, estas relaciones contienen una gran cantidad de información debido a que modelan el dominio de conocimiento, este conocimiento es útil, sin embargo como definimos en secciones anteriores, Onto-ETL se centra en extraer conocimiento muy puntual y especifico, en concreto solo se requiere saber los elementos del dominio de conocimiento, sus propiedades y sus relaciones.

Para fines mas simples, Onto-ETL propone la captura de las reglas de negocio mediante un sencillo procedimiento:

1. Se tendrá una lluvia de ideas con el experto del dominio, a fin de obtener todos los conceptos que estarán inmiscuidos en el dominio de conocimiento que se quiere modelar.
2. Se definirá con ayuda del experto, las propiedades de todos y cada uno de los conceptos, tales como nombre, tipo de datos, dominios y rango de valores y relaciones.

3. Se definirán las relaciones existentes entre los diversos elementos del dominio de conocimiento, propiamente dicho estas relaciones serán las reglas de negocio, la forma de definir las relaciones, será por medio de asociaciones binarias entre elementos asta cubrir el total de las relaciones del dominio.

4. Una vez obtenidos los elementos, propiedades y relaciones, se revisaran con ayuda del experto del dominio, ya que con estos ingredientes, se procederá a construir la ontología.

Onto-ETL ha definido el procedimiento general para la captura de las reglas de negocio y su mapeo a elementos base para la construcción de la ontología, en la siguiente sección se proponen las herramientas y medios necesarios para poder hacer la implementación de la ontología con base a los elementos extraídos.

4.2.3. Construcción de la ontología.

En la actualidad, las ontologías son implementabas bajo lo que se denomina lenguajes ontológicos, en el capitulo 2 se hablo de algunos de ellos [34, 33, 36, 23], el elemento principal a considerar al elegir un lenguaje para la implementación de una ontología, es el nivel de expresividad, es decir el poder que tiene un lenguaje para poder modelar los conceptos y relaciones de un dominio de conocimiento. Tomando este punto en consideración, la definición del modelo ontológico y las necesidades de la ontología que se construirá y después del trabajo de investigación relacionado Onto-ETL propone como lenguaje de implementación de ontologías a OWL (Web Ontology Language) [16].

OWL es un lenguaje para la definición de ontologías que se apoya en RDF [34], fue propuesto por la W3C para la definición de ontologías, como un proyecto que apoya la construcción de la Web Semántica.

La construcción de la ontología se realizara tomando como elementos de partida los productos obtenidos en la parte de captura de las reglas de negocio, se definirá el mapeo correspondiente de los diversos componentes a su definición dentro de la ontología.

Como se observa en el paso 3 del proceso de extracción de las reglas de negocio, la relaciones se definen mediante asociaciones binarias, esto facilita el mapeo de estas estructuras a su correspondiente representación dentro de la ontología, debido a que las relaciones se hacen de la misma forma, siendo un mapeo automático entre los diversos elementos.

A continuación mostramos las herramientas que se emplean para la construcción de la ontología.

Herramientas y entorno de trabajo

Onto-ETL adopta OWL (Web Ontology Language) como el lenguaje para la creación de la ontología, que a su ves esta definido con ayuda de RDF, RDF es un lenguaje

para la especificación de meta-datos. identifica los recursos usando los Uniform Resource Identifiers o URIs, también describe los recursos en términos de propiedades simples y valores. Una descripción RDF es un conjunto de proposiciones simples (también llamadas sentencias o declaraciones), una proposición se conoce también como una tripleta, porque está compuesta de 3 cosas: un sujeto, un predicado y un objeto:

- **El sujeto**: es la cosa de la cual se habla en una declaración.
- **Predicado**: la propiedad o característica del sujeto que se esta describiendo.
- **Objeto**: la parte que identifica el valor que toma la propiedad.

En RDF tanto los sujetos, como las propiedades y los objetos, son recursos, a continuación mostramos dos ejemplos de una construcción básica en OWL:

- **http://www.ejemplo.com/index.html** *tiene un* **creation-date** *cuyo valor es* **marzo 12, 2005**
- **http://www.ejemplo.com/index.html** *tiene un* **language** *cuyo valor es* **Español.**

Las cosas se describen mediante los valores que toman sus propiedades y los recursos pueden describirse mediante declaraciones. Los objetos declarados en sentencias o declaraciones RDF pueden ser URIrefs, o valores constantes llamados literales (literals).

Debido a que las relaciones que define OWL con de carácter binario, todas y cada una de las relaciones definidas en el proceso de captura de reglas de negocio, pasaran de forma transparente y automática a su implementación dentro de OWL.

Para poder definir las ontologías mediante el lenguaje OWL, se emplea la herramienta de trabajo Protege, una aplicación visual para la definición y construcción de ontologías con base a los términos y relaciones definidas.

Algunos elementos extras que se pueden considerar para el desarrollo de componentes relacionados con las ontologías son:

- **JENA**: es un marco de trabajo para la creación y manipulación de ontologías desde java, ademas de definir mecanismos de persistencia de ontologías sobre bases de datos relacionales.
- **SPARQL**: es un lenguaje de recuperación basado en RDF; su nombre es un acrónimo del inglés SPARQL Protocol and RDF Query Language. Se trata de una recomendación para crear un lenguaje de consulta dentro de la Web Semántica. Es de utilidad para la recuperación y organización de información dentro de una ontología.

Una ves definidos los elementos que ayudaran a crear la ontología y ya habiendo realizado su desarrollo, solo queda pasar a ver el proceso de mantenimiento que seguirá, para poder tener información al día

4.2.4. Publicación y mantenimiento de la ontología.

El primer punto una ves realizada la creación de la ontología, es su publicación, esto una ves revisada por parte de los gestores del conocimiento y cotejada en colaboración con los expertos del dominio; al publicar la ontología, esta queda a disposición de todos la ontología con la finalidad de poder ser consultada y enriquecida con mas conocimiento; algunas tareas que se definen para poder llevar a cabo el mantenimiento de la ontologías son definidas en los siguientes diagramas de actividad:

En el diagrama 4.3 podemos ver el proceso de la integración del conocimiento contenido en una ontología existente a nuestra ontología maestra.

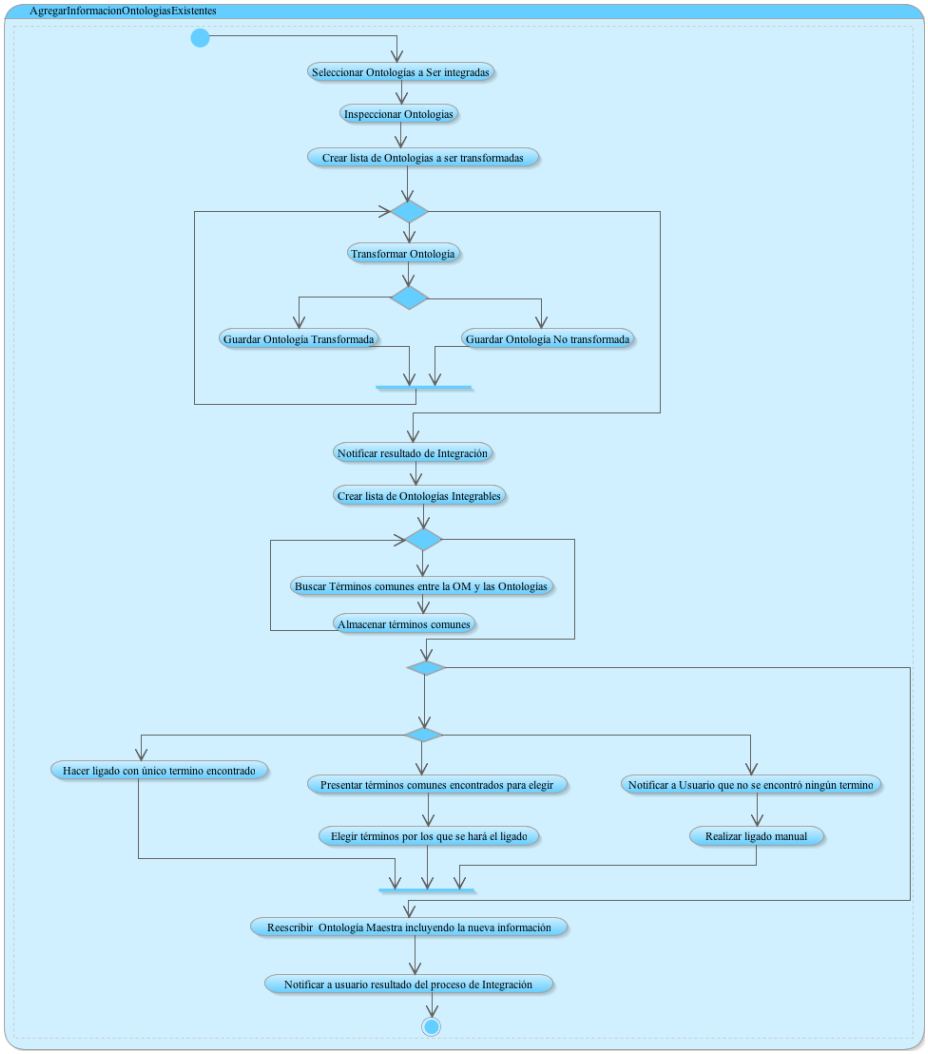

Figura 4.3: Adición de ontologías existentes a la ontología

4.3. El proceso ETL basado en ontologías

El proceso ETL que propone Onto-ETL basa su idea en la clasificación y categorizaron automática de los meta-datos de las fuentes de información que intervienen en el proceso ETL con ayuda de ontologías, así mismo propone con ayuda de la ontología: mapeos y transformaciones a nivel sintáctico y semántico de las fuentes origen al almacén destino, un modelo del Data Warehouse basado en ontologías y la colaboración para la generación de rutinas de extracción, inserción y creación sobre los repositorios de datos que intervienen en el proceso ETL; de forma adicional define un metodología para poder gestionar el linaje de los datos a lo largo de todo el proceso ETL.

El corazón la metodología que propone Onto-ETL para llevar a cabo el proceso ETL, descansa en la idea de *la ontología maestra como entidad central de la definición de los términos y relaciones presentes dentro del proceso ETL y sus diversas fuentes de información.* **La ontología maestra** como le llamaremos de aquí en adelante, es la ontología definida en la primer metodología de Onto-ETL.

La metodología propuesta por Onto-ETL para llevar a cabo el proceso ETL se muestra en la figura 4.4, y sigue los pasos siguientes:

- El primer proceso es el responsable de la extracción de la meta información de las diferentes fuentes de datos.
- La meta información es buscada dentro de la ontología maestra a fin de ligar la información de las diversas fuentes de datos a su definición formal dentro de la ontología maestra.
- El modelo lógico de datos es creado basado en la ontología maestra. El modelo generado es presentado entonces al administrador ETL, para que este haga los ajustes y correcciones pertinentes.
- El administrador ETL selecciona el modelo físico sobre el cual se generara la carga de información.
- Los mapeos del modelo lógico al modelo físico son generados automáticamente, incluidas las transformaciones.
- Las rutinas de extracción y de inserción de datos son generadas y el Data Warehouse es poblado.

4.3.1. Descripción y elementos principales

La ontología maestra contiene el conocimiento especifico del dominio de aplicación del Data Warehouse, lo que garantiza que dentro del proceso ETL que se lleva a cabo se tendrá el conocimiento de la gran mayoría de términos y de relaciones que se

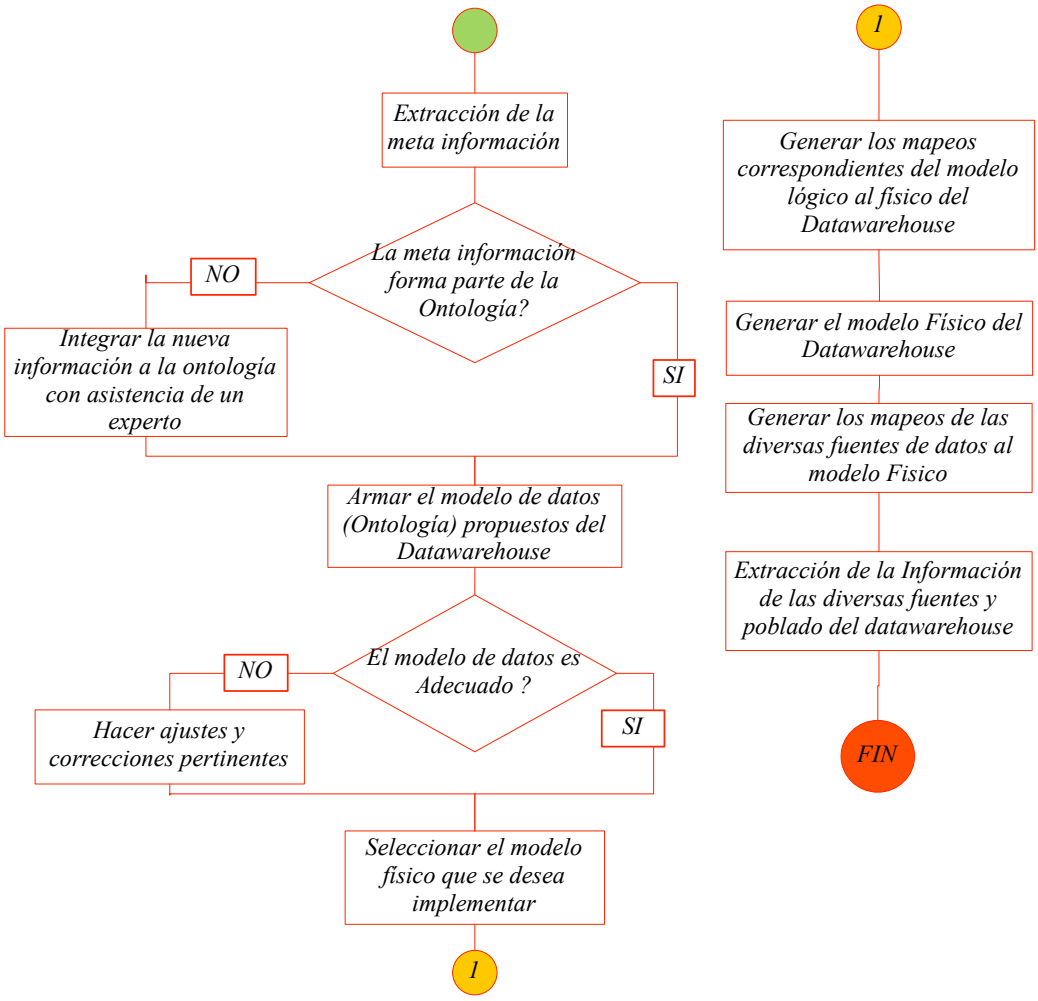

Figura 4.4: El proceso ETL

encuentren en las diversas fuentes de información; así mismo se conocerán su propiedades y meta-datos, lo que simplificara y podrá ayudar al proceso en general como se detallara en los párrafos siguiente. Tomando como base estas aclaraciones, podemos formular los objetivos de esta metodología.

Onto-ETL plantea como puntos centrales de esta metodología, definir los lineamientos generales para poder realizar un proceso ETL con la asistencia del conocimiento contenido en la ontología maestra, ademas de:

- Definir un mecanismo de extracción estandarizado para fuentes de información diversas.
- Definir mecanismos de Caracterización y clasificación de los meta-datos
- Definir mecanismos para la creación y poblado de un Data Warehouse
- Definir mecanismos para el linaje de los datos.

4.3.2. Las fuentes de información

Como revisamos en el capitulo 3, dentro del proceso ETL existe una gran cantidad fuentes de información distintas, Onto-ETL se limita a atacar el problema del proceso ETL considerando tres fuentes de información diferentes: archivos XML, bases de datos relacionales y archivos separados por comas (CSV). Para Onto-ETL lo mas importante de cada una de estas fuentes es la recuperación de los meta-datos, ya que todo el proceso esta fundamentado en un correcto entendimiento y clasificación de la información con ayuda de sus meta-datos. A continuación damos una breve descripción de los tipos de fuentes de información que se tomaron en cuenta:

1. **XML**: Se consideran los documentos XML bien formados y validos, así mismo como documentos de definición de estructura y restricciones de los documentos XML se consideran con XML schema.

2. **Bases de Datos**: las bases de datos que se consideran son de tipo relacional, en la actualidad existen una gran cantidad de gestores de bases de datos relacional, por este motivo se contempla el trabajo con este tipo de fuentes de información, ademas de que existen tecnologías que permiten la recuperación y acceso a los meta-datos de las mismas.

3. **Archivos separados por comas**: son un estándar ampliamente usado dentro de la industria, hay una gran cantidad de repositorios de información definidos bajo esta estructura simple pero poderosa, los archivos de texto plano que tiene como cabecera el nombre de los campos separados por comas, si es que los vemos como una tabla de un esquema relacional, este encabezado nos representa la meta información asociada a los mismos.

4.3.3. Modelos lógicos y físicos del Data Warehouse

Onto-ETL propone con base al estudio e investigación realizada, un Data Warehouse multidimensional basado en un modelo entidad relación, es decir un modelo ROLAP, en el capitulo 3 se dio una descripción de los sistemas OLAP multidimensionales basados en esquemas relacionales, para el caso de Onto-ETL, se consideran como posibles estructuras de los almacenes de datos finales, esquemas OLAP en copo de nieve o en estrella.

Debido a que se implementan estos esquemas sobre modelos relacionales, se tiene la ventaja de poder generar la estructura y las rutinas de inserción con sentencias SQL estándar, por tal razón una ves definida la estructura del mismo, se podrá proceder a elaborar y poblar el almacén de forma simple.

Fueron dejados de lado las bases de datos multidimensionales, debido a la complejidad de mapear un modelo basado en ontologías (muy parecido al relacional) a las sentencias de creación e inserción de las bases de datos multidimensionales.

Debido a que Onto-ETL trabaja con esquemas ROLAP, se da a la tarea de hacer los mapeos de las diversas fuentes de información y sus contenidos a las dos tipos de

tablas que se tienen en este tipo de esquemas: las tablas de hechos y las tablas de dimensiones.

Como veremos mas adelante Onto-ETL define un esquema para insertar la información a nivel de tupla en las tablas, ya sean de hechos o de dimensiones, siendo este ultimo paso, el ultimo eslabón en la cadena del linaje de los datos.

4.3.4. La gestión de Meta-datos

El proceso de gestión de Meta-datos comienza con la extracción de los mismos de las diferentes fuentes de datos, mostrado en la figura 4.5, este proceso toma como entrada las fuentes de información, ya sean bases de datos, archivos XML, archivos SCV. El proceso entonces es dividido en 4 tareas:

- **Interconexión con las diferentes fuentes de datos**: en este proceso se definen los mecanismos de comunicación que se usaran para poder tener acceso a las diversas fuentes de información, en el caso de los archivos XML se realizo el parseo y extracción con el API de Xerces, para los archivos CSV se hizo el proceso con GATE y el estándar JDBC de java para el trabajo con las bases de datos.

- **Extracción de los meta-datos**: en cada una de las fuentes de información tratadas encontramos meta-datos, las bases de datos las tienen en su tablas, atributos y relaciones, los archivos XML dentro de sus XSD en lo cuales encontramos tipos de datos simples y compuestos y en el caso de los archivos CSV tenemos los encabezados. Este proceso tiene el objetivo de localizar y extraer todos y cada uno de los meta-datos de las fuentes de información.

- **La generación de Entidades java**: con las entidades extraídas se generaran objetos java que encapsulen la información de los meta-datos de cada una de las fuentes de información a nivel de entidad.

- **El emparejamiento con la ontología maestra**: con ayuda del enfoque presentado en [55], a partir de las entidades java, se generan consultas SPARQL con la finalidad de encontrar los ítems dentro de la ontología maestra. Como resultado final de este proceso, se tiene una lista con los mapeos de los objetos de las fuentes de información hacia su respectivo elemento dentro de la ontología, así como también una lista de los elementos que no pudieron ser encontrados de forma automática.

Antes de seguir adelante, cabe hacer un paréntesis para describir un poco mas a fondo dos tareas importantes que tienen lugar en el proceso de extracción de meta información:

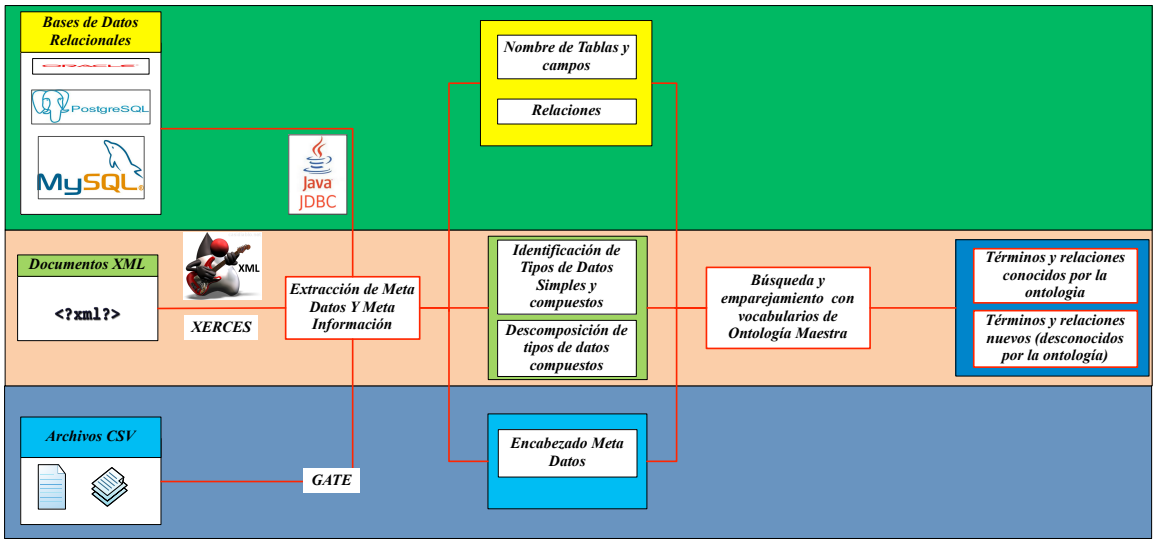

Figura 4.5: Extracción de Meta Información

Mecanismo de consulta y clasificación de meta-datos

El mecanismo de consulta, que determina que elemento de la ontología será asociado una entidad java que encapsula los meta-datos de las fuentes de información, toma en el nombre de la entidad, cuenta los atributos, su nombre, y tipo ademas de las relaciones que tiene, con la finalidad de generar una matriz de incidencias donde se registran las diferencias entre la entidad y el elemento de la ontología, esto permite proponer las respectivas transformaciones que se harán sobre las fuentes de información origen.

El etiquetado semántico de las fuentes de información

El termino de etiquetado semántico, es un termino adoptado por la comunidad de la Web Semántica, para describir el proceso de poner identificadores o marcas dentro del contenido de la web, para poder asociar el contenido a una ontología y posibilitar así la comprensión de los conceptos contenidos en la pagina por parte de una computadora.

En el caso de Onto-ETL, propone un equivalente a ese etiquetado semántico, para hacer una asociación entre los meta-datos de las fuentes de información que intervienen en el proceso ETL y su correspondiente definición dentro de la ontología. Esto se hace mediante un mapa de correspondencia que asocia a cada uno de los meta-datos de las fuentes de información con su respectivo concepto o definición dentro de la ontología, de esta forma sabremos el significado de todos y cada uno de los elementos involucrados en el proceso y será mas fácil su clasificación, aparte de tener ahora asociada una definición formal.

Categorización y clasificación de los elementos desconocidos

En el proceso de extracción y categorización de los meta-datos, habrá ciertas ocasiones en que la ontología maestra desconozca ciertos términos definidos en las entidades java, en tal caso se procederá a seguir el proceso descrito en la figura 4.6:

- **Agrupación de los términos relacionados**: con la ayuda del marco de trabajo propuesto en[56], las entidades java con atributos y relaciones en común son agrupadas dentro de una matriz de asociación, con la finalidad de construir pequeños kernles de información.

- **Construcción de relaciones**: con base al análisis de la matriz de asociación construida en le paso anterior, las relaciones entre los diversos objetos son creados.

- **Generación de micro ontologías**: una ves establecidas las relaciones entre los objetos, estos son transformados a su equivalente representación ontológica con ayuda de mapeos objeto-ontologico **[55]**. Debido a que no todos los términos quedan relacionados, obtenemos así algunas ontologías aisladas (micro - ontologías).

- **Adición de las micro ontologías a la ontología maestra**: con base a lo expuesto en el algoritmo de mezcla de ontologías propuesto en [19], el cual esta implementado como una utilidad de Protege [24], las ontologías aisladas son adicionadas a la ontología maestra. El resultado de la integración es entonces presentado a los expertos de dominio para corroborar su consistencia y para hacer los ajustes pertinentes, para finalmente tener una nueva versión de la ontología maestra con los nuevos términos.

La adición de micro-Ontologías a la ontología maestra (mezcla de ontologías)

Debido a la importancia del proceso de adición de las micro-ontologias, el algoritmo empleado para realizar esta tarea se detalla en la figura 4.7 mediante un diagrama de actividad.

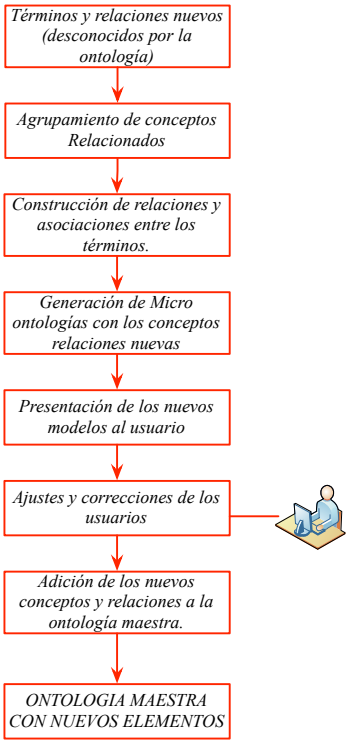

Figura 4.6: Inserción términos desconocidos

4.3.5. Generación del modelo lógico de datos

Al terminar el proceso de la gestión de meta-datos, se tendrán identificados con ayuda de la **ontología maestra**, a todos y cada uno de los conceptos que intervendrá en la integración de la información; las fuentes de información a este punto se tendrán etiquetadas y categorizadas, siguiendo así el proceso de generación del modelo lógico del Data Warehouse, el cual es mostrado en la figura 4.8, las tareas del proceso son las siguientes:

- **Generación de la lista de términos involucrados**: debido a que no todos los términos contenidos dentro de la ontología maestra serán empleados en el proceso ETL, se generara una lista de los términos de la ontología y entidades java que sí participaran.

- **Generación de consultas de extracción**: con la ayuda del algoritmo de extracción y corte de ontologías definido en JENA [20], se generan sentencias para la extracción de una sub ontología de la ontología maestra, para de esta forma tener en una sola ontología la representación del modelo lógico del Data Warehouse.

- **Extracción de términos**: las sentencias generadas son ejecutadas y entonces se obtiene un conjunto de términos contenidos dentro de una o varias ontologías

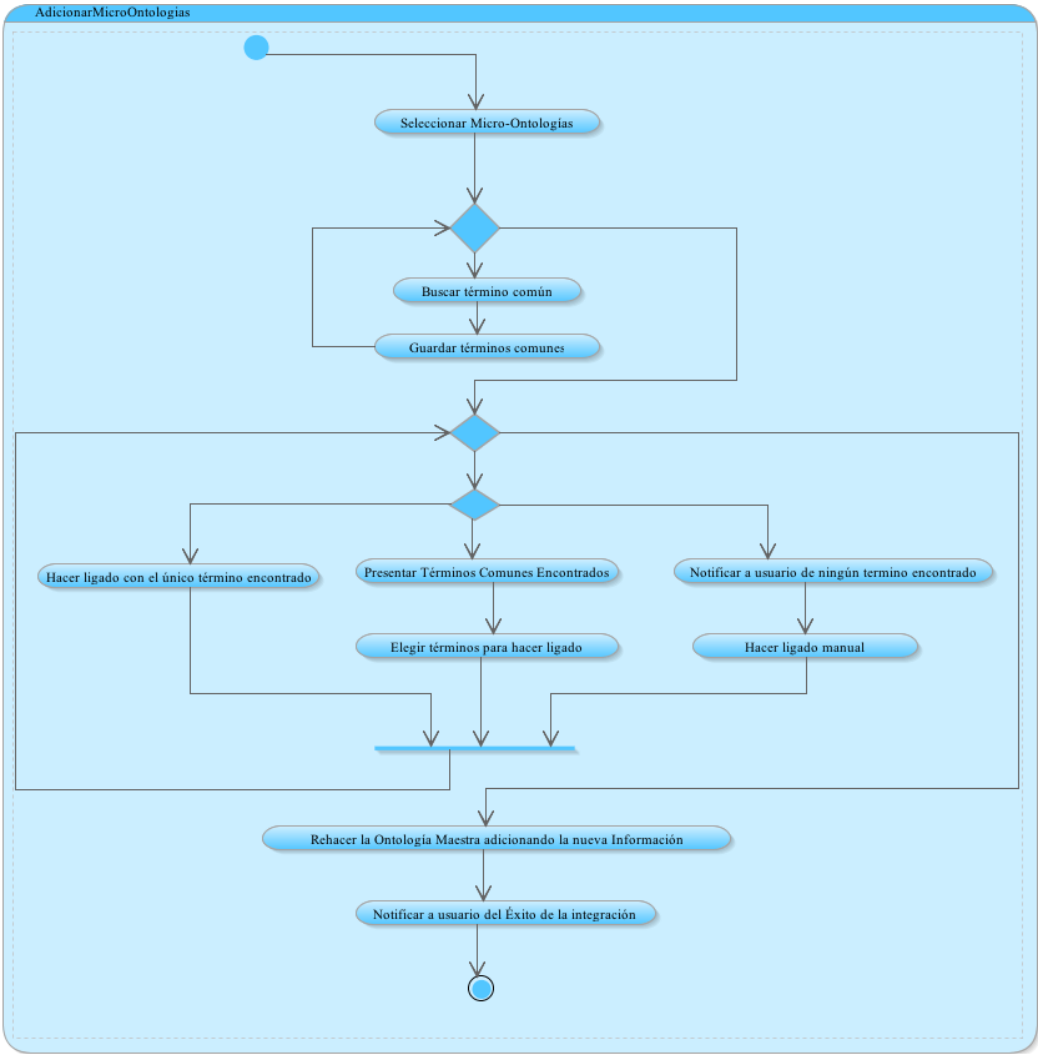

Figura 4.7: El proceso de la adición de micro-Ontologías

aisladas, todos ellos en conjunto representan la definición formal de los términos involucrados en el proceso ETL.

- **Construcción de la nueva ontología**: con la finalidad de construir la ontología final que será el modelo lógico del Data Warehouse, los términos y ontologías aisladas son mezcladas con el algoritmo presentado en [19], finalmente la ontología obtenida es un modelo de datos formal de las fuentes involucradas.

El modelo lógico generado se presentara al usuario, con la finalidad de que este corrobore dicho modelo y pueda hacer modificaciones como: modificar relaciones, eliminar conceptos, etc. Para que una vez aprobado se tenga listo y en ayuda de este se pueda elaborar el modelo físico de datos.

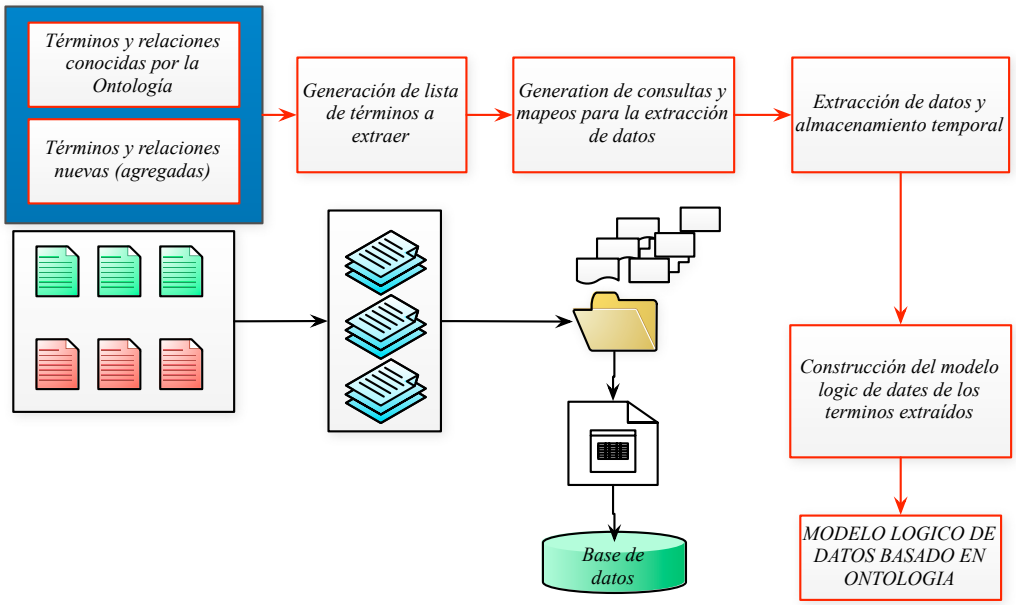

Figura 4.8: Generación del modelo lógico de datos

4.3.6. Creación y poblado del modelo físico

Dado el modelo lógico basado en la ontología, se procederá a generar el modelo físico del mismo, esto con la intervención del usuario, puesto que existirán opciones para la construcción del mismo, dependiendo del modelo que elija el usuario, que podrá ser un esquema de estrella o de copo de nieve ambos basados en esquemas relacionales.

El proceso es mostrado en la figura 4.9, tiene la función de generar un esquema físico en particular a parte del modelo lógico de datos. Las tareas del proceso son:

- **Selección del modelo físico de datos**: el administrador ETL selecciona una de los dos tipos de modelos físicos.
- **Generación de los mapeos del modelo lógico a modelo físico**: de cada uno de los elementos del modelo lógico, se obtendrá la entidad java asociada. Posteriormente de cada una de estas identidades, se invocara un método interno que regresa una sentencia SQL standard de creación, para poder crear la tabla dentro del RDBMS, Finalmente un mapa de datos es creado entre los objetos java entidad del modelo lógico y las sentencias generadas.
- **Ejecución de procedimientos para la creación del modelo físico**: las sentencias generadas en el punto anterior, son entonces ejecutadas para crear la estructura del modelo fisco del Data Warehouse.
- **Generación de los mapeos de las fuentes de almacena al modelo físico**: en este paso de igual forma se recuperan los objetos java entidad asociados al

modelo lógico, pero en esta ocasión, es invocado otro método interno que regresa esta ves un conjunto de sentencias de inserción para todos y cada uno de los datos asociados a esta entidad java. Como ultimo punto, con ayuda del mapa de datos que relaciona objetos java entidad y las sentencias de extracción, se genera un nuevo mapa de datos que relacionara los elementos del modelo físico y las sentencias de inserción creadas.

- **Generación de los procedimientos de extracción e inserción**: con la colaboración de los mapas de correspondencia entre las fuentes de información y su relación con el modelo físico y lógico, se generan los scripts generales para el poblado del Data Warehouse.
- **Ejecución de los procedimientos**: el script de extracción se ejecuta con la finalidad de extraer los datos y enseguida de esto el script de inserción es ejecutado para poblar el Data Warehouse.

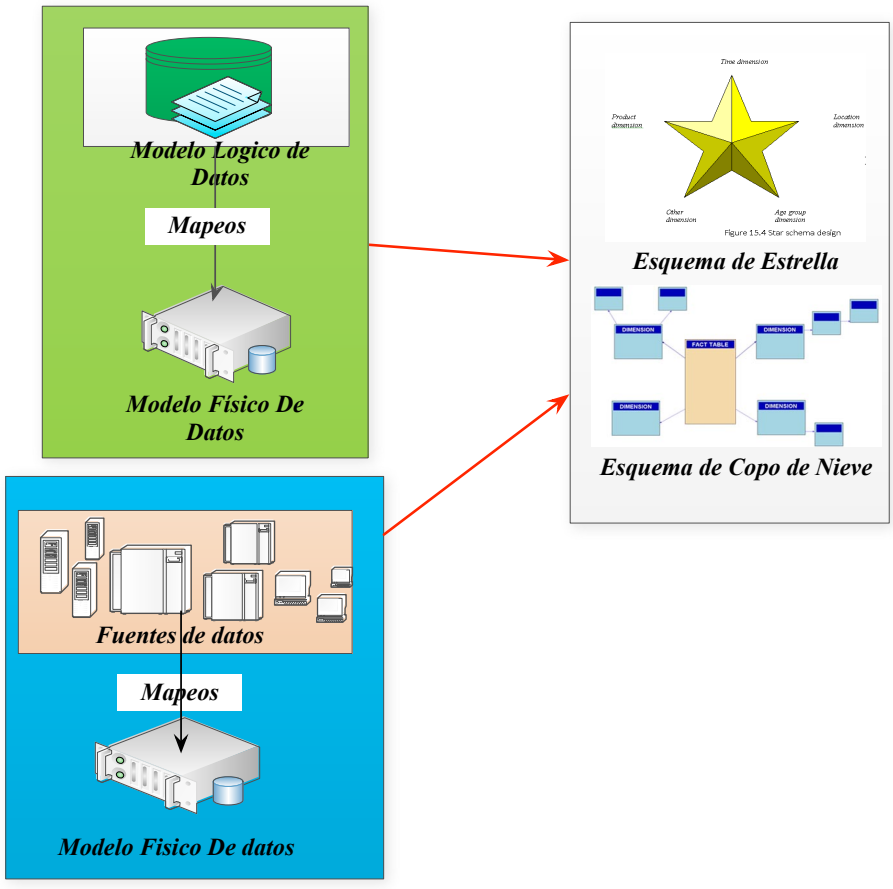

Figura 4.9: Generación del modelo Físico

Desde el momento en que los meta-datos de las fuentes de información son extraídos de las fuentes de información, se comienza una serie de mapeos y transformaciones

de los mismos, para poder ir generando diversos elementos que requiere el proceso, en la figura 4.5 mostramos las principales transformaciones y relaciones que sufren los meta-datos a lo largo del proceso:

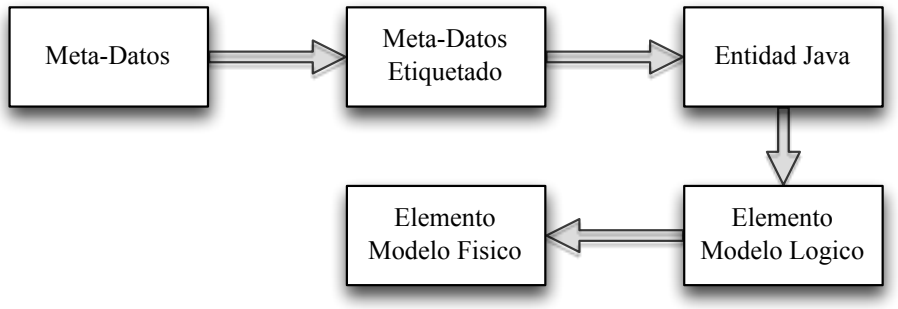

Figura 4.10: Las transformaciones de los meta-datos en el proceso ETL

A continuación detallamos los mapeos y sus transformaciones aplicadas sobre elementos que define Onto-ETL:

El mapeo de las fuentes de datos al modelo lógico

El primer ligado de las fuentes de datos hacia el proceso de Onto-ETL se realiza con el etiquetado semántico de los meta-datos de las mismas, posteriormente al definir los elementos de la metodología maestra que intervendrán en el proceso, se hace un mapeo de todos y cada uno de los elementos de las fuentes origen a su respectiva representación dentro del modelo lógico, el ligado para asociar un elemento de las fuentes de información hacia su respectivo elemento de la ontología maestra es a través de la creación de una entidad java, la cual encapsula los atributos, relaciones y propiedades de los meta-datos del elemento de la fuente origen ligado a su respectivo elemento ontológico. Cabe mencionar que un elemento del modelo lógico pude tener asociados varios objetos entidad java, debido a que múltiples elementos de las fuentes de datos origen pueden ser clasificadas sobre el mismo tipo de elemento dentro de la ontología maestra, esto genera una asignación automática de los elementos de las fuentes de datos origen a su respectivo destino.

El mapeo del modelo lógico al modelo físico

EL segundo ligado o transformación de los meta-datos, se realiza cuando se pasa del modelo lógico al modelo físico, aquí se hace un mapa de correspondencia entre los elementos del modelo lógico a su respectivo elemento dentro del modelo físico. Puesto que un modelo entidad relación puede ser visto como un caso particular de una ontología[57] y puesto que Onto-ETL emplea enfoques R-OLAP, se hace uso del algoritmo clásico mapeo E-R a su respectivo modelo físico, al finalizar el proceso de mapeo y construcción, solo se hace un ligado de los elementos del modelo lógico a

su respectivo elemento dentro del modelo fisco, ademas de generar un mapa entre las entidades java con su respectivo elemento dentro del modelo físico.

Creación de instrucciones para la inserción de información en el modelo físico

El ultimo punto a detallar dentro de la metodología para realizar el proceso ETL, es la generación de las sentencias de inserción para poblar el modelo físico, esto es posible debido a los mapeos que se realizaron durante el proceso y debido a la definición de las clases java que dan soporte a los meta-datos.

Al finalizar el proceso de mapeo, se tiene perfectamente identificado, el origen cada un de los elementos de las fuentes de información y su respectivo destino dentro del modelo físico generado, ahora bien los objetos java entidad que encapsulan los meta-datos de las fuentes de información, implementan una interfaz que define de forma genérica, propiedades definir su ubicación, tipo, elementos, etc.; ademas de tener métodos para extracción de información, extracción de meta-datos, generación de sentencias de inserción y de creación. Esta interfaz java es implementada pos los objetos entidad java, los cuales definen de manera interna el tratamiento que se dará en cada uno de los diferentes casos:

- Bases de datos
- Archivos XML
- Archivos CSV

En la siguiente sección se dará una explicación mas detallada de los componentes software que intervienen en las anteriores metodología, para tener un mejor entendimiento del marco de trabajo.

4.4. La Biblioteca de componentes software

Con la finalidad de dar soporte a las tareas de las metodologías antes descritas, se desarrolló una biblioteca componentes de software sobre una estructura conceptual en capas (figura 4,11). Siguiendo la arquitectura de abajo hacia arriba, tenemos las siguientes capas:

- **Capa de datos origen**: en esta capa están definidos los componentes que realizan la interconexión con las diversas fuentes de información.
- **Capa de extracción de meta información**: esta capa contiene los componentes que ayudan a la extracción y encapsulación de la información.
- **Capa de abstracción y generalización**: contiene objetos que soportan el tratamiento e interacción con elementos de la ontología.
- **Capa de modelo de Data Warehouse**: contiene objetos para la administración de los modelos lógicos y físicos.
- **Capa de integración de la información**: define componentes encargados de las tareas sobre el Data Warehouse final.

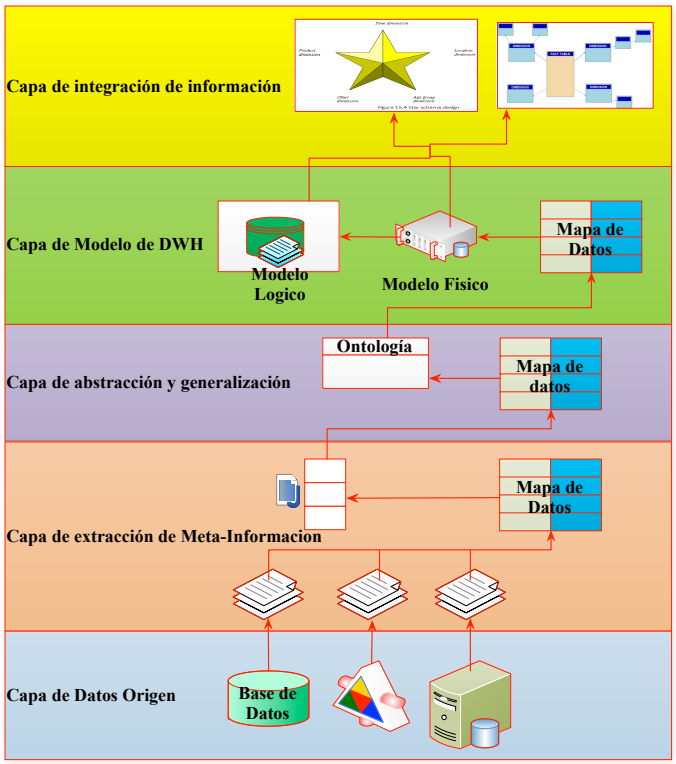

Figura 4.11: Arquitectura General por capas

Una visión mas integral de los componentes que tiene la biblioteca de Onto-ETL, se puede ver en la figura 4.11, donde se muestra el diagrama de clases del marco de trabajo.

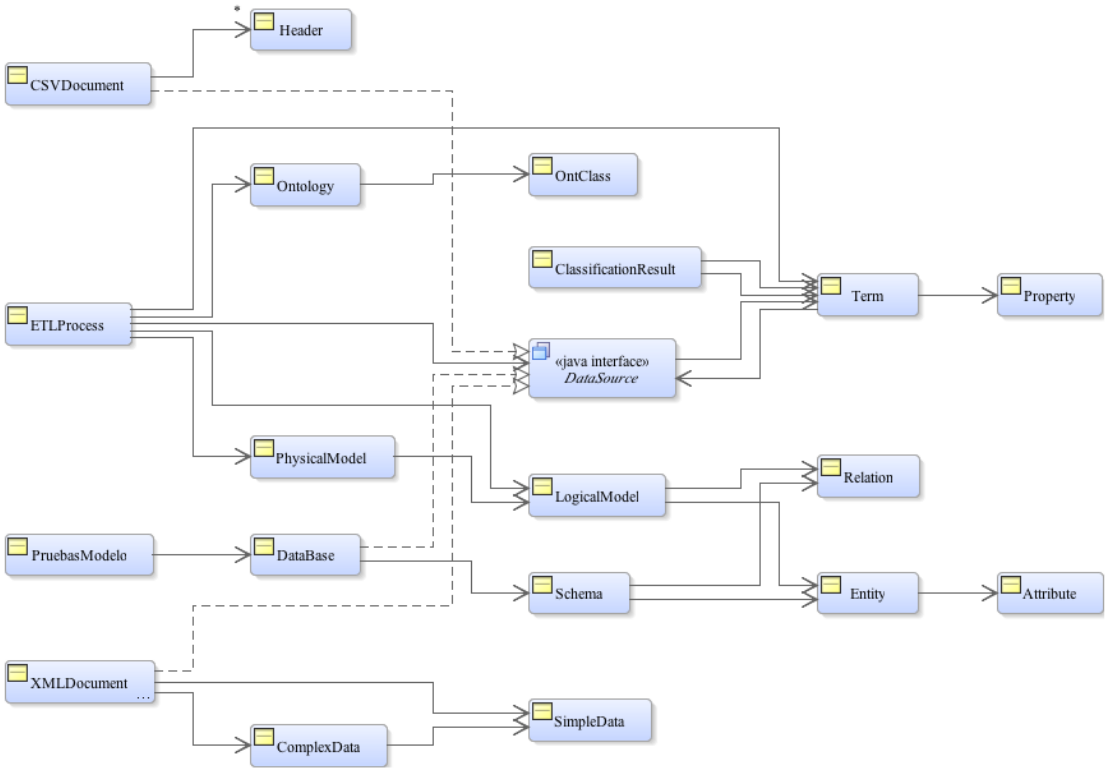

Figura 4.12: Los elementos de la Biblioteca de Onto-ETL

A continuación describimos los principales componentes de cada una de las capas, exponiendo cada componente mediante su tarjeta CRC (clase responsabilidad y colaboración)

4.4.1. Capa de fuentes de datos

En esta capa están definidos los componentes que realizan la interconexión con las diversas fuentes de información, esta capa tiene la peculiaridad de albergar un solo componente, en el cual esta el corazón y puente entre las fuentes de información del proceso ETL y Onto-ETL.

DataSource	
RESPONSABILIDADES	**COLABORADORES**
Definir las propiedades y comportamientos genéricos de las fuentes de información que intervienen en el proceso.	CSVDocument DataBase XMLDocumento ETLProcess Term

4.4.2. Capa de extracción de información

En este nivel encontramos los componentes que ayudan a la extracción y encapsulación de la información, tanto en el proceso de captura de conocimiento como en el proceso de extracción de información de las fuentes de datos; de manera adicional, contiene los objetos encargados de la asociación entre los elementos del modelo físico, las estructuras de captura del conocimiento y su respectiva representación en objetos java.

CSVDocument	
RESPONSABILIDADES	**COLABORADORES**
Implementar los métodos para la recuperación de meta información y datos de los documentos CSV.	DataSource Header

Header	
RESPONSABILIDADES	**COLABORADORES**
Gestionar la información relacionada con el elemento header contenido dentro de los archivos CSV.	CSVDocument

DataBase	
RESPONSABILIDADES	**COLABORADORES**
Implementar los métodos para la recuperación de meta información y datos sobre bases de datos con soporte de conectividad JDBC.	DataSource Schema

Schema	
RESPONSABILIDADES	**COLABORADORES**
Gestionar la información a nivel de esquema dentro de una base de datos, incluidos sus propios elemento tales como: relaciones y entidades.	DataBase Entity Relation

Entity	
RESPONSABILIDADES	**COLABORADORES**
Gestionar la información y meta información de las bases de datos a nivel de entidad.	Schema Attribute LogicalModel

Attribute	
RESPONSABILIDADES	**COLABORADORES**
Gestionar la información y meta información de las bases de datos a nivel de atributo.	Entity

Relation	
RESPONSABILIDADES	**COLABORADORES**
Manejar la información de las relaciones entre entidades dentro de la base de datos.	Schema Entity

XMLDocument	
RESPONSABILIDADES	**COLABORADORES**
Implementar los métodos para la recuperación de meta información y datos de archivos XML. (tanto para el proceso ETL como para la captura del conocimiento)	DataSource ComplexData SimpleData

ComplexData	
RESPONSABILIDADES	**COLABORADORES**
Gestionar el manejo de meta-datos e información de tipos de datos compuestos dentro de archivos XML.	XMLDocument SimpleData

SimpleData	
RESPONSABILIDADES	**COLABORADORES**
Gestionar el manejo de meta-datos e información de tipos de datos simples dentro de archivos XML.	XMLDocument ComplexData SimpleData

Term	
RESPONSABILIDADES	**COLABORADORES**
Gestionar de manera única todos los meta-datos de las diversas fuentes de información llevadas a un nivel de entidades o conceptos. Proporcionar al proceso ETL las entidades que intervienen en el proceso a nivel de fuentes de información. Ser el puente entre las fuentes de datos y el proceso ETL. Ser el puente entre las la ontología maestra y las fuentes de información.	DataSource Property ETLProces ClasificationResult

Properti	
RESPONSABILIDADES	**COLABORADORES**
Gestionar a nivel de propiedades, las diversos atributos de las fuentes de datos origen.	Term

4.4.3. Capa de abstracción y generalización

A este nivel encontramos objetos que soportan el tratamiento e interacción con elementos de la ontología, para hacer procesos de adición, modificación y eliminación de conocimiento; ademas de búsqueda, etiquetado semántico; como complemento se tienen objetos que ayudan a crear las asociaciones entre objetos java y elementos de la ontología.

Ontology	
RESPONSABILIDADES	**COLABORADORES**
Administrar la ontología en general, facilitar tareas como adición, eliminación y modificación de elementos. Proporcionar operaciones al proceso ETL para la clasificación de los meta-datos de las fuentes de información y la generación del modelo lógico.	ETLProcess OntClass

OntClass	
RESPONSABILIDADES	**COLABORADORES**
Gestionar y administrar las clases que forman parte de una ontología.	Ontology

ClassificationResult	
RESPONSABILIDADES	**COLABORADORES**
Almacenar los resultados y las relaciones que resultan al clasificar los términos de las fuentes de información con la ontología.	Term Properti ETLProcess

4.4.4. Capa de modelo de Data Warehouse

Esta capa guarda componentes que exponen métodos para la extracción del modelo lógico, el mapeo del modelo lógico al físico, la construcción del modelo físico y de las transformaciones que sufrirán los datos antes de ser insertados al almacén final.

LogicalModel	
RESPONSABILIDADES	**COLABORADORES**
Facilitar tareas de administración sobre el modelo lógico del Data Warehouse y sus elementos, algunas de ellas son la creación, modificación o generación del modelo físico.	Relation Entity ETLProcess PhysicalModel

PhysicalModel	
RESPONSABILIDADES	**COLABORADORES**
Facilitar tareas de administración sobre el modelo físico del Data Warehouse y sus elementos, tales como la creación o modificación. Facilitar tareas de de inserción de datos sobre el Data Warehouse.	LogicalModel ETLProcess

4.4.5. Capa de Integración de información

Define componentes encargados de la interacción con el Data Warehouse final, recabar las instrucciones para la generación de los scripts de creación de la estructura y de inserción de la información.

ETLProcess	
RESPONSABILIDADES	**COLABORADORES**
Administrar los diversos elementos presentes en el proceso ETL	Term Ontology DataSource LogicalModel PhysicalModel

4.5. Comentarios finales

La definición de las metodología para la captura del conocimiento y para realizar el proceso ETL son elementos cruciales para poder tener éxito al implementar un procesos ETL enfocados en ontologías. A pesar de que actualmente existen enfoques para atacar el proceso ETL con ontologías, ninguno de ellos guarda una proporción equilibrada entre, la definición estricta pero poco aplicable a la vida real de la ontología [49, 10] y la definición relajada y sobre ajustada de la misma pero la resolución de algunos problemas [52]. Después de definir las metodologías y la biblioteca de componentes de Onto-ETL, podemos afirmar que mostramos un camino claro a seguir en el desarrollo de proyectos que buscan solventar la problemática ETL con ayuda de ontologías.

Las metodologías desarrolladas son intuitivas, claras y aplicables a casos de prueba real, toman en cuenta la perspectiva de los diversos roles involucrados en el proceso ETL, los expertos del dominio, los administradores del conocimiento y los administradores del proceso ETL.

Capítulo 5

Caso de estudio

Onto-ETL es un marco de referencia (framework), de uso general en problemas que se adapten ala metodología propuesta, por lo que el producto final de este trabajo de tesis, fueron dos metodología acompañadas de una biblioteca de componentes que en conjunto definen la forma de realizar un proceso ETL, no obstante se decidió realizar un caso de prueba aplicando el marco de trabajo desarrollado, para lo cual se desarrollo una sencilla aplicación que hace uso de Onto-ETL para la resolución de un caso de estudio.

El capitulo se divide en cuatro secciones, en la primera se da un panorama general del problema que planteo caso de estudio, posteriormente se presentan las dos aplicaciones realizadas para poder atacar el problema con ayuda de Onto-ETL, finalmente presentamos los resultados obtenidos.

Para propósitos ilustrativos y de una mejor explicación, en algunas partes describiremos la solución a la par de la explicación de la implementación basada en Onto-ETL.

5.1. Descripción del problema la aplicación

La problemática del caso de estudio que se soluciono con ayuda de Onto-ETL, consistía en la integración de múltiples fuentes de información para la generación de un Data Warehouse que centralizara la información de los diversos sistemas de la empresa, cabe señalar que las fuentes de información y sus respectivos datos fueron prestados con la intención de poder realizar la prueba de la aplicación.

La compañía dedicada al giro de las auto partes, contaba con diversas fuentes de datos, algunas de ellas eran: bases de datos, archivos XML y archivos CSV, por lo que se pudo aplicar Onto-ETL en la solución del problema.

De acuerdo al enfoque propuesto por Onto-ETL, en primer lugar se identificaron los objetivos y aspectos centrales que se iban a modelar en el Data Warehouse, dichas áreas de interés eran las siguientes:

- Ventas: se requiera tener concentrada la información relacionada a venta, como lo es el volumen en el tiempo, los artículos vendidos, los clientes, los lugares de

venta, etc.

- Inventario: se requería concertar información recalada a productos y proveedores.

- Recursos humanos: se requiera unificar información de ingresos, prestaciones, productividad.

- Finanzas: se requería unificar información de facturación, compras y ventas.

Puesto que Onto-ETL plantea dos metodologías para poder llevar a cabo el proceso ETL, se construyeron dos aplicaciones de software java stand alone, una de ellas para realizar el proceso se captura del conocimiento y construcción de la ontología, y la segunda para realizar el proceso ETL.

5.2. La aplicación para la Gestión de Conocimiento

Construida para llevar a cabo la metodología de Gestión de Conocimiento y capturar el conocimiento de los expertos a través de una interfaz visual basada en grafos a partir de los cuales se realiza la creación de la ontología, se muestra en la figura 5.1, permite la adición de entidades y definición de sus propiedades, la definición de las reglas de negocio y la generación de la ontología con base a los elementos capturados. A continuación describimos mas a detalle sus funcionalidades.

Siguiendo la metodología propuesta, esta aplicación permite la creación de términos, provenientes de las lluvias de ideas y reuniones con los expertos, facilita a los expertos de dominio, mediante capturas de información simple.

- El panel de «**Elementos ontología**», permite la definición de términos y de sus propiedades, una de las pestañas lista los términos dados de alta hasta el momento, y la otra permite capturar las propiedades, la lista de términos permite arrastrar y soltar los términos dentro del área de ontología, donde se definen las relaciones.

- El panel de **«ontología»** muestra las relaciones entre los diferentes términos en un enfoque basado en grafos, esta área de trabajo permite editar los elementos y crear nuevas relaciones por medio de un enfoque drag and drop.

- El panel de «**Reglas de negocio**», lista las reglas de negocio dadas hasta el momento y permite agregar o eliminar reglas existentes.

Un punto a señalar es que este modelo genera de forma automática la ontología subyacente, ademas de que permite guardar el modelo creado y reabrir cualquier trabajo previamente guardado.

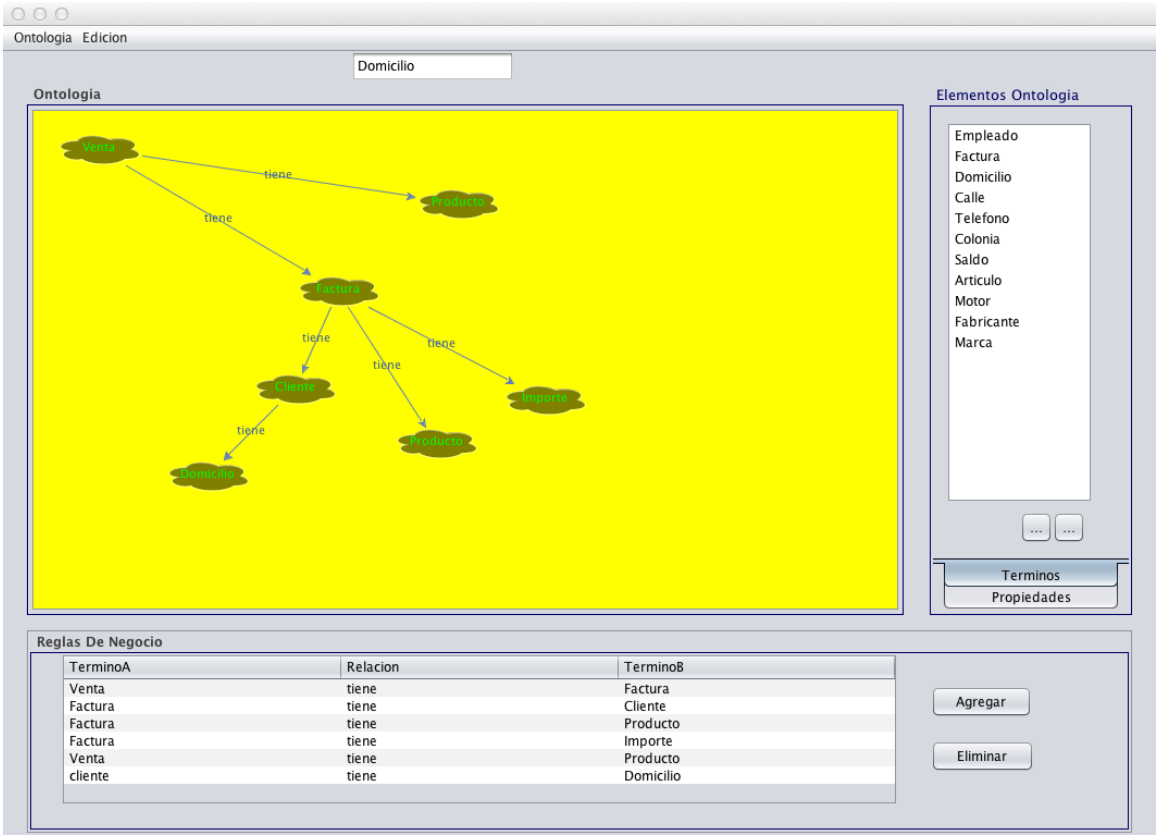

Figura 5.1: Interfaz para la Gestión de Conocimiento

5.3. La aplicación para el proceso ETL

La segunda aplicación fue creada para poder efectuar el proceso ETL siguiendo la metodología propuesta por Onto-ETL, en ella se tienen componentes visuales para la configuración y selección de las fuentes de información, la clasificación de las fuentes de información y la generación y poblado del Data Warehouse con las fuentes de datos origen.

5.3.1. Interfaz de usuario para la gestión del proceso ETL

La interfaz principal definida para realizar el proceso ETL se muestra en la figura 5.2, esta permite configurar los parámetros iniciales del proceso ETL, se divide en los siguientes paneles:

- **Panel de fuentes de información**: muestra las fuentes de información que intervendrán en el proceso ETL, así mismo permite agregar nuevas fuentes, modificar los parámetros de alguna o eliminar una del proceso. Para realizar modificaciones y altas de fuentes de información se apoya de la ventana de configuración de fuentes de datos origen (figura 5.3).

- **Panel de Ontología**: Muestra la ontología que va a regir el proceso ETL, con base a la cual se seguirá la metodología definida por Onto-ETL, permite ademas agregar la ontología, modificar la ontología que se ha seleccionado y eliminar la ontología.

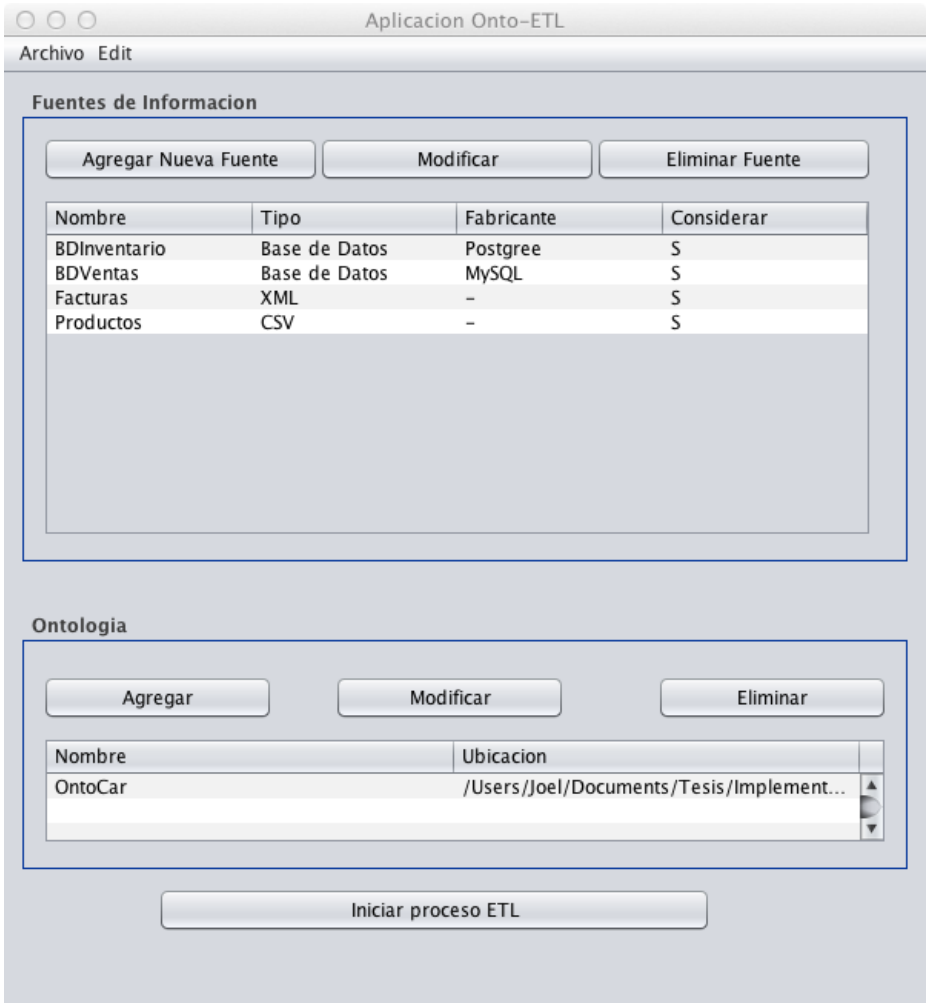

Figura 5.2: Interfaz de usuario para la gestión del proceso ETL

5.3.2. Interfaz para la configuración de fuentes de datos origen

La ventana de configuración de fuentes de información, la cual se muestra en la figura 5.3, permite capturar los parámetros necesarios para trabajar con una fuente de información, estos varían dependiendo el tipo de fuente que con el que se quiera trabajar, se permite seleccionar el tipo de fuente con el quiere trabajar y en función de ello se ajustan los parámetros de la ventana.

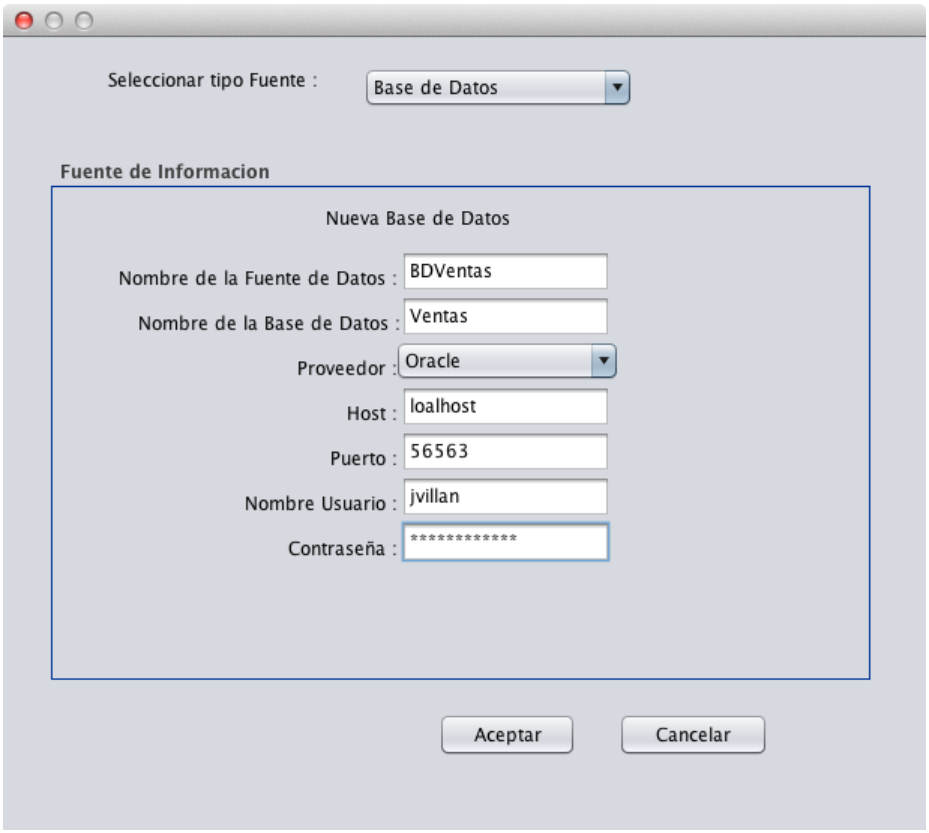

Figura 5.3: Configuración de fuentes de datos origen

5.3.3. Interfaz de usuario para la gestión del modelo Lógico

Dentro de la interfaz general del proceso ETL (figura 5.2), se tiene el botón de iniciar el proceso ETL, para poder comenzar con el proceso, este como esta definido por Onto-ETL, comienza con la extracción y clasificación de los meta-datos con ayuda de la ontología maestra.

Al comenzar el proceso ETL se muestra la ventana de gestión del modelo lógico, la cual esta compuesta por tres paneles distintos:

- **El panel de Modelo lógico**: muestra el esquema ontológico, con los términos y relaciones que se encontraron al realizar el proceso ce clasificación de los meta-datos, es decir los meta-datos y relaciones existentes entre las diversas fuentes de información.

- **El panel de Términos no encontrados**: muestra en primera instancia los términos no encontrados dentro de la ontología maestra, ademas de las propiedades de los mismo, así mismo ofrece la opción de resolver los conflictos siguiendo el proceso definido por Onto-ETL, al cabo de esto, se tendrán identificados todos los términos y se podrá proceder a la creación del modelo físico.

- **El panel de propiedades y características asociadas**: el mostrado en la

figura 5.4, muestra las propiedades ontológicas de los diversos elementos que componen el modelo lógico mostrado en el área de trabajo central (Panel de Modelo Lógico).

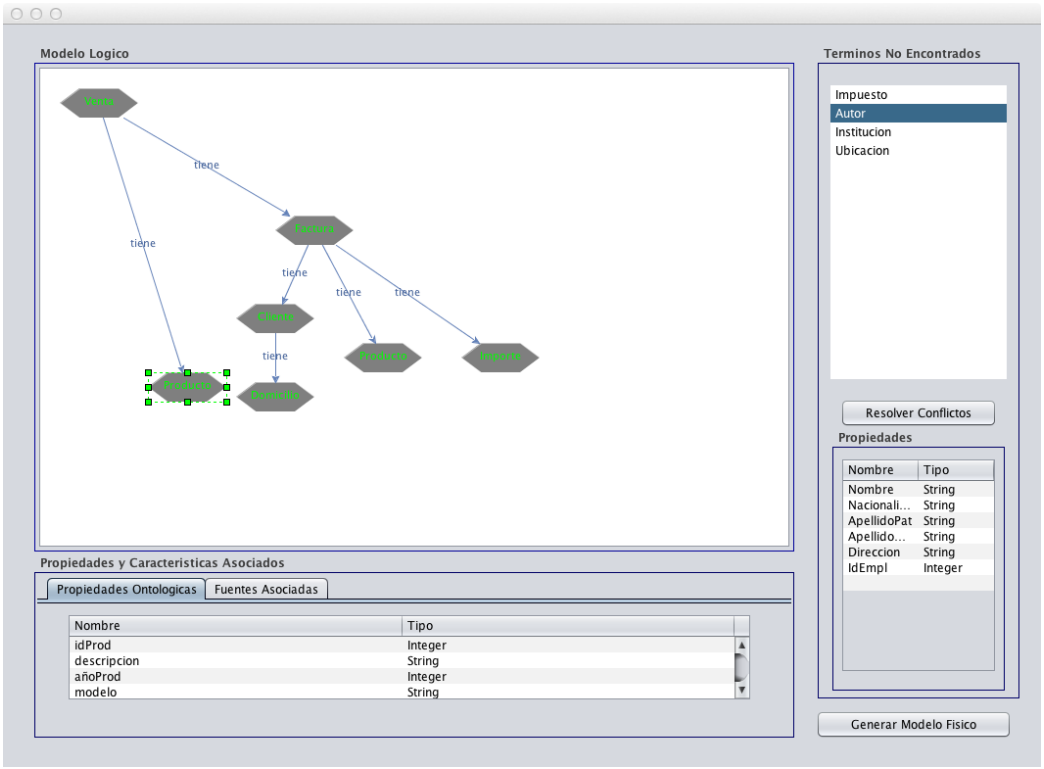

Figura 5.4: Interfaz de usuario para la gestión del modelo Lógico

- **El panel de propiedades y características asociadas** (2): el mostrado en la figura 5.5, muestra las fuentes de datos asociadas a un elemento del modelo lógico de datos, es decir los términos de las diversas fuentes de información, que fueron clasificados como parte de dicho elemento de la ontología, la lista muestra el nombre de la fuente de datos de donde proviene el elemento, el tipo de fuente de datos y el nombre bajo el cual se denomina dentro de su fuente de datos origen.

5.3.4. Interfaz para la gestión del Modelo Físico

Invocada al presionar el botón de Generar Modelo Físico de la figura 5.5, esta interfaz permite la definición del modelo físico final y se sus configuraciones, se compone de los siguientes paneles:

- **Panel de Modelo Físico de Data Warehouse**: muestra el modelo físico generado a partir del modelo físico, de igual forma es un área de trabajo donde

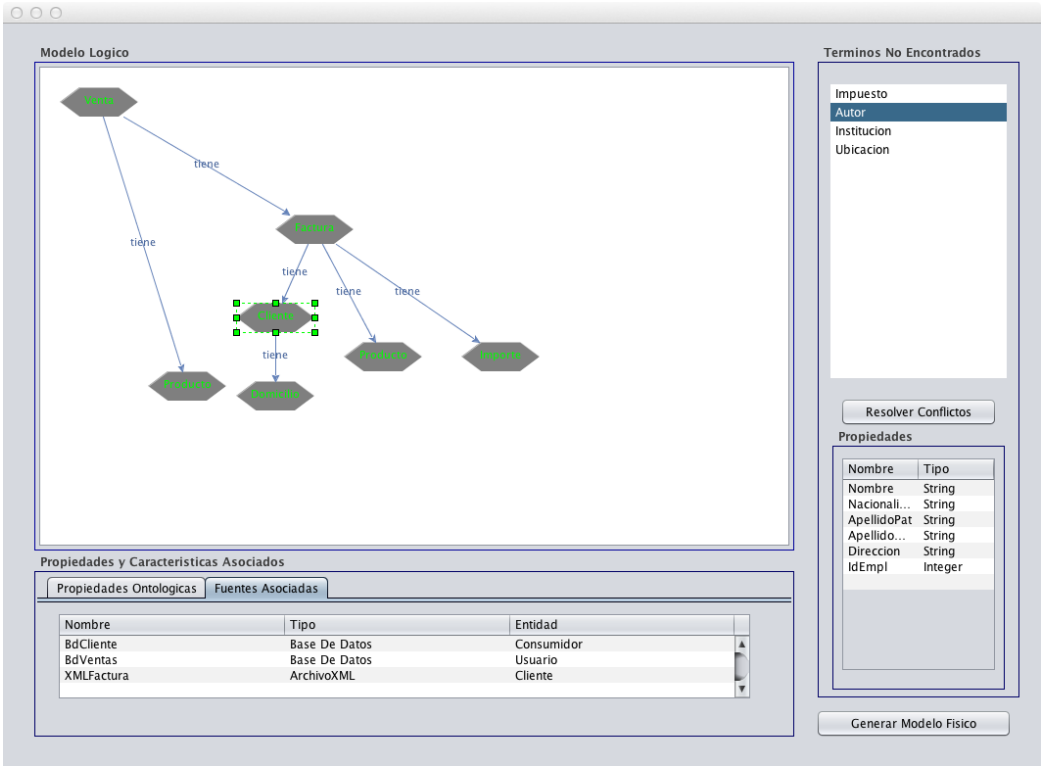

Figura 5.5: Interfaz de usuario para la gestión del modelo Lógico, fuentes asociadas

se pueden modificar las configuraciones propuestas, debido a que el enfoque de Onto-ETL es trabajar con esquemas ROLAP, los esquemas propuestos se elaboran base de tablas hecho y tablas dimensión.

- **Panel de propiedades Elemento**: muestras las propiedades de las entidades del modelo propuesto, con la opción de modificarlas y actualizar los nuevos valores en el modelo del área de trabajo.

- **Panel de Configuración Modelo físico**: permite definir las configuraciones que se tomaran en cuenta al generar el modelo físico, entre ellas la base de datos destino y si se quiere generar solo las sentencias de creación de estructura o solo de inserción de datos.

- **Botones de operación final**: destinados a realizar las tareas finales del proceso, la generación de la estructura del Data Warehouse o el poblado del mismo.

Con ayuda de las aplicaciones detalladas en las sub secciones anteriores se implemento la solución sobre el caso de estudio ya explicado, en las siguiente sección hablaremos de los resultados que se tuvieron y de las métricas tomadas en cuenta para poder ponderar dichos resultados.

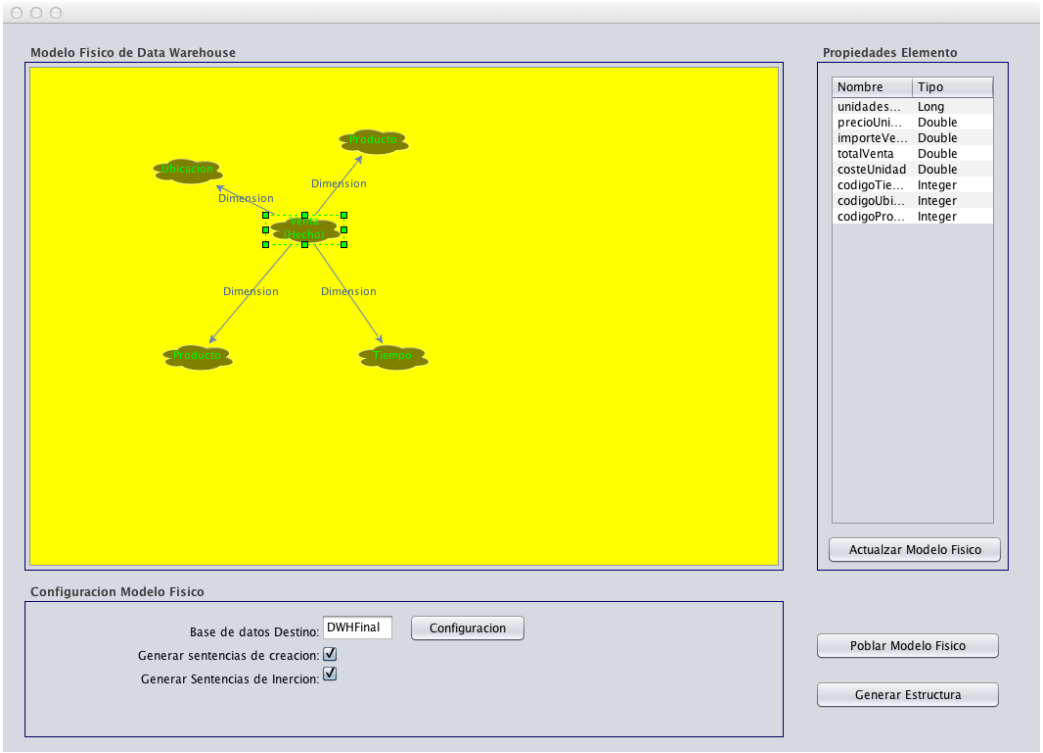

Figura 5.6: Interfaz para la gestión del Modelo Físico

5.4. Estadísticas y resultados

En esta hablaremos de los detalles técnicos que involucro el problema que se afronto con la ayuda de Onto-ETL, tales como son las características especificas de las fuentes de datos con las que se trabajo, las personas que intervinieron en el proceso y por ultimo daremos los resultados sobre algunas métricas que propone la literatura para evaluar aplicaciones ETL.

5.4.1. Datos técnicos del problema

En la fase de captura del conocimiento se tuvo interacción con personal con experiencia en las áreas objetivo de la implementación del Data Warehouse (ventas, inventario, recursos humanos y finanzas), al tener el contacto con los integrantes de la organización, se encontró renuencia a seguir la metodología y procedimientos del marco de trabajo, pero al ver las ventajas y los resultados obtenidos al aplicar Onto-ETL, se cambio un poco la perspectiva y se valoro un poco mas la Gestión de Conocimiento aplicada a la solución de problemas dentro de la organización.

Con respecto a las características de las fuentes de datos con las que se trabajo e implemento Onto-ETL, tenían las siguientes características:

- **Base de datos My SQL** : dentro de la base de datos se tenia 35 tablas que sumadas tenían al rededor de 20,000 registros

- **Base de datos Postgress**: dentro de la base de datos se tenían 26 relaciones que sumadas tenían aproximadamente 16,000 registros.
- **Archivos XML**: se tenían alrededor de 30 archivos XML que en suma representaban 15,000 registros.

En cuanto al Data Warehouse final se implemento sobre un sistema gestor MySQL, sobre el cual se elaboraron los esquemas ROLAP obtenidos durante el proceso y se pobló con las sentencias que se generaron con la aplicación, se obtuvieron 5 esquemas multidimensionales poblados con los datos provenientes de las fuentes de información citadas en el punto anterior.

5.4.2. Métricas y resultados

Es difícil evaluar el desempeño de un marco de trabajo, debido a que involucra demasiados aspectos cualitativos, como la definición de las tecnologías que le darán soporte, los procedimientos, las tareas, por mencionar algunos, ademas de que el enfoque es único en su categoría, sin embargo consideramos importantes los siguientes aspectos par la valoración de los resultados obtenidos:

- El numero de elementos de las fuentes de información que no pudo ser extraído de manera adecuada.
- El numero de errores en el modelo lógico generado de forma automática.
- El numero de sentencias de creación creadas de forma incorrecta.
- El numero de sentencias de inserción creadas de forma incorrecta.

Los resultados principales de la implementación se muestran en la siguiente tabla:

Aspecto	Resultado
Tiempo de implantación	1 semana
Tiempo de integración	1 día
Porcentaje de datos correctamente integrados	95 %
Porcentaje de errores en el modelo lógico generado	2 %
Porcentaje de errores en el modelo físico generado	3 %
Porcentaje de errores de las consultas de creación	2 %
Porcentaje de errores de las consultas de inserción generadas	4 %

Cuadro 5.1: Resultados de la implementación

- **El tiempo de implantación**: es el tiempo que tomo la implementación total de la metodología, desde que se tuvo la primera colaboración y se comenzó el desarrollo del proyecto, hasta la entrega final del Data Warehouse.
- **El tiempo de integración**: es el tiempo que le tomo a las aplicaciones desarrolladas ejecutar el proceso ETL desde la primera interfaz hasta la generación del Data Warehouse.
- **El porcentaje de datos correctamente integrados**: se refiere al porcentaje al porcentaje de registros que llegaron de forma correcta desde su origen hasta su destino
- **El porcentaje de errores en el modelo lógico generado**: el porcentaje de errores encontrados en el modelo lógico generado, tales como fuentes mal asignadas o no clasificadas.
- **El porcentaje de errores en el modelo físico generado**: el porcentaje de errores encontrados en el modelo físico generado, tales como fuentes mapeos o transformaciones mal realizadas.
- **El porcentaje de errores en las consultas de creación**: el porcentaje de errores encontrados en las consultas de creación, debido a una mala sintaxis.
- **El porcentaje de errores en las sentencia de inserción**: el porcentaje de errores encontrados en las consultas de inserción, debido a una mala sintaxis.

5.5. Comentarios finales

La implementación de un caso de estudio con Onto-ETL de la mano, mostró que es importante el tener una metodología bien definida y una biblioteca de componentes acordes a la misma, lo que reduce tiempos de implementación, hace el análisis mas simple y deja al descubierto los problemas verdaderamente importantes dentro del proceso ETL.

Sin duda alguna hay un sin fin de casos de estudio susceptibles de ser resueltos con la asistencia de Onto-ETL, sin embargo es necesario probar el marco de trabajo bajo diversos escenarios para poder afianzar su factibilidad. En este reporte de tesis solo se presenta un caso de estudio con los suficientes elementos para evaluar la factibilidad del modelo propuesto.

Algunas de las métricas empleadas para medir el desempeño de la implementación fueron tomadas de [58, 59], con el objetivo de ganar parcialidad en las aseveraciones acerca de los resultados obtenidos.

Capítulo 6

Conclusiones y Trabajo a futuro

El presente trabajo representa un esfuerzo para simplificar las tareas del proceso ETL aprovechando las bondades que ofrece la Gestión de Conocimiento. En la actualidad hay enfoques basados en ontologías para atacar la problemática del proceso ETL, la mayoría de ellos tienden a tener éxito de forma parcial debido a diversos factores. Algunos ejemplos de estos inconvenientes son: la falta de una fuente de conocimiento de calidad en la creación y definición de la ontología, ya que por lo general en los enfoques planteados la ontología es elaborada por una persona que no es experta en el dominio de la ontología que se esta definiendo. Asimismo encontramos inconvenientes en herramientas ETL provenientes de la industria, la mayoría de ellos propone esquemas donde las transformación y relaciones de las fuentes de datos origen a los contenedores finales de datos queda en manos del usuario, ademes de ser costosos, difíciles de considerar y sin una metodología claramente definida. Analizando los enfoques ETL basados y no en ontologías, encontramos las principales tendencias, deficiencias y puntos a mejorar, dando como resultado este trabajo de investigación.

En las siguientes dos secciones hablaremos de las conclusiones obtenidas y aportaciones realizadas después del desarrollo del trabajo de investigación para finalmente hablar del trabajo que queda por hacer con la finalidad de extender y mejorar Onto-ETL.

6.1. Aportaciones y conclusiones

La investigación realizada mostró el gran numero de enfoques para realizar la Gestión de Conocimiento y atacar la problemática del proceso ETL con ontologías, así mismo se mostró la complejidad del proceso ETL y las razones principales por las que los enfoques actuales no han tenido existo en la solución del problema de interoperabilidad que presenta el proceso. La mas importante como se ha remarcado, surge debido a la que la construcción y definición de la ontología siempre se hace sin consultar a los expertos en el dominio del conocimiento que se define, esto ocasiona que las ontologías sean modeladas en función el problema y no el problema sea resuelto en función de la ontología. En otras palabras el conocimiento y definición de la ontología

debe de ser independiente del problema que se resuelve puesto que es algo general y una ves que es definido por un experto de dominio se convierte como una regla a seguir.

Otro problema que presentan las soluciones basadas en ontologías es lo contrario a lo definido en el párrafo anterior, se proponen esquemas y definiciones del proceso ETL con base a ontologías que resultan ser muy idealistas y difícilmente aplicables a casos prácticos.

Como observamos es difícil mantener un balance entre la correcta definición conceptual del proceso ETL basada en ontologías y la incorrecta o relajada definición de la ontología pero la resolución parcial del problema ETL.

El principal aporte de esta tesis es el enfoque integral con el que se aborda la problemática central del proceso ETL con ayuda de ontologías, ya que presenta a diferencia de los enfoques actuales, un balance entre una correcta definición de la ontología y la solución a la problemática del proceso ETL, por tal motivo se presenta en el trabajo, una metodología para la captura del conocimiento y otra para poder llevar el proceso con ayuda de las ontologías, siendo ambas complementadas con la librería de componentes software.

El segundo aporte central de la tesis radica en la forma de obtener y capturar el conocimiento por medio de la Gestión de Conocimiento, pero a diferencia de los enfoques clásicos, esta ves se acota y define para resolver un objetivo especifico, el cual es la definición de una entidad reguladora entre fuentes de información heterogénea para facilitar el intercambio de información.

El tercer gran aporte son los problemas sobre los cuales trabajan las ontologías dentro del proceso propuesto, dentro de las cuales destacan: clasificación de los metadatos de las fuentes de información origen, la identificación y propuesta de las transformaciones que se aplicaran a los datos, la clasificación y asignación de los destinos dentro del almacén datos final y finalmente métodos para la creación de rutinas de extracción y de inserción de la información dentro del proceso, a diferencia de enfoques actuales basados en ontologías, solo se hacen propuestas a nivel conceptual. Adicional a esto el resultado final del proceso es un Data Warehouse cuyo modelo esta basado en una ontología.

Ademas de los aportes antes mencionados, este trabajo presenta algunas aportaciones respecto a los enfoques actuales en el área:

1. Se hace un análisis del proceso de Gestión de Conocimiento con ayuda de las ontologías, con la finalidad de extraer detalles como: el patrón que definen sus metodologías, tareas y elementos específicos; para así proponer una metodología con las mejores prácticas de los enfoques existentes.

2. Se hace un análisis de los enfoques ETL basados y no en ontologías, se analizan a nivel conceptual, a nivel de configuración e implementación, para poder comprender a fondo el proceso, su problemática y sus puntos clave a favor y en contra. Esto para poder extraer el comportamiento típico de éste, su ciclo de vida y sus elementos centrales, con base a ello se propone una metodología que

reúne los mejores procesos y elementos para definir el proceso ETL basado en ontologías.

3. A diferencia de enfoques actuales, se propone un marco de trabajo desde una perspectiva que considera dos grandes partes del proceso ETL: la definición del dominio de conocimiento de las fuentes de información y el uso de este conocimiento para poder realizar el proceso ETL.

4. El marco de trabajo Onto-ETL a diferencia de enfoques similares, define no solo componentes software de una biblioteca, sino también las metodologías para poder llevar a cabo los procesos de Gestión de Conocimiento y ETL.

5. Una arquitectura de clases basada en capas, que empalma con los diversos niveles de interoperabilidad que presenta el proceso ETL y que también toma en cuenta el proceso de Gestión de Conocimiento.

6. Se consideraron como posibles fuentes de orígenes, bases de datos de Oracle, MySQL, PostgreSQL, archivos XML y archivos CSV y se proponen objetos que encapsulan y abstraen tareas de extracción de meta-información, información y transformación.

7. Se propone un algoritmo de clasificación, etiquetado semántico y emparejamiento de meta-datos para bases de datos, archivos XML y CSV con ayuda de las ontologías.

8. Se propone un algoritmo para transformación de información sobre bases de datos, archivos XML y CSV basado en meta-datos y ontologías.

9. Se tomaron algunas fuentes de información y se sometieron a la metodología propuesta, para mostrar la factibilidad del marco de trabajo propuesto.

Al poder implementar el marco de referencia sobre un caso de prueba, se pudo constatar que el proceso ETL es una actividad compleja de implementar, pero teniendo como guía las metodologías y bibliotecas de Onto-ETL se puede llevar a cabo de forma mas ágil.

Otro punto que se pudo constatar es que las organizaciones ven la definición de un repositorio de conocimiento (ontología), como una perdida de tiempo, pero al poder constatar algunas de las ventajas que se pueden obtener se terminan convenciendo de que es mas una inversión.

No existen dentro de la literatura métricas o modelos para un análisis entre metodología para el proceso ETL, lo que hace a Onto-ETL y sus implementaciones algo único.

6.2. Trabajo a futuro

Onto-ETL puede ser visto como un punto de partida para la correcta definición y administración de la información, los datos y el conocimiento en sistemas analíticos en linea que trabajan con grandes volúmenes de información basados en Data Warehouse. Puesto que Onto-ETL trabaja con base a los meta-datos, ademas de que gestiona y organiza los datos la información y el conocimiento.

Los resultados obtenidos de la definición del marco de trabajo son buenos pero aun se pueden hacer mejoras y extensiones de este como las siguientes:

- Ampliar el modelo de la ontología y redefinir el proceso de captura del conocimiento para poder ampliar el rango de conocimientos a ser almacenado.
- Agregar el soporte para que el marco de referencia permita consultas de sistemas externos y así poder aprovechar de otras formas el conocimiento almacenado.
- Realizar la Gestión de Conocimiento sobre ambientes cooperativos, es decir diseñar la metodología de forma tal que permita la construcción de la ontología por parte de múltiples usuarios.
- Agregar el soporte para fuentes de información que no se tomaron en cuenta para el proceso ETL.
- Aprovechar el modelo del Data Warehouse basado en ontologías, para poder realizar tareas tales como:
    - Uso de razonadores para encontrar patrones o tendencia de información.
    - Generación de conjuntos de datos personalizados
    - Generación de rutinas para verificar la integridad y consistencia de la información.
- Generar enfoques para gestionar la evolución histórica de los datos.

Bibliografía

[1] P. Ponniah. *Data warehousing fundamentals: a comprehensive guide for IT professionals*. Number v. 1. Wiley, 2001.

[2] R. Kimball and R. Merz. *The data Webhouse toolkit: building the Web-enabled data warehouse*. Wiley computer publishing. John Wiley & Sons, 2000.

[3] R. Kimball, M. Ross, W. Thornthwaite, J. Mundy, and B. Becker. *The Kimball Group Reader: Relentlessly Practical Tools for Data Warehousing and Business Intelligence*. John Wiley & Sons, 2010.

[4] R. Kimball and J. Caserta. *The data warehouse ETL toolkit: practical techniques for extracting, cleaning, conforming, and delivering data*. ITPro collection. Wiley, 2004.

[5] Inc. ACM. The 1998 acm computing classification system.

[6] E F Codd, S B Codd, and C T Salley. Providing olap (on-line analytical processing) to user-analysts: An it mandate. *Codd and Date*, 32:31, 1993.

[7] L.L. Reeves. *A manager's guide to data warehousing*. Wiley Pub., 2009.

[8] Marko Niinimäki and Tapio Niemi. An etl process for olap using rdf/owl ontologies. In Stefano Spaccapietra, Esteban Zimányi, and Il-Yeol Song, editors, *Journal on Data Semantics XIII*, volume 5530 of *Lecture Notes in Computer Science*, pages 97–119. Springer Berlin Heidelberg, 2009.

[9] Panos Vassiliadis, Alkis Simitsis, Panos Georgantas, and Manolis Terrovitis. A framework for the design of etl scenarios. In *CAISE 03*, pages 520–535, 2003.

[10] Dimitrios Skoutas and Alkis Simitsis. Ontology-based conceptual design of etl processes for both structured and semi-structured data. *Int. J. Semantic Web Inf. Syst.*, 3(4):1–24, 2007.

[11] K Wiig. Knowledgemanagement: Where did it come from and where will it go?. *Expert Systems with Applications*, 14:23–26, 1997.

[12] Narasimhaiah Gorla. Features to consider in a data warehousing system. *Commun. ACM*, 46:111–115, November 2003.

[13] Anne Geraci. *IEEE Standard Computer Dictionary: Compilation of IEEE Standard Computer Glossaries*. IEEE Press, Piscataway, NJ, USA, 1991.

[14] W.H. Inmon, B.K. O'Neil, and L. Fryman. *Business metadata: capturing enterprise knowledge*. Elsevier/Morgan Kaufmann, 2008.

[15] Schnurr H.P. Studer R. Sure Y Staab, S. Knowledge processes and ontologies. *IEEE Intelligent Systems*, 16:1, 2001.

[16] King M Uschold, M. Towards a methodology for building ontologies. *Workshop on Basic Ontological Issues in Knowledge Sharing*, 5:56–59, 1995.

[17] Vanwelkenhuysen J. Ikeda M. Mizoguchi, R. Towards very large knowledge bases: Knowledgebuilding and knowledge sharing. *Task Ontology for Reuse of Problem Solving Knowledge*, II:46–59, 1995.

[18] D.E. O'Leary. Reengineering and knowledge management. *Knowledge Acquisition, Modeling and Management, Lecture Notes in Artificial Intelligence*, 1621:1–12, 1999.

[19] Oscar Corcho, Mariano Fernández-López, and Asunción Gómez-Pérez. Methodologies, tools and languages for building ontologies: where is their meeting point? *Data Knowl. Eng.*, 46:41,64, July 2003.

[20] Hewlett-Packard. Jena team. Jena a semantic web framework for java, 10 2011.

[21] Csongor Nyulas, Martin J. O'Connor, Samson W. Tu, David L. Buckeridge, Anna Okhmatovskaia, and Mark A. Musen. An ontology-driven framework for deploying jade agent systems. In *IAT*, pages 573–577. IEEE, 2008.

[22] The Apache Software Foundation. Xerces java parser, 6 2011.

[23] GlassFish Community. Xml schema object model, 6 2011.

[24] Stanford Center for Biomedical Informatics Research. The protege ontology editor and knowledge acquisition system, 06 2011.

[25] I. Kant. *Lectures on metaphysics - Part III Metaphysik L2*. Cambridge University Press, 2001.

[26] Fikes R.E. Finin T. Gruber T.R. Senator T. Swartout W.R. Neches, R. Enabling technology for knowledge sharing. *AI Magazine*, 3:36–56, 1991.

[27] T. R. Gruber. A translation approach to portable ontology specifications. *Knowledge Acquisition*, 5:199–200, 1993.

[28] Guha R.V Lenat, D.B. *Building large knowledge-based systems*. Addison-Wesley Publising Company, 1990.

[29] M.S Gruninger, M. Fox. *The logic of enterprise modelling In J. Brown & D.O.* Chapman & Hall, 1995.

[30] Laresgoiti I. Corera J. Bernaras, A. Building and reusing ontologies for electrical network applications. *In Proceedings of the European Conference on Artificial Intelligence*, 196:298–302, 1996.

[31] Patil R. Knight K. Russ T. Swartout, B. Toward distributed use of large-scale ontologies. *Spring Symposium Series on Ontological Engineering*, 3:1–97, 1997.

[32] Biebow B. Szulman S Aussenac-Gilles, N. Modelling the travelling domain from a nlp description with terminae. *Workshop on Evaluation of Ontology Tools, European Knowledge Acquisition Workshop*, 1:70–78, 2002.

[33] Heflin J. Luke, S. Shoe 1.01 proposed specification shoe project, 02 2000.

[34] Webick R Lassila, O. Resource description framework (rdf) model and syntax specification, 06 2002.

[35] P. Kent. Conceptual knowledge markup language (version 0.2), 06 2003.

[36] Chaudhri V. Thomere J Karp, R. Xol: An xml-based ontology exchange language, 05 2004.

[37] Fensel D. Broekstra J. Decker S. Erdmann M. Goble C. van Harmelen F. Klein M. Staab S. Studer R. Motta E. Horrocks, I. *OIL: The Ontology Inference Layer. Technical Report.* Vrije Universiteit Amsterdam, Faculty of Sciences, 2000.

[38] Hibbard J. Knowing what we know. *InformationWeek*, 1997 October 20.

[39] E. Turban. *Expert Systems and Applied Artificial Intelligence.* Macmillan, 1992.

[40] Takeuchi H. Nonaka, I. *The Knowledge Creating Company: How Japanese Companies Create the Dynamics of Innovation.* Oxford University Press, 1995.

[41] F. Martín-Rubio. *La gestión del conocimiento corporativo: una tecnología emergente.* Universidad de Murcia., 1998.

[42] E. F. Codd. A relational model of data for large shared data banks. *Commun. ACM*, 13:377–387, June 1970.

[43] Oracle Inc. Warehouse builder 11gr2 home page on otn, 2011.

[44] Pentaho Corporation. Business analytics and business intelligence leaders - pentaho, 2001.

[45] Microsoft Corporation. Data integration services microsoft sql server 2008 r2.

[46] IBM. Ibm - data integration, connectivity, scalable platforms - infosphere datastage - software.

[47] Informatica Corporation. Informatica powercenter etl software to enhance your etl process, 06 2011.

[48] SAS Institute Inc. Etl (extraction, transformation and loading) and elt sas, 06 2011.

[49] Jiarui Ni Longbing Cao and Dan Luo. Ontological engineering in data warehousing. *Frontiers of WWW Research and Development.*, 3841:923–929, 2006.

[50] Zhang C.Z. Liu J.M. Cao, L.B. Ontology-based integration of business intelligence. *Web Intelligence and Agent Systems*, 4:4–10, 2006.

[51] Diana Maynard Horacio Saggion, Adam Funk and Kalina Bontcheva. Ontology-based information extraction for business intelligence. *The Semantic Web*, 4825:843–856, 2007.

[52] Yi-Chuan Lu Hilary Cheng and Calvin Sheu. An ontology-based business intelligence application in a financial knowledge management system. *Expert Systems with Applications*, 36:3614–3622, 2009.

[53] Mitsuru Ikeda, Kazuhisa Seta, and Riichiro Mizoguchi. Task ontology makes it easier to use authoring tools. In *Proceedings of the 15th international joint conference on Artifical intelligence - Volume 1*, pages 342–347, San Francisco, CA, USA, 1997. Morgan Kaufmann Publishers Inc.

[54] Mizoguchi R. Vanwelkenhuysen J. Ikeda M. Task ontology for reuse of problem solving knowledge. *Towards Very Large Knowledge Bases Knowledge Building and Knowledge Sharing*, pages 46–59, 1995.

[55] Peter Bartalos and Maria Bielikova. An approach to object-ontology mapping. In *Slovak University of Technology*, pages 9–16, 2007.

[56] Maynard D. Bontcheva K. &Tablan V Cunningham, H. Gate: A framework and graphical development environment for robust nlp tools and applications. *AnnualMeeting of the Association for Computational Linguistics*, 40:169–175, 2002.

[57] Nadine Cullot, Raji Ghawi, and Kokou Yétongnon. Db2owl : A tool for automatic database-to-ontology mapping. In Michelangelo Ceci, Donato Malerba, and Letizia Tanca, editors, *SEBD*, pages 491–494, 2007.

[58] Len Wyatt, Brian Caufield, and Daniel Pol. Principles for an etl benchmark. In Raghunath Nambiar and Meikel Poess, editors, *Performance Evaluation and Benchmarking*, volume 5895 of *Lecture Notes in Computer Science*, pages 183–198. Springer Berlin Heidelberg, 2009.

[59] Christian Thomsen and Torben Pedersen. A survey of open source tools for business intelligence. In A Tjoa and Juan Trujillo, editors, *Data Warehousing and Knowledge Discovery*, volume 3589 of *Lecture Notes in Computer Science*, pages 74–84. Springer Berlin Heidelberg, 2005.