UNIVERSIDAD AUTÓNOMA METROPOLITANA

Casa abierta al tiempo

Construcción de las capacidades semánticas para un sistema de almacenamiento distribuido

Para obtener el grado de
Maestra en Ciencias
(Ciencias y Tecnologías de la Información)

Presenta:
Ana Bertha Rios Alvarado

Asesores

Dra. Reyna Carolina Medina Ramírez
Dr. Ricardo Marcelín Jiménez

Sinodales

Presidente: Dr. Miguel Ángel Gutiérrez Andrade
Secretario: Dra. Reyna Carolina Medina Ramírez
Vocal: Dr. Víctor Jesús Sosa Sosa

28 de Noviembre de 2008

UNIVERSIDAD AUTÓNOMA METROPOLITANA

Construcción de las capacidades semánticas para un sistema de almacenamiento distribuido

Idónea comunicación de resultados para obtener el grado de

MAESTRA EN CIENCIAS

(CIENCIAS Y TECNOLOGÍAS DE LA INFORMACIÓN)

por

Ana Bertha Rios Alvarado

Asesores:

Dra. Reyna Carolina Medina Ramírez

Dr. Ricardo Marcelín Jiménez

24 de noviembre de 2008

Con todo mi amor para:
mis padres Francisco Javier y Ana Luisa
y
mi hermano Javis

Resumen

En los últimos años la información en la Web se ha incrementado considerablemente, haciendo difícil su uso y administración, este problema también está presente en una organización, pero a una escala menor. De hecho, la representación, la búsqueda y el intercambio de información se han vuelto operaciones cruciales tanto para los individuos como para las organizaciones (grupos de investigación o empresas).

En esta tesis proponemos una metodología general para incorporar las capacidades semánticas de representación, almacenamiento y recuperación a una memoria corporativa almacenada sobre una red par a par (P2P), con el fin de gestionar los recursos documentales de un grupo de investigación, aprovechando su contenido semántico. Por capacidades semánticas, se debe entender la utilización de elementos clave de la Web semántica como son: ontologías, anotaciones y lenguajes de representación.

Nuestro trabajo se ha divido en dos etapas. La primera etapa, comprende la propuesta de una metodología para adquirir y representar el conocimiento de un grupo de investigación. Para esto, hemos construido una ontología de dominio llamada *Ontología del Área de Redes y Telecomunicaciones* (ODARyT) y una interfaz de usuario que permite la edición de ontologías de dominio y de anotaciones, así como la consulta de las anotaciones realizadas.

En la segunda etapa hemos propuesto un mecanismo semántico de almacenamiento y recuperación para los recursos del grupo de investigación. Esta propuesta incluye un algoritmo para la localización semántica de los recursos documentales del grupo de investigación, sobre una red de almacenamiento par a par (P2P), así como también el modelo de recuperación de un recurso documental. El algoritmo propuesto para la localización resuelve el problema de *gráficas empotradas* que es un problema NP-Completo, usando la metaheurística *Colonia de hormigas.*

Abstract

The huge amount of information available on the Web has become overwhelming problem, this problem is faced also by organizations but in a different scale. In fact, representation, search and exchange of information have become crucial operations, not only for individuals but also for organizations (scientific communities or companies). In this MsC thesis we focused in incorporating semantic representation, storage and retrieval capabilities in a P2P corporate memory. By semantic capabilities, we mean the key elements of Semantic Web, i.e. ontologies, annotations and those languages supporting knowledge representation.

In this MSc thesis, we have proposed a methodology for extracting and representing the existing knowledge in a scientific group (Corporate Semantic Web or memory). We have built a domain ontology called *Ontology on the Domain of Networks and Telecommunications* (ODARyT) and developed our own annotation and ontology-edition interface, this interface is able to query an annotation data base.

We have also proposed a semantic framework for storing and retrieving the document contents that make up a Corporate Semantic Web (CSW). For solving content location, we have solved the graph embedding problem. Graph embedding problem is an NP-Complete problem, then we decided using the *Ant Colony Optimization* algorithm.

Agradecimientos

"Muchas veces, a lo largo de un mismo día,
me doy cuenta de que mi propia vida y sus logros
se han construido gracias
al trabajo de las personas que me rodean.
También comprendo con cuánta seriedad
debo esforzarme para darles, en correpondencia,
tanto como he recibido"
Albert Einstein

Muchas gracias a:

Mis asesores Dra. Carolina Medina y Dr. Ricardo Marcelín, por aceptarme en este proyecto, por sus consejos, enseñanzas, apoyo y comentarios oportunos. Gracias por ser mi motivación para seguir adelante en esta profesión y por la gran oportunidad de presentar nuestro trabajo en WebMedia 2008, Brasil.

Mi tutor Dr. Miguel Pizaña por sus consejos y su tiempo, fueron de gran ayuda durante mi estancia en la licenciatura y la maestría.

La Dra. Elizabeth Pérez por sus consejos, así como, por el apoyo recibido para asistir a MICAI 2006 en Tlaxcala y la 6ta. Escuela de Otoño sobre Sistemas Distribuidos 2007 en Colima.

Mis profesores de la MCyTI por todas sus enseñanzas y conocimiento transmitido.

Mis compañeros y amigos de la MCyTI por brindarme su amistad y por todos los momentos compartidos.

Mis padres y mi familia, que siempre han creido en mi y han estado en todo momento a mi lado, apoyándome y alentándome a seguir siempre adelante aún con las adversidades que la vida nos puede presentar. Muchas gracias de todo corazón.

A.B.R.A.

Contenido

Lista de figuras

Lista de tablas

Capítulo 1

Introducción

En los últimos años la información en la Web se ha incrementado considerablemente, haciendo difícil su uso y administración. En la Web la búsqueda e intercambio de información son operaciones de vital importancia para los individuos. La recuperación de información en el enfoque clásico está basada en la ocurrencia de las palabras en un documento, esto genera resultados no pertinentes (ruido), en el contexto de la consulta, y conlleva a que el usuario tenga que revisar y seleccionar los documentos que le son de interés. Problemas implícitos en la búsqueda de información tenemos: la sinonimia (términos que significan lo mismo pero que por no estar en el documento pero sí en la consulta, no son regresados como respuesta) y la ambigüedad (términos que dependiendo del contexto adquieren diferentes significados).

Este tipo de problemas nos muestran que debe considerarse la semántica en la búsqueda e indización de los recursos a ser almacenados y recuperados por algún sistema de búsqueda.

La Web semántica [1] aparece como la próxima generación de la Web donde el objetivo es dar a la información una representación semántica con el fin de hacerla accesible y entendible no sólo por los humanos sino también por las computadoras. Este enfoque se apoya en las ontologías (para la búsqueda e intercambio de información), en las anotaciones (para la representación del contenido de documentos) y en lenguajes de representación de conocimientos (para la representación de ontologías y anotaciones).

Por otro lado, dentro de una organización (empresa o grupo de investigación) encontramos los mismos problemas y necesidades de administración, uso, visualización, búsqueda e intercambio de información a una escala menor que en la Web. Asi mismo, de igual forma que en la Web, dentro de una organización se requiere tener un acceso a la información de forma fácil y eficiente.

Una memoria corporativa es una representación explícita, incorpórea, y persistente del conocimiento en una organización, para facilitar su acceso, intercambio y reutilización por los miembros de una organización para la realización de sus tareas [2]. El objetivo principal en la construcción de una memoria corporativa es la integración coherente del conocimiento disperso, con el objetivo de

estimular la capitalización del conocimiento y en general preservar el conocimiento dentro de una organización.

La Web Semántica provee mecanismos interesantes para concretar y estructurar las memorias corporativas, facilitar su administración y visualización [3].

En el contexto de la búsqueda de información documental, la representación del contenido de los documentos utilizando los términos pertenecientes a una ontología para una explotación y gestión posterior, parecen adecuados para obtener resultados pertinentes a las necesidades de los usuarios de esta base documental.

Los sistemas par a par (P2P) utilizan recursos distribuidos para realizar una función crítica de una manera descentralizada [4] y han traído consigo una gran revolución en términos de nuevas aplicaciones distribuidas. Las características principales de los sistemas P2P son: escalabilidad, robustez, descentralización, costos distribuidos, anonimato, seguridad, persistencia de información, balance de carga y localización rápida de los recursos. El problema de la búsqueda en un sistema P2P consiste en encontrar un recurso, sin servidores centralizados y considerando aspectos de escalabilidad. Los nodos son autónomos en la forma en cómo almacenan la información de manera local. Cada nodo en la red P2P puede ser cliente o servidor, puede comunicarse directamente con otro nodo y es autónomo.

Considerando las características de escalabilidad, localización rápida de los recursos y descentralización, que ofrecen los sistemas P2P, se pueden aprovechar estas características para el almacenamiento y recuperación basados en la semántica de los recursos documentales de una memoria corporativa.

En particular se propone incorporar las capacidades semánticas de almacenamiento y recuperación a un sistema par a par (P2P), para almacenar y recuperar los recursos documentales de una memoria corporativa, aprovechando su contenido semántico. Por capacidades semánticas se debe entender la utilización de elementos clave de la Web semántica (WS), como son: anotaciones, ontologías y lenguajes de representación del conocimiento.

El objetivo general de este trabajo de tesis es construir una capa semántica sobre una red P2P para el almacenamiento y recuperación de los recursos de una memoria corporativa. Los objetivos especificos son:

- Representar el conocimiento de un grupo de trabajo a través de la adaptación y/o construcción de una ontología de dominio.
- Describir semánticamente los recursos del grupo de trabajo.
- Proponer un modelo semántico para el almacenamiento y la recuperación de los recursos.

La investigación desarrollada en esta tesis se ha dividido en dos etapas. La primera etapa del proyecto consiste en la adquisición y representación semántica del conocimiento de un grupo de trabajo. Esta comprende 1) la caracterización de los recursos disponibles del grupo de trabajo, de sus requerimientos de almacenamiento y necesidades de recuperación, 2) la adaptación y/o construcción de una ontología de dominio 3) la descripción semántica de los recursos disponibles en el grupo de trabajo y 4) la validación de la representación semántica propuesta.

La segunda etapa del proyecto describe el modelo semántico propuesto para el almacenamiento y la recuperación de los recursos de un grupo de trabajo sobre una red P2P. En particular se desarrolló un algoritmo para resolver el problema de *empotrado de gráficas* basado en la metaheurística denominada *Colonia de hormigas* y 2) la implementación y evaluación del algoritmo propuesto a través de un simulador.

El presente documento se encuentra organizado de la siguiente manera:

En el Capítulo 2 se presenta una breve descripción de trabajos que integran el manejo de la información con la semántica y la arquitectura par a par.

La primera parte de este trabajo de tesis, descrito en el Capítulo 3, aborda la adquisición y representación de los recursos de una memoria corporativa, en particular del grupo de investigación de Redes y Telecomunicaciones. También se describe la metodología usada para la construcción de una ontología de dominio, llamada ODARyT (Ontología del Area de Redes y Telecomunicaciones) y el desarrollo de una interfaz para la edición de ontologías y para la edición y consulta de anotaciones.

La segunda parte de este trabajo de tesis se presenta en el Capítulo 4, el cual describe el modelo de almacenamiento y recuperación de la información sobre una arquitectura par a par. Esta parte comprende el desarrollo de un algoritmo que resuelve el problema de *empotrado de gráficas* basado en la metaheurística *Colonia de Hormigas*. Se presenta el algoritmo, su implementación y su evaluación a través de un simulador de eventos discretos.

Finalmente, en el Capítulo 5, encontramos las conclusiones generales y las perspectivas futuras acerca de este trabajo de investigación.

Capítulo 2

Estado del arte

La integración del manejo de la información con un enfoque semántico en una arquitectura par a par (P2P) ha surgido recientemente. Existen algunos trabajos que incorporan estas dos temáticas. Entre los trabajos considerados en este estado del arte y que se describen en las secciones siguientes, se tomaron en cuenta tres enfoques:

- Gestión de Conocimiento Distribuido (GCD) [5] se refiere a los procesos de adquisición, ordenamiento, análisis, intercambio y difusión del conocimiento dentro de una organización de manera descentralizada. Para hacer más eficientes estos procesos, se ha incluido el uso de nuevas tecnologías que permiten llevar a cabo tales procesos. En este sentido se han propuesto sistemas que incluyen la semántica para la gestión del conocimiento en un sistema distribuido P2P. En la Sección 2.1 se describe un sistema de gestión de conocmiento distribuido sobre una red P2P.

- Búsqueda de información en sistemas P2P. La indización y recuperación de resultados son aspectos importantes en la búsqueda de información. El representar el componente semántico que tienen los recursos almacenados en tales sistemas ha sido abordado desde diferentes puntos de vista. En la Sección 2.2 describimos algunos enfoques en la búsqueda de información.

- Web semántica y sistemas P2P. En este tipo de trabajos el componente semántico de los recursos es representado a través de ontologías (vocabularios conceptuales). En la Sección 2.3 se abordan algunos sistemas importantes que integran la Web semántica con los sistemas P2P para gestionar la información almacenada en tales sistemas.

2.1. Gestión de Conocimiento Distribuido (GCD)

La Gestión de Conocimiento Distribuido (GCD) [6] se refiere a los procesos de adquisición, ordenamiento, análisis, intercambio y difusión del conocimiento (esto es, documentos, bases de datos,

repositorios de información) dentro de una organización de manera descentralizada, con el fin de aprovechar el conocimiento. La idea de descentralización surge del hecho, de que las organizaciones estan integradas por diferentes grupos autónomos (departamentos, grupos de trabajo, etc.) que generan su propio conocimiento. Este conocimiento necesita ser difundido, usado e intercambiado entre los miembros de la organización, para ello se han propuesto sistemas que permiten la gestión del conocmiento de manera distribuida. Bonifacio [6] propuso un sistema llamado *Knowledge Exchange System* (KEEx) el cual esta basado en la tecnología P2P que sigue la idea de la GCD. La plataforma KEEx otorga a los nodos del sistema P2P un alto nivel de autonomía para gestionar su conocimiento local. En KEEx, cada nodo contiene conocimiento y es representado por un nodo llamado K-peer. Cada nodo proporciona todos los servicios requeridos para crear y organizar el conocimiento local, y define estructuras para alcanzar una coordinación a nivel semántico, es decir, buscar documentos en nodos remotos. Tiene un componente controlador que se encarga de la distribución de las consultas. La decisión sobre a que pares debe enviarse la consulta se hace aplicando un algoritmo *Selector de Pares* basado en el conocimiento previo que posee sobre estos. La *interfaz de usuario* permite visualizar el conocimiento en diferentes formas: jerarquía de conceptos, mapas temáticos, etc.

2.2. Búsqueda de información en sistemas P2P

En el contexto de la búsqueda de información, recientemente se han incorporado a los sistemas P2P *índices semánticos* que hacen referencia a un documento. Según Risson y Moors [7] proponen dos mecanismos de indización semántica, una de ellas es *Vector Space Model* (VSM) y la otra es *Latent Semantic Indexing* (LSI). El VSM representa documentos y consultas como vectores de términos, donde un término puede ser una palabra o una frase. La recuperación del documento se hace a través del empate entre el VSM de una consulta particular con el VSM que describa a algún documento almacenado.

LSI [8] es un mecanismo utilizado para la recuperación de información. Además de guardar las palabras claves que caracterizan a un documento, este método examina todos los documentos de la colección, para saber que otros documentos contienen algunas de esas palabras claves. LSI considera que los documentos que tienen muchas palabras claves en común, tienen cercanía semántica y los que tienen pocas palabras en común son semánticamente distantes. Este mecanismo simple, correlaciona documentos considerando su contenido. Así la recuperación de información, regresa como resultado los documentos que son semánticamente cercanos, aunque algunos de ellos no contengan las palabras claves que conforman la consulta inicial.

Estos dos mecanismos de indización aún se quedan lejos de aprovechar el contenido semántico de un documento. VSM y LSI, no incorporan mecanismos de inferencia del conocimiento, ni

tampoco modelan el conocimiento utilizando las herramientas propuestas por la Web semántica. PeerSearch [9] es un ejemplo de sistemas que utilizan los mecanismos VSM y LSI para la indización de documentos.

2.3. Web semántica y sistemas P2P

La integración de la Web semántica con sistemas P2P para gestionar la información almacenada en tales sistemas, ha dado lugar a los sistemas *Peer Data Management System* (PDMS).

Los sistemas PDMS tienen como base la arquitectura P2P y con ella todas las características que ofrecen los sistemas P2P. En un PDMS cada nodo (llamado peer) tiene su fuente local de conocimiento (que puede ser su propio sistema de archivos local, direcciones de correo electrónico, entre otras) y representa el conocimiento local del peer que puede intercambiar a través de la red.

Por otro lado en un PDMS, cada peer tiene asociado un *esquema*. Un esquema es una especificación de la estructura que representa el dominio de interés del peer. Las *relaciones semánticas* entre peers son provistas localmente entre peers (o pequeños conjuntos) de peers. Para recorrer las trayectorias semánticas del mapeo de la red, una consulta sobre un peer puede obtener datos relevantes de cualquier peer alcanzable en la red. Las trayectorias semánticas son trazadas por reformulación de consultas a un peer y las consultas ejecutadas a sus vecinos [7]. Los PDMS's representan un paso natural en la integración de sistemas reemplazando los esquemas locales, con una integración de mapeo semántico entre esquemas de peers individuales. A continuación mencionamos algunos ejemplos representativos de los sistemas *Peer Data Management System*:

Piazza [11] es un sistema que consiste de una red de diferentes sitios (también llamados nodos o peers), cada uno de ellos contribuye con recursos al sistema. Piazza trabaja en una escala global, y ofrece un lenguaje de mediación entre los recursos del sistema, para hacer el mapeo de la estructura del dominio y la estructura de los documentos. Soporta dos mecanismos para la mediación semántica: 1) *mediación por mapeo*, donde los recursos son relacionados a través de un esquema de mediación o una ontología, y 2) mapeos punto a punto, donde los datos son descritos conforme al esquema de otro sitio. Cada nodo tiene un esquema, expresado en XML Schema, el cual define los términos y las restricciones estructurales del nodo. Cada vez que un nodo se agrega a la red, debe pasar por el proceso de mediación. Piazza es un sistema que trabaja a nivel global, lo que caracteriza la heterogeneidad de los recursos, así como de los esquemas conceptuales de los mismos, esto lleva a tener diferentes estructuras que deben ser transformadas a un esquema común, de ahí la necesidad de un *esquema de mediación*. El esquema de mediación genera costos por el envió de mensajes a través de los peers. Por otro lado, si cada uno de los peers no conoce el esquema global de la semántica del dominio, la consulta, en su trayectoria a través del sistema, puede acceder a nodos que no contienen

la información solicitada, generando costos de tiempo de ejecución en la recuperación de resultados.

Edutella [12] es un sistema que define una infraestructura basada en metadatos Resource Description Framework (RDF) [37] para aplicaciones P2P, es capaz de integrar peers heterogéneos (en términos de funcionalidad, esto es, servicios que ofrecen), así como diferentes tipos de esquemas de metadatos. Edutella ofrece una serie de servicios como consulta, replicación, mapeo, mediación y agrupamiento para el manejo de recursos. Los recursos almacenados en la red son descritos en RDF. La funcionalidad en la red Edutella es mediada a través de enunciados RDF y consultas sobre ellos. Los metadatos RDF se encuentran almacenados en repositorios RDF distribuidos y el lenguaje de consulta es transmitido en formato RDF/XML. Al ser Edutella un sistema que ofrece una estructura para las aplicaciones P2P basada en metadatos RDF, requiere incorporar servicios de mediación y mapeo entre grupos de peers usando un esquema común. Como Edutella trabaja a una escala global, la consulta del usuario se debe traducir a ese esquema común. Sin embargo, al igual que Piazza, el mapeo no contempla que los nodos conozcan el esquema global, lo que provoca que el mecanismo de mediación y de consulta, genere costos en tiempo de ejecución, al recuperar los resultados.

RDFPeers [13] es un repositorio distribuido y escalable de descripciones RDF. Permite la indización, almacenamiento y recuperación de sentencias RDF. En este sistema cada uno de los nodos conoce que nodo es responsable de almacenar las tripletas RDF que son distribuidas por sus elementos (sujeto, predicado, objeto), esto es, cuando una tripleta RDF es insertada en la red, esta será almacenada tres veces aplicando una función *hash* para cada uno de los valores de sujeto, predicado y objeto, y cada nodo conocerá el nodo al que fueron asignados los elementos de la tripleta, que puede estar solicitando, esto permite encaminar la consulta hacia el nodo apropiado. RDFPeers es un sistema con un mecanismo de almacenamiento distribuido que garantiza la recuperación de las tripletas RDF, usadas como descripciones de los recursos almacenados, son necesarias para la recuperación de los recursos, lo que su distribución garantiza confiabilidad. Este sistema aporta la idea de mantener el conocimiento global en cada uno de los nodos de la red, para enviar la consulta al nodo apropiado.

Por otro lado en el trabajo de Owens [14] se describen sistemas de almacenamiento semántico como Sesame, 3Store, RDFStore, Kowari, y Jena2, donde el almacenamiento semántico se refiere a almacenar y gestionar las descripciones semánticas de los recursos. Estos sistemas se ejecutan sobre modelos centralizados, lo que lleva a que sean vulnerables a perder la base de anotaciones, las cuales son necesarias para la recuperación de la información.

Finalmente existen proyectos como [15] y [16] que proponen la creación de capas semánticas sobre los datos almacenados en los nodos de una red P2P. En SOWES [15] el proceso de *clustering* esta basado en vectores de palabras clave que caracterizan el contenido de los documentos. Primero

se crean zonas, a través de un algoritmo distribuido, donde cada zona tiene asociados varios nodos. Después cada zona recolecta los vectores de sus nodos y crea nuevos clusters basados en las similitudes de los vectores. En [16] se propone relacionar semánticamente los nodos de una red, es decir crear *Semantic Overlay Network (SON)*, sobre un sistema de almacenamiento de archivos de música. Lo cual permite que las consultas de los usuarios sean redireccionadas sobre las apropiadas SON's. La creación de una SON, esta basada en la clasificación jerárquica de conceptos asociados a las características semánticas de los archivos. Los archivos son *clasificados* en su respectiva SON. Para la búsqueda, dada una consulta de un usuario, esta es clasificada, es decir, asociada a un concepto clave de la jerarquía de conceptos, y a través de un recorrido en profundidad sobre la jerarquía se localiza el nodo asociado entre la consulta y la SON solicitada.

Capítulo 3

Parte I. Adquisición y representación del conocimiento en una memoria corporativa

3.1. Introducción

En la actualidad la búsqueda e intercambio de información se han vuelto de vital importancia tanto para los individuos como para las organizaciones. En particular, en un grupo de investigación, el acceso, uso e intercambio de la información son esenciales para su crecimiento y la generación de nuevo conocimiento. En un grupo de investigación, una fuente de información puede materializarse en recursos documentales tales como son tesis, artículos, reportes, entre otros, así como también los propios investigadores representan una fuente de conocimiento. Aprovechando los elementos clave que provee la Web semántica (ontologías, anotaciones y lenguajes de representación), se pretende representar el conocimiento de un grupo de investigación para la gestión posterior de sus recursos documentales.

3.2. Memoria corporativa

Una memoria corporativa es una representación explícita, incorpórea, y persistente del conocimiento en una organización, para facilitar su acceso, intercambio y reutilización por los miembros de una organización para la realización de sus tareas [2]. El objetivo principal en la construcción de una memoria corporativa es la integración coherente del conocimiento disperso, su capitalización del conocimiento y preservarlo dentro de una organización. El conocimiento de un grupo de investigación puede ser visto como una memoria corporativa.

Existe una relación entre las memorias corporativas u organizacionales y la Web [17, 18]. Ambas

afrontan necesidades y problemas similares pero a escalas diferentes: recuperación pertinente de información, información distribuida y heterogénea tanto en contenido como en formato, actualización continua de la información, visualización de los documentos pertinentes a una consulta, por mencionar solo algunos. La Web semántica [1] aparece como el siguiente paso en la evolución de la Web donde el objetivo es dar a la información una representación semántica con el fin de hacerla accesible y entendible no sólo por las personas sino también por las computadoras. Este enfoque se apoya en las *ontologías* (para la búsqueda e intercambio de información), en las *anotaciones* (para la representación del contenido de documentos) y en *lenguajes de representación* del conocimiento (para la representación de ontologías y anotaciones).

La Web semántica provee mecanismos interesantes para concretar y estructurar las memorias corporativas, facilitar su administración y visualización [3]. Es en el marco anterior que surge el concepto de *Web Semántica Corporativa*, que esta integrada por: *ontologías*, *recursos* (tales como documentos o personas) y *descripciones semánticas* sobre esos recursos. Tales descripciones conocidas comunmente como anotaciones se apoyan en el contenido de los documentos o las caracteristicas y/o habilidades de las personas a describir. Las anotaciones semánticas están basadas en el vocabulario de las ontologías. En la figura 3.1 se muestran los elementos de una Web Semántica Corporativa.

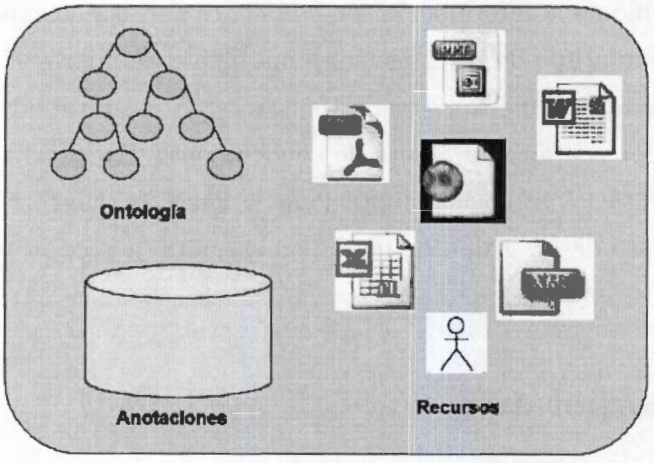


Figura 3.1: Elementos de una Web Semántica Corporativa

En el contexto de la búsqueda de información documental, la representación del contenido de los documentos utilizando los términos pertenecientes a una ontología para su aprovechamiento y gestión posterior, parecen adecuados para obtener resultados pertinentes a las necesidades de los usuarios de esta base documental.

En este proyecto se ha seleccionado un grupo de investigación en el dominio de Redes y Telecomunicaciones. En particular, el grupo de investigación que pertence al área de Redes y Telecomunicaciones (RyT) del Departamento de Ingeniería Eléctrica de la UAM-Iztapalapa. Para construir la memoria corporativa de este grupo de investigación nos apoyamos en el conocimiento del mismo, residente en sus recursos documentales, así cómo en los propios investigadores (expertos).

La memoria corporativa que representa el conocimiento del grupo de investigación de RyT es heterogénea por varias razones: hay diferentes subdominios involucrados: Redes, Telecomunicaciones, Sistemas Distribuidos, Sistemas Digitales y gestión administrativa. Otra razón de heterogeneidad es que entre los documentos existe una gran variedad de formatos y han sido obtenidos por diferentes metodologías a través de varios años.

En particular la memoria corporativa del grupo de investigación RyT está representada por artículos, tesis, reportes técnicos, notas de curso, entre otros. Para la representación del contenido de los recursos documentales de la memoria corporativa, se propuso al grupo RyT la creación y evaluación de una ontología *ad-hoc* a su dominio de conocimiento.

Para la composición de la memoria corporativa se solicitó a los expertos que proporcionaran los recursos a gestionar (documentos digitales o en papel), además de responder un cuestionario sobre los recursos proporcionados: título del documento, autor (es), descripción, palabras clave, nombre del archivo digital, formato, año de publicación, preguntas que llevan a ese documento, así como otras referencias a las preguntas dadas. Esta colección de documentos constituye la memoria corporativa del grupo de investigación RyT.

3.3. Ontología del Área de Redes y Telecomunicaciones (ODARyT)

3.3.1. Definición de ontología

La definición de ontología se ha ido enriqueciendo en los últimos años, debido al gran auge que ha adquirido en el desarrollo de nuevas aplicaciones, como por ejemplo, en la búsqueda de información, en donde se pueden obtener resultados pertinentes. A continuación se dan algunas definiciones de ontología que han sido consideradas en esta tesis.

Gruber [19] define una ontología como una *especificación explícita de una conceptualización.* Según Studer [20] *una ontología es una especificación formal y explícita de una conceptualización compartida*, por lo que la información puede ser representada como una taxonomía de conceptos, relaciones y axiomas. Además al ser una conceptualización compartida es indispensable que exista un consenso entre los miembros del grupo que representan a los expertos en ese dominio del conocimiento.

Para el W3C [21] *Una ontología define los términos a utilizar para describir y representar un área*

de conocimiento. Las ontologías son utilizadas por las personas, las bases de datos, y las aplicaciones que necesitan compartir un dominio de información (un dominio es simplemente un área de temática específica o un área de conocimiento, tales como medicina, fabricación de herramientas, bienes inmuebles, reparación automovilística, gestión financiera, etc.). Las ontologías incluyen definiciones de conceptos básicos del dominio, y las relaciones entre ellos, que son útiles para los ordenadores [...]. Codifican el conocimiento de un dominio y también el conocimiento que extiende los dominios. En este sentido, hacen el conocimiento reutilizable.

Con base en lo anterior, en esta tesis se adopta la siguiente definición:

> **Una ontología es un modelo de representación del conocimiento de un dominio, que permite la estandarización de términos y significado de conceptos, así como también, la reutilización y organización de ese conocimiento, con fines de almacenamiento y recuperación de recursos documentales.**

En este caso para la representación semántica del conocimiento del grupo de investigación RyT se propone el desarrollo de una ontología y la descripción de los recursos (anotaciones) usando la ontología propuesta. El dominio modelado en este proyecto es el de Redes y Telecomunicaciones. La ontología propuesta se denomina *Ontología del Área de Redes y Telecomunicaciones* (ODARyT).

3.3.2. Metodología para la construcción de ODARyT

En los últimos años han surgido una gran cantidad de metodologías, lenguajes y herramientas que permiten la construcción de una ontología.

Corcho [23] hace un análisis de metodologías, herramientas y lenguajes para la construcción de una ontología y propone que para la construcción de una ontología se deben considerar tres aspectos: ¿qué metodología usar para la construcción de una ontología?, ¿qué herramienta de apoyo usar para el desarrollo de una ontología? y ¿qué lenguaje usar para la implementación de una ontología? La comparación entre las metodologías analizadas se basa en el grado de dependencia que existe entre la ontología y la aplicación final en donde se le utiliza.

Fernández [25] propone una metodología llamada METHONTOLOGY donde considera la construcción de una ontología como un proyecto en donde se deben realizar las siguientes actividades: Planificación, Especificación, Adquisición del conocimiento, Conceptualización, Formalización, Integración, Implementación, Evaluación, Documentación y Mantenimiento. Está metodología se enfoca en el mantenimiento del ciclo de vida de una ontología, permitiendo incluso que se combine con un proceso iterativo para su desarrollo.

El método IDEF5 [26], está diseñado para asistir en la creación, modificación y mantenimiento de ontologías. Esta metodología sugiere que no es prudente adoptar "una receta de cocina", para

el desarrollo de una ontología. El procedimiento general lleva a cabo las siguientes fases: establecer las especificaciones de la ontología, recolectar y analizar los datos, desarrollar una ontología inicial, refinar la ontología y validarla. Cada una de las fases propuestas puede adoptar sus propios métodos. La metodología IDEF5 es flexible, sencilla y se basa en el refinamiento de las salidas producidas en cada fase.

Por otro lado, la construcción de una ontología puede partir de la reutilización de trabajos previos, o incluso seguir una metodología que permita su construcción de manera colaborativa. Para la reutilización o adaptación de una ontología se debe tener un conocimiento previo acerca del dominio.

Actualmente no hay una metodología única para la construcción de una ontología, en este sentido, el proceso de construcción depende del dominio de conocimiento y de la aplicación que utilizará esa ontología.

Para la construcción de ODARyT, nos apoyamos en METHONTOLOGY [25] e IDEF5 [26]. Las fases de METHONTOLOGY tomadas en cuenta para este proyecto son: Especificación, Adquisición del Conocimiento, Conceptualización, Implementación y Mantenimiento. De la metodología IDEF5 se consideró la fase de refinamiento de la ontología y la fase de validación. La metodología propuesta y usada para el desarrollo de ODARyT sigue las siguientes etapas:

- *Especificación*: definir la meta, el alcance y la granularidad de la ontología.
- *Adquisición del conocimiento*: se refiere al proceso de recolectar el conocimiento, ya sea a través de libros, diccionarios, cuestionarios o entrevistas con los expertos.
- *Conceptualización*: organizar y estructurar el conocimiento adquirido, se pueden usar tablas, jerarquías o lenguajes de representación como UML(Lenguaje de Modelado Unificado).
- *Refinamiento*: identificar redundancia e inconsistencias.
- *Validación*: revisar el contenido y la estructura de la ontología.
- *Implementación*: formalizar e implementar el modelo conceptual con lenguajes formales, tales como: OWL y RDF.
- *Mantenimiento*: efectuar modificaciones, así como, agregar nuevos conceptos, relaciones y axiomas.

En la etapa de especificación, se estableció que ODARyT es una ontología que tiene como propósito el almacenamiento y recuperación semánticos de la memoria corporativa del grupo de

Elementos	Definición
Concepto	Representa un grupo de objetos o seres que comparten características que permiten sean reconocidos como miembros de ese grupo.
Grupos semánticos (jerarquía)	Clasificación basada en la semántica.
Definición de un concepto	Significado del concepto en lenguaje común.
Relación	Asociación o enlace entre conceptos generalmente expresada por un término o un gráfico.

Tabla 3.1: Elementos generales de una ontología[2]

investigación RyT. La ontología propuesta integra conceptos que permiten describir las características de los recursos documentales (título, autor, año, formato), así como su contenido, para una recuperación pertinente de la información.

Los elementos que caracterizan generalmente a una ontología son descritos en la tabla 3.1.

	Conceptos	Grupos semánticos	Definciones	Relaciones
Contenido temático	Si	Si	No	No
Taxonomia ACM	Si	Si	No	No
Diccionario de Redes	Si	Si	Si	No
Tesauro WorldBank	Si	No	No	Si
Tesauro Redes	Si	Si	No	Si
Ontología Telecomm	Si	Si	Si	Si

Tabla 3.2: Sistemas de clasificación del conocimiento

Para la etapa de adquisición del conocimiento buscamos y analizamos algunos recursos en la Web. También se analizaron seis sistemas de clasificación del conocimiento. La tabla 3.2 muestra los sistemas comparados así como los criterios establecidos para tal comparación. En dicha tabla tenemos que un *contenido temático* y la *taxonomía ACM* [27], si bien, tienen una jerarquía de conceptos y están agrupados semánticamente, no tienen definiciones, ni tampoco relaciones explícitas entre los conceptos. El *diccionario de redes* [28] (disponible en línea) tiene conceptos agrupados semánticamente y definiciones, pero no describe las relaciones entre los conceptos. Cabe mencionar que algunos libros proporcionan un *glosario de conceptos* tales como [29] y [30]. Un *tesauro* es un sistema de representación del conocimiento que contiene un listado de términos y relaciones

de equivalencia, jerarquía y asociación. Los tesauros [31] y [32] mostrados en la tabla 3.2, si bien, contienen una lista de conceptos agrupados semánticamente y describen las relaciones entre los conceptos, no presentan las definiciones de tales conceptos.

Dentro de las estructuras de representación analizadas, se encuentra una ontología de Telecomunicaciones que pertenecen al proyecto SIMS [33], esta representa el conocimiento del dominio de las Telecomunicaciones, sin embargo, se aborda desde una perspectiva comercial y, en contraste, el conocimiento del grupo de Redes y Telecomunicaciones, se inscribe en un contexto educativo y de investigación.

En cuanto a la fase de conceptualización, se tomaron en cuenta tres actividades:

- el análisis de los sistemas descritos anteriormente
- la revisión de la bibliografía recomendada por los investigadores
- entrevistas informales con los investigadores

Con las actividades anteriores obtuvimos un glosario de los conceptos más representativos del dominio. Estos conceptos se organizaron en tablas en donde se incluyeron las definiciones para cada uno de ellos. Usando una estrategia *top-down* (general a lo particular) se construyó una taxonomía. El resultado considera las tres subáreas de investigación del grupo de RyT: Redes y servicios de telecomunicaciones, Sistemas de comunicación digital y Sistemas distribuidos. Se reutilizaron algunos conceptos de la ontología de Telecomunicaciones [33] para la subárea de investigación de Sistemas de comunicación digital y se agregaron más conceptos correspondientes a la misma. Además se incorporó el grupo semántico perteneciente a los elementos de un Documento [34], estos conceptos permiten hacer descripciones más precisas de los recursos documentales, con respecto a sus características. Las relaciones establecidas entre los conceptos de ODARyT corresponden a relaciones de jerarquía. Esta fase de conceptualización tuvo una iteración lo que permitió refinar nuestra ontología.

Finalmente se validó la propuesta de ODARyT por parte de los investigadores del grupo de RyT. Las fases de implementación y mantenimiento de ODARyT se realizaron a través de la interfaz descrita en la Sección 3.5. La implementación de ODARyT está escrita en lenguaje OWL [36].

3.3.3. Descripción de ODARyT

ODARyT (ver Anexo A) está compuesta por 315 conceptos agrupados semánticamente en los grupos de Redes y Servicios de Telecomunicaciones, Sistemas Distribuidos, Sistemas de Comunicación Digital y Documento. En la figura 3.2 se observa un fragmento de la jerarquía de ODARyT y en la tabla 3.3 se muestra la cantidad de conceptos que hay en cada grupo semántico.

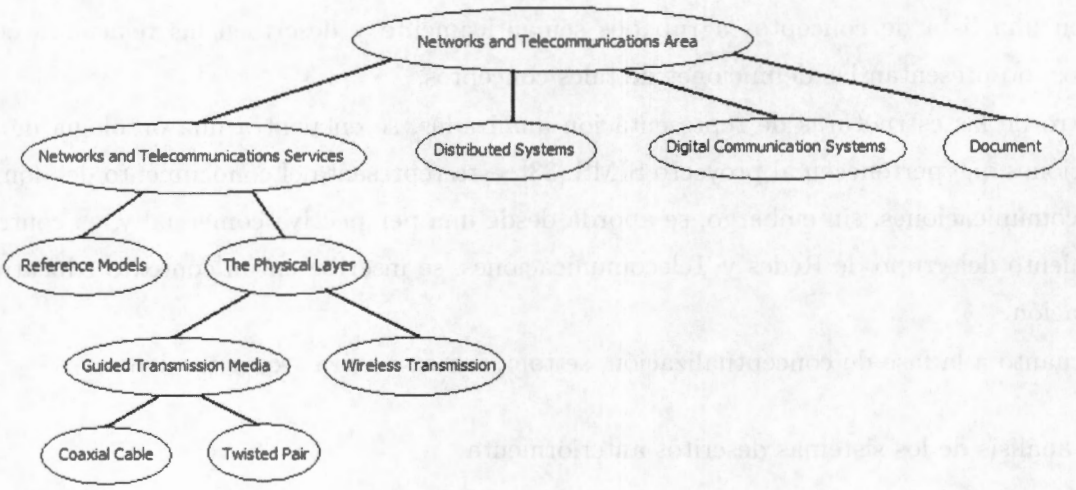


Figura 3.2: Fragmento de la jerarquía de ODARyT

Grupo semántico	Número de conceptos
Redes y servicios de telecomunicaciones	145
Sistemas Distribuidos	42
Sistemas de Comunicación Digital	81
Documento	47

Tabla 3.3: Grupos semánticos de ODARyT

ODARyT tiene una profundidad máxima de 6 niveles. En el grupo semántico de Redes y servicios de Telecomunicaciones, podemos encontrar como subgrupos principales: las capas de red, según el modelo OSI, la seguridad en redes, la evaluación de desempeño y los estándares. Entre los subgrupos principales que caracterizan al grupo semántico de Sistemas distribuidos están: los modelos de arquitectura, los objetos distribuidos, los sistemas operativos, los sistemas par a par, las transacciones distribuidas, la replicación, el estado global y la memoria compartida distribuida. En cuanto al grupo semántico de Sistemas de comunicación digital, los subgrupos principales son los elementos característicos del modelo de comunicación: la fuente de información, la codificación, la decodificación, el canal de comunicación, la modulación, la demodulación, los filtros, el ruido y los tipos de conectividad.

Las definiciones y conceptos de ODARyT están expresados en inglés, esto permite su reutilización en otros grupos de Redes y Telecomunicaciones. También es útil para la construcción de las anotaciones, esto por que la gran mayoria de los recursos documentales están escritos en inglés.

3.4. Anotaciones

La Web Semántica Corporativa se apoya en la inclusión de anotaciones, que son descripciones semánticas (por ejemplo, habilidades de las personas, contenido de los documentos, características de los servicios, etc.) de los recursos, para la representación de su contenido, a través de un vocabulario controlado (ontología).

Las anotaciones de los recursos de la memoria corporativa, están compuestas por los siguientes atributos:

- Título del documento
- Autor
- Descripción
- *Ontokeywords*
- Palabras clave
- Nombre del archivo
- Formato
- Año de publicación

Las *ontokeywords* (palabras clave de la ontología) corresponden a los conceptos representativos del documento, que pertenecen a ODARyT, por otro lado las *palabras clave* son otros conceptos representativos del documento que no están en ODARyT. Un ejemplo de una anotación en lenguaje RDF [37] para el documento que lleva por título *Distributed Site Partitioning* es la siguiente:

```
<Document rdf:ID="Distributed">
  <title>Distributed Site Partitioning</title>
  <authors>R. Marcelín Jiménez</authors>
  <description>Resolve a problem about combinatorial optimization</description>
  <ontokeywords>distributed storage system</ontokeywords>
  <otherkeywords>ants systems, combinatorial problems</otherkeywords>
  <filename>DSP_RMJ.pdf</filename>
  <format>propio</format>
  <year>2003</year>
</Document>
```

3.5. Interfaz para la edición de ontologías y anotaciones

En la Web semántica (WS) las ontologías y las anotaciones se representan a través de lenguajes específicos que proveen la sintaxis y el estilo formal requerido para su representación.

Web Ontology Language (OWL) [36] es un lenguaje de Ontologías Web. OWL es un lenguaje recomedado por el W3C (Consorcio World Wibe Web) desde febrero de 2004. Se usa para compartir conocimiento, de tal manera que las máquinas puedan entenderlo. OWL utiliza RDF para añadir las siguientes capacidades a las ontologías: capacidad de distribuirse a través de varios sistemas, es más general, escalable y compatible con los estándares Web de accesibilidad e internacionalización. Permite una interoperabilidad entre las ontologías.

Resource Description Framework (RDF) [37] es un lenguaje recomendado por el consorcio W3C que permite describir los recursos (documentos, imágenes, archivos de sonido, personas, etc.) en la Web. La representación de una descripción se define como una tripleta (S,P,O), donde S, es el sujeto, P es el predicado o la propiedad y O es el valor de la propiedad, esto es: el sujeto S tiene la propiedad P con valor iguala a O. RDF utiliza el lenguaje XML (Lenguaje de Marcado Extensible) para la codificación, y así se facilita la creación e intercambio de los metadatos.

La construcción y mantenimiento de una ontología en un lenguaje formal (por ejemplo, OWL) es complicada y consume bastante tiempo, sobre todo si el (o los) usuario(s) que deberán darle mantenimiento no estan familiarizados con esos lenguajes. Por otro lado, dado que las anotaciones a un recurso, deben estar basadas en los conceptos de una ontología, es útil tener una guía (plantilla) de los elementos que constituyen la anotación y poder elegir de entre todos los conceptos de la ontología, cuales describen mejor el recurso.

En la actualidad podemos encontrar gran cantidad de herramientas que permiten construir ontologías. Estas herramientas generalmente proveen una interfaz que ayudan a los usuarios al desarrollo y uso de ontologías. Goméz-Pérez [23] distingue los siguientes grupos de herramientas: 1) de desarrollo de ontologías, 2) de evaluación de ontologías, 3) de combinación y alineamiento de ontologías, 3) de anotaciones basasdas en ontologías y 4) de consulta de ontologías y motores de inferencia.

Dado que existe una gran diversidad en las herramientas, en el uso que tienen estas sobre las ontologías y las metodologías que emplean para su desarrollo, tomando en consideración que los usuarios finales no necesariamente están familiarizados con los elementos de la Web semántica. Se propone el desarrollo de una interfaz de usuario que permita cubrir las necesidades de edición amigable y consulta de las anotaciones. Los propios investigadores del grupo de RyT serán quienes editen las anotaciones de sus recursos, seleccionando los conceptos de ODARyT y puedan además, hacer consultas sobre las anotaciones realizadas.

Herramienta	Edición de Ontologias	Edición de Anotaciones	Navegación por la Ontología	Lenguaje	Soporte de Consultas	Lenguaje de Consultas
Jena	No	Si	No	RDF	Si	SPARQL
Sesame	Si	Si	No	RDF (S)	Si	RQL
Protégé	Si	No	Si	RDF OWL	-	-
Sewese	Si	Si	Si	RDF (S) OWL	Si	SPARQL

Tabla 3.4: IDE's para el desarrollo de aplicaciones de la Web semántica

Para la representación de una ontología de dominio y las anotaciones, se necesita de una interfaz con las siguientes características:

- Edición (altas, bajas, cambios) de los conceptos de la ontología.
- Navegación a través de la ontología, despliegue de definiciones.
- Edición de anotaciones.
- Soporte para un lenguaje de consulta.

Para encontrar una aplicación que nos permitiera cubrir los requerimientos de la interfaz, se analizaron algunas aplicaciones para la edición de ontologías y anotaciones, los resultados de este análisis se muestran en la tabla 3.4. Estas aplicaciones corresponden a entornos de desarrollo integrados (IDE), que incluyen una serie de elementos (*toolkits*) para el desarrollo de aplicaciones de la WS, excepto por Protégé, que es considerado sólo un editor de ontologías. A continuación se describen las aplicaciones comparadas.

Jena [38] es un API de Java creado por los laboratorios Hewlett Packard (HP) para los programadores de la WS, permite manipular y analizar sintácticamente documentos y gráficas RDF. Jena provee varias herramientas como un analizador sintáctico RDF/XML, un lenguaje de consultas, un módulo de E/S para N3, N-triples y RDF/XML y puede almacenar de forma persistente gráficos RDF en memoria o en bases de datos. Las principales metas de Jena son: a) permitir el fácil acceso y la manipulación a los datos en los gráficos y tripletas RDF, b) visualizar el gráfico RDF. Las tareas para la base de datos en donde se almacenan los gráficos RDF, son: agregar, almacenar, eliminar, remover y buscar enunciados RDF. La operación de buscar recupera todos los enunciados que empatan con un patrón (S,P,O).

Sesame [39] es una arquitectura para el almacenamiento y consulta de información RDF y RDFS. Es un sistema open source y esta implementado en Java, desarrollado por Aidministrator Nederland como parte del European IST project On-To-Knowledge. Sesame permite almacenamiento persistente de datos RDF e información de esquemas, provee métodos de acceso a esta información a través de módulos de exportación y consulta. Sesame adopta el lenguaje *RDF Query Language* (RQL) como lenguaje de consultas.

Protégé [40] es un editor de ontologías de código abierto que soporta el lenguaje OWL. Permite crear y guardar ontologías OWL, editar y visualizar conceptos, propiedades y reglas de inferencia. Protégé soporta consultas a través de módulos externos.

SeWeSe (*Semantic Web Server*) [41] es una plataforma que provee primitivas y componentes reutilizables, configurables y extensibles que reducen la cantidad de tiempo gastado en el desarrollo de nuevas aplicaciones de la WS, y así, permitir centrarse durante el desarrollo de estas aplicaciones en el dominio del problema. Está construido sobre el motor de búsqueda CORESE [42] y sobre el servidor Tomcat [43], y agrega una librería JSP (Java Server Pages) de etiquetas para tareas específicas. Provee etiquetas para desarrollar un editor de ontologías, un editor genérico de anotaciones y un editor básico de reglas. Además integra un conjunto de primitivas para construir interfaces para consultas, edición y navegación. En la tabla 3.4, se puede observar que Sewese es la aplicación más completa de acuerdo a nuestras necesidades.

Haciendo uso de la librería JSP, provista por Sewese, y sobre el servidor Tomcat, desarrollamos una interfaz de usuario *ad-doc* con las necesidades del proyecto, que integra los elementos necesarios para la edición de ontologías y anotaciones.

La interfaz desarrollada esta integrada por varias vistas. Para el desarrollo de esta interfaz, a cada vista le corresponde un archivo fuente que contiene etiquetas de la librería JSP que nos provee Sewese, estos archivos JSP son interpretados por el servidor web Tomcat y se genera la vista en un navegador Web.

A continuación se describen las principales vistas de la interfaz desarrollada:

- **Vista de la ontología**: Contiene un menú principal con las opciones: *New Ontology*, *Open Ontology*, *New Annotation* y *Open Annotation*. La opción *New Ontology*, crea una ontología nueva, y sólo requiere del nombre para la nueva ontología. La opción *Open Ontology*, despliega, en un formato de lista de enlaces, los conceptos de la ontología y permite navegar a través de estos, mostrando su definición, sus conceptos padres y sus conceptos hijos (ver figura 3.3). La opción *New Annotation*, crea un nuevo archivo de anotaciones. La opción *Open Annotation*, despliega la información del archivo de anotaciones seleccionado.

- **Vista de la edición de la ontología**: En la figura 3.4 se puede observar que esta pantalla

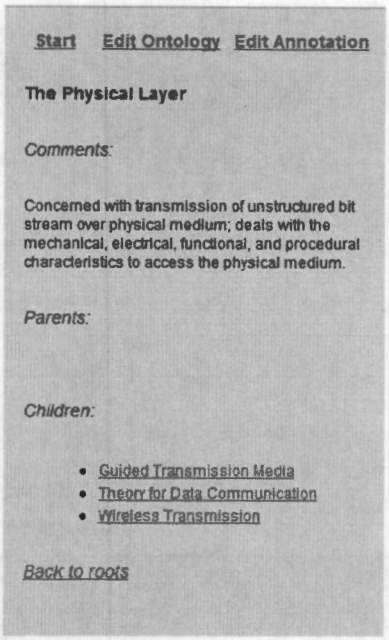


Figura 3.3: Interfaz: Definición del concepto *capa física*

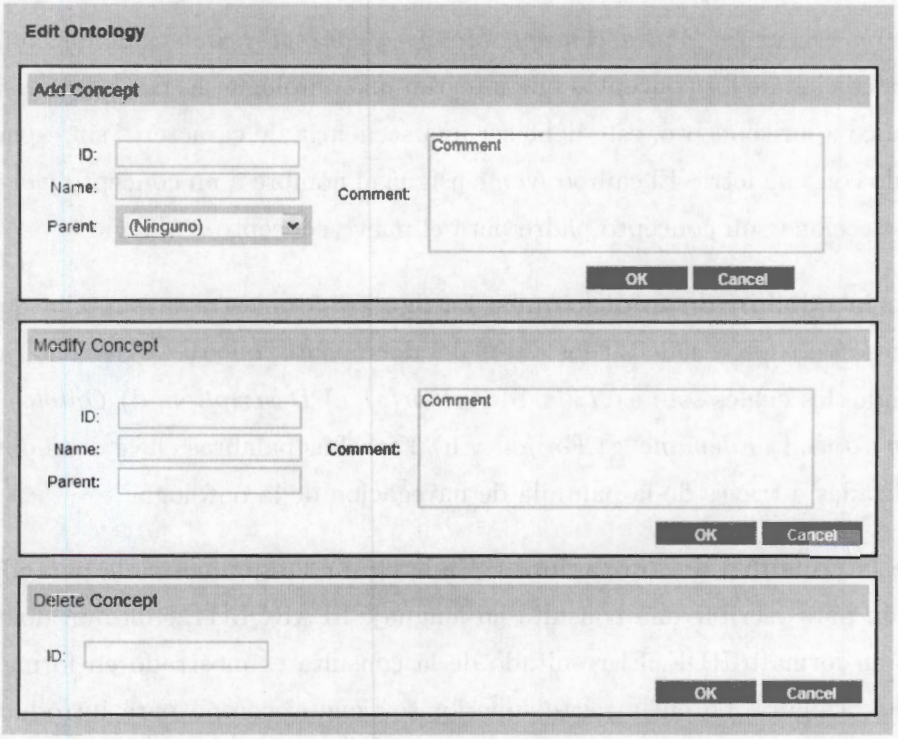


Figura 3.4: Interfaz: Edición de una ontología

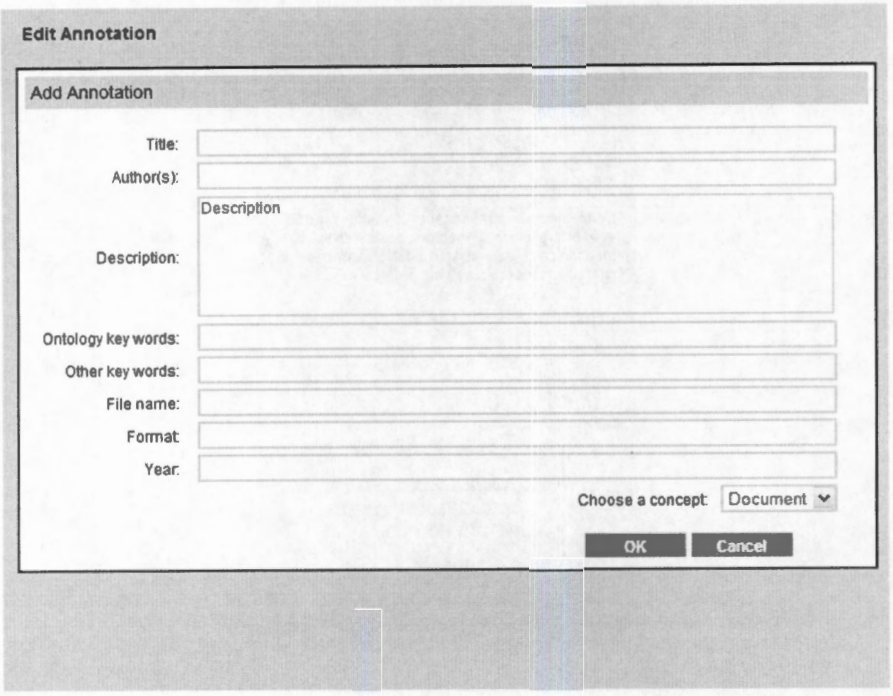
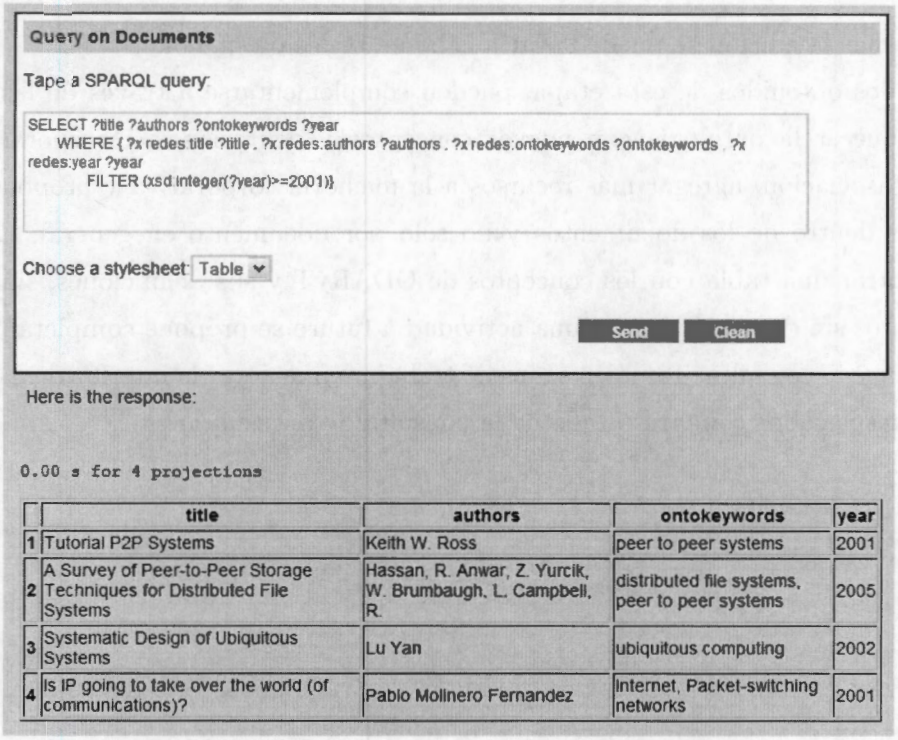


Figura 3.5: Interfaz: Edición de anotaciones

contiene tres secciones: *Add a concept*, *Modify a concept* y *Delete a concept*. Esta pantalla permite la edición de los conceptos que integran una ontología. El campo *ID* asigna un identificador único a un concepto, este debe ser una secuencia de caracteres sin espacios en blanco, empezando con una letra. El campo *Name* asigna el nombre a un concepto y el campo *Parent*, permite seleccionar un concepto padre para el nuevo concepto a agregar.

- **Vista de la edición de anotaciones**: La figura 3.5 muestra la pantalla que presenta los campos necesarios para solicitar los datos del documento, que describen sus características y su contenido, los cuales son: a) *Title*, b) *Author(s)*, c) *Description*, d) *Ontology key words*, e) *Other key words*, f) *Filename*, g) *Format* y h) *Year*. Las palabras clave de la ontología pueden ser visualizadas a través de la pantalla de navegación de la ontología.

- **Vista de la consulta de anotaciones**: En la figura 3.6 se muestra la pantalla que contiene una sección para escribir una consulta en lenguaje SPARQL[44] sobre un documento de anotaciones en formato RDF. El resultado de la consulta es mostrado en forma de tabla. Por ejemplo en la figura 3.6 en la parte superior se tiene el campo para introducir una consulta, esta consulta corresponde en lenguaje natural a: "Desplegar el título, autor, *ontokeywords* (palabras clave de la ontología) para los documentos anotados con año de publicación mayor o igual a 2001", el resultado de esta consulta se observa en forma de tabla en la parte inferior de



Query on Documents

Tape a SPARQL query:

```
SELECT ?title ?authors ?ontokeywords ?year
    WHERE { ?x redes:title ?title . ?x redes:authors ?authors . ?x redes:ontokeywords ?ontokeywords . ?x
redes:year ?year
        FILTER (xsd:integer(?year)>=2001)}
```

Choose a stylesheet: Table

Send Clean

Here is the response:

0.00 s for 4 projections

	title	authors	ontokeywords	year
1	Tutorial P2P Systems	Keith W. Ross	peer to peer systems	2001
2	A Survey of Peer-to-Peer Storage Techniques for Distributed File Systems	Hassan, R. Anwar, Z. Yurcik, W. Brumbaugh, L. Campbell, R.	distributed file systems, peer to peer systems	2005
3	Systematic Design of Ubiquitous Systems	Lu Yan	ubiquitous computing	2002
4	Is IP going to take over the world (of communications)?	Pablo Molinero Fernandez	Internet, Packet-switching networks	2001

Figura 3.6: Interfaz: Consulta de anotaciones

la pantalla, en esta tabla podemos observar que la primera columna corresponde al título del documento, la segunda a los autores, la tercera a las ontokeywords y por último se muestra el año de publicación. Para este ejemplo tenemos 4 documentos anotados con año de publicación mayor o igual a 2001.

3.6. Conclusiones y trabajo futuro

ODARyT es una ontología que modela la memoria corporativa del grupo de investigación de Redes y Telecomunicaciones, sin embargo, puede extenderse a una comunidad más amplia. Del mismo modo, ODARyT puede reutilizarse según se considere conveniente en otros proyectos o áreas de investigación afines.

La metodología empleada para la construcción de ODARyT puede generalizarse y exportarse a otros grupos de investigación que busquen la construcción de su memoria corporativa basada en ontologías.

La interfaz para la edición de ontologías y anotaciones se orienta a usuarios comunes que no tienen un conocimiento previo sobre la Web semántica, sus elementos y aplicaciones, es por ello que la interfaz presentada permite la edición de ontologías y anotaciones de forma sencilla y fácil para

el usuario. Además la edición se puede llevar a cabo en inglés o español.

Los resultados obtenidos de esta etapa, pueden complementarse a través de las siguientes actividades: enriquecer la ontología con nuevos conceptos y definiciones, incorporar relaciones de equivalencia y asociación, agregar más recursos a la memoria corporativa y proponer anotaciones por fragmentos dentro de los documentos y no sólo por documento en general. En el Anexo A podemos encontrar una tabla con los conceptos de ODARyT y sus definiciones, sin embargo, hay algunos conceptos sin definición, como una actividad a futuro se propone completar esta tabla con las defincciones que hacen falta. También se puede trabajar en mejorar la interfaz de usuario así como los elementos desplegados como resultado de la consulta de un usuario.

Capítulo 4

Parte II. Modelo semántico de almacenamiento y recuperación de información

4.1. Introducción

La memoria es un recurso fundamental de la computadora. Los sistemas de almacenamiento ofrecen esta capacidad a un costo inferior que la RAM y con una estabilidad de largo plazo. No obstante, también presentan ciertos inconvenientes, por ejemplo, anchos de banda que limitan las tasas de transferencia y latencias de acceso muy altas. Estos atributos costo, persistencia, ancho de banda y latencia son las métricas con las que tradicionalmente se evalúan los sistemas de almacenamiento. Sin embargo, el acentuado desarrollo de redes de telecomunicaciones observado en las últimas décadas, hace más complejo el escenario [45].

En la actualidad los usuarios de Internet consultan documentos cuyo despliegue requiere la recuperación de información desde docenas de sitios diferentes alrededor del mundo. Éste ha sido el primer sistema de almacenamiento distribuido de información, con una escala global, lo cual ilustra el impacto tecnológico, económico y cultural del enfoque distribuido. Sin embargo, su fragilidad semántica y operacional limitan sus aplicaciones.

Por otro lado, el desarrollo de los grandes sistemas de almacenamiento actuales está tomando en cuenta los mecanismos de funcionamiento de los sistemas P2P. Esto añade una gran cantidad de retos de investigación, sobre todo en cuatro grandes áreas de investigación: búsqueda, almacenamiento, seguridad y aplicaciones [47, 46, 49]. Es sobre las dos primeras categorías que la propuesta de esta tesis se enfoca, haciendo un fuerte énfasis en las relaciones semánticas que pueden presentarse en los objetos almacenados en un sistema P2P.

Por lo anterior, es evidente la necesidad de fusionar varias tecnologías para poder almacenar y

gestionar mejor los recursos de un grupo de trabajo o una comunidad científica, materializados en diferentes formatos y distribuidos en una red institucional.

4.2. Sistemas de almacenamiento par a par

A finales de la década de los 90's se comenzó a difundir una aplicación llamada Napster, la cual permitía que un gran número de usuarios almacenaran y compartieran archivos musicales, sobre todo en formato MP3, con un servicio básico de búsqueda centralizado. Las características de Napster, dieron lugar al nacimiento de sistemas que ahora conocemos como Sistemas par a par (P2P). El término *peer-to-peer* (o par-a-par en castellano) se refiere a una clase de sistemas que utilizan recursos distribuidos para realizar una función crítica de una manera descentralizada [4].

El advenimiento de los sistemas de almacenamiento par a par al inicio de la década actual, ha traído consigo una gran revolución en términos de nuevas aplicaciones distribuidas. Dado que el principio fundamental de Internet es un servicio del mejor esfuerzo (*best-effort*), los sistemas P2P se ejecutan en los extremos de Internet. Esto significa que todos los procedimientos de control y de comunicación en este tipo de sistemas, deben realizarse de extremo a extremo y no se modifican los mecanismos de funcionamiento de Internet. Con esto en mente, un sistema de este tipo puede lograr cantidades enormes de almacenamiento de recursos y de procesamiento, mientras que al mismo tiempo se minimizan los costos de dimensionamiento [4, 7], el cual se refiere a estimar el crecimiento potencial que puede tener el acervo documental y conocer las dimensiones del equipo de almacenamiento que se necesitará en un futuro.

Una tarea importante de los sistemas de almacenamiento P2P es la búsqueda de información. El problema de la búsqueda en un sistema P2P consiste en encontrar un recurso, sin servidores centralizados y considerando aspectos de escalabilidad. Los nodos son autónomos en la forma como almacenan la información de manera local. Cada nodo en la red P2P puede ser cliente o servidor y puede comunicarse directamente con otro nodo. Los nodos del sistema trabajan de una manera colaborativa para obtener resultados a la solicitud de búsqueda realizada por un usuario, mediante interacciones punto a punto sobre la red P2P.

Algunas de las características principales de un sistema de almacenamiento distribuido P2P [10] son:

- *Escalabilidad*: Es deseable que ante el incremento de los nodos en la red, el sistema siga funcionado correctamente.

- *Robustez*: Se refiere a seguir en funcionamiento aún con fallos en los nodos de la red.

- *Descentralización*: Quiere decir que cada uno de los nodos puede tomar el rol de cliente o de servidor y que la comunicación entre pares es simétrica.
- *Costos distribuidos*: Al compartirse los recursos, los costos también se reparten entre los nodos participantes en la red.
- *Anonimato*: Es deseable mantener oculta la información acerca del autor de un contenido, el lector, el servidor que lo alberga y la petición para encontrarlo.
- *Seguridad*: Identificar y evitar nodos con contenido malicioso.
- *Persistencia de información*: El sistema debe ser capaz de proveer acceso a los datos, y se debe asegurar que los datos almacenados en el sistema están disponibles y protegidos.
- *Balance de carga*: Se refiere a que el sistema debe ser capaz de hacer una distribución óptima de los recursos, basada en la capacidad y disponibilidad de los nodos.
- *Localización rápida de los recursos*: Es una de las características más importantes, pues como los recursos están distribuidos en diversos pares se necesita de un mecanismo eficiente para la localización y recuperación del recurso solicitado.

Por su topología las redes P2P pueden clasificarse en:

- *Estructuradas*: usan DHT (*Distributed Hash Tables*) (función o método para generar llaves que representan la localización de un documento) para la localización de los recursos y dirigir la búsqueda de los mismos, ejemplos de estos sistemas son: CAN, Chord, Pastry, Tapestry.
- *No estructuradas*: los recursos son localizados aleatoriamente. La búsqueda es a través de "inundación"de la red, y por lo tanto es general e ineficiente. Ejemplos: Gnutella, Kazaa.

4.3. Descripción general

Es bien sabido que el enrutamiento en Internet, basado en tablas, está sustentado en dos procedimientos complementarios: el primero encargado del mantenimiento y actualización de las tablas, el segundo encargado de consultarlas cuando se requiere alcanzar un destino en la red.

Como en el caso de las tablas de enrutamiento, se propone organizar el almacenamiento y recuperación de los documentos de una Web Semántica Corporativa (WSC) mediante dos procedimientos: el primero que resuelve el emplazamiento o localización de las tablas de rangos semánticos y, el segundo, que efectúa la búsqueda de contenidos consultando la información (índices) de las tablas.

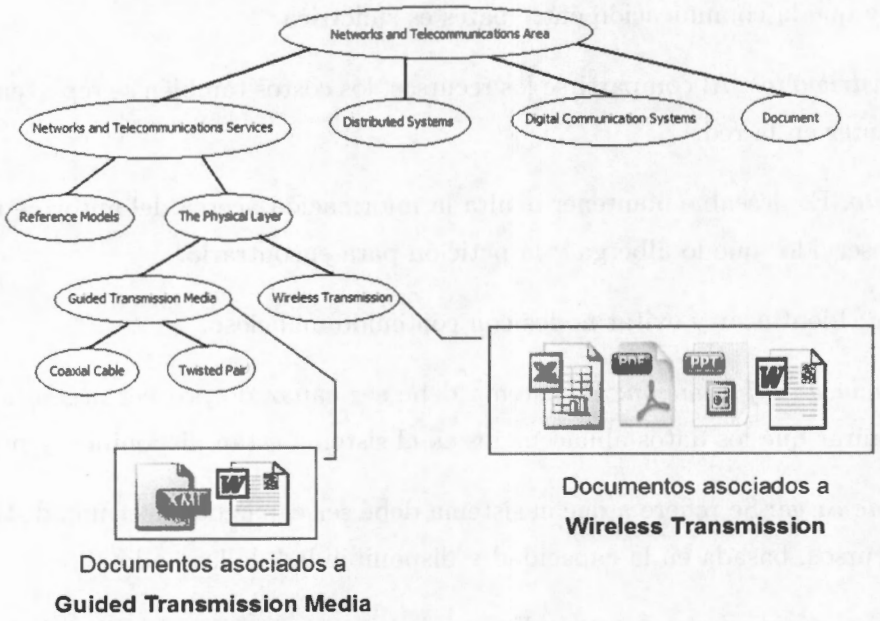


Figura 4.1: Ejemplo de documentos asociados a un concepto de ODARyT

Una WSC consta de elementos como ontologías y recursos documentales. En la primera parte de esta tesis, describimos la WSC perteneciente al grupo de Redes y Telecomunicaciones, así como la construcción de una ontología de dominio llamada ODARyT. En la figura 4.1 observamos un fragmento de ODARyT y un ejemplo de la asociación de documentos para los conceptos *Guided Transmission Media* y *Wireless Transmission*, con un número diferente de documentos asociados. En este sentido los conceptos de ODARyT tienen asociada una cantidad de documentos que poseen un tamaño en bytes (peso). Nuestra propuesta consiste en localizar y recuperar estos documentos, tomando en cuenta su contenido, dado que un documento puede ser asociado a un concepto que pertence a ODARyT.

Lo anterior implica el desarrollo de un algoritmo que distribuya los documentos de una WSC dentro de los nodos de la red P2P, para almacenar la memoria corporativa. Un concepto corresponde a un índice semántico y un rango semántico agrupa uno o varios índices semánticos. Una vez, que se ha encontrado la mejor solución para la distribución de los documentos dentro de la red P2P, los recursos se almacenan en cada nodo, de acuerdo al rango semántico al que pertenecen.

Dado que una ontología puede ser vista como un grafo (por la jerarquía de conceptos), y los nodos de la red P2P como otro grafo, entonces el problema a resolver es del tipo *Empotrado de Gráficas* (GE, por sus siglas en inglés) [50]. Los problemas GE, pertenecen a problemas de optimización combinatoria y se clasifican como problemas NP-Completos [51]. Un mecanismo de solución de los

problemas GE es a través de heurísticas [50].

4.4. *Colonia de Hormigas*

Una *colonia de hormigas* puede entenderse como un sistema distribuido que, basado en un conjunto simple de reglas de comportamiento mostradas por sus individuos, puede presentar una alta organización estructurada. Como resultado de esta organización las colonias de hormigas pueden resolver tareas complejas como la búsqueda de alimento, transporte cooperativo, división del trabajo, entre otros. Los algoritmos hormiga se derivan de la observación del comportamiento real de las hormigas y son usados como modelos para la solución de problemas de optimización discreta.

Debido a su escasa habilidad visual, en algunas especies nula, las hormigas tienen otro medio de comunicación con su entorno, esto es a través de la segregación de una sustancia química llamada *feromona.*

Por ejemplo se sabe que las colonias de hormigas resuelven el problema de búsqueda de alimento que puede modelarse como un problema de optimización combinatoria. Para ello, cada hormiga comunica a los demás el camino que ha seguido descargando en él la feromona. Las hormigas tienden a seguir estos rastros de feromonas. Al cabo de cierto tiempo, los caminos más cortos entre el *hormiguero* y las fuentes de comida se recorren con más frecuencia y van acumulando más y más feromona, lo que a su vez refuerza el rastro anterior [45].

En trabajos como [47] se propone el uso de un algoritmo hormiga, para resolver el problema de partición de un grafo, que también es un problema NP. El algoritmo usado es una adaptación del algortimo de busqueda en profundidad (*Depth-First-Search*) distribuido. Por otro lado el proyecto AntHill [52], propone un marco de referencia para diseñar aplicaciones P2P, basadas en el modelo biológico de la colonia de hormigas. AntHill, toma en cuenta características de bajo nivel, tales como comunicación, seguridad y calendarización para implementar nuevos protocolos P2P. En cuanto al almacenamiento de documentos el algoritmo hormiga, sigue la política de asignar espacio de almacenamiento a los documentos más recientemente usados y descartar documentos raramente usados.

4.5. Modelo del problema

Una ontología puede ser vista como una jerarquía de conceptos. Un concepto corresponde a un índice semántico y cada índice semántico puede tener asociada una colección de documentos pertenecientes a la memoria corporativa. Un rango semántico es un conjunto de índices semánticos agrupados en un mismo nodo, puede tener uno o varios índices semánticos. Es por esto que una

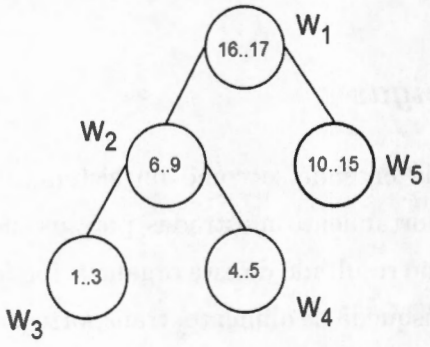


Figura 4.2: G_1, Web Semántica Corporativa

WSC puede ser modelada como un grafo G_1, donde cada nodo tiene 1) un rango semántico y 2) un peso ω dado por la colección de documentos que comparten ese rango semántico.

En la gráfica de la figura 4.2 se observa un ejemplo de una WSC. Nótese como los nodos $(i = 1, \ldots, 5)$ muestran los índices semánticos que abarcan, así como el peso de los documentos que les corresponden. No existe *a priori* una relación entre la longitud del rango semántico y el peso del nodo. Algunas veces pueden existir muchos documentos con los mismos índices, otras veces todos los documentos tienen índices distintos.

Por su parte, la *red de almacenamiento* se modela por una gráfica G_2. Cada nodo (almacén) tiene asociado una capacidad c que caracteriza la máxima cantidad de información que puede guardar. La gráfica de la figura 4.3 muestra un caso de red de almacenamiento. Cada almacén j, $(j = s_1, \ldots, s_3)$ tiene asociada una capacidad. En este caso particular decimos que todos los almacenes son de capacidad homogénea c, donde $c_1 = c_2 = c_3$.

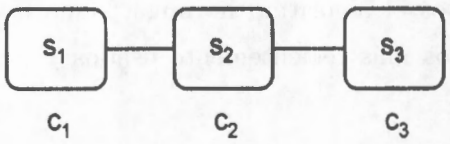


Figura 4.3: G_2, Red de almacenamiento P2P

La localización o *emplazamiento* implica la solución del problema de empotrado de G_1 dentro de G_2. Este consiste en asociar o guardar el mayor número de nodos de G_1 dentro de los nodos de G_2 de modo que el peso acumulado de los primeros no exceda la capacidad del almacén al que quedan adscritos y, además, se use el menor número de almacenes. Cuando se concreta el empotrado, cada

almacén guarda también los índices semánticos asociados con cada emplazamiento ocupado, i.e. se guarda una tabla que indica donde se han almacenado los documentos ordenados por su rango semántico. La figura 4.4 muestra como G_1 (figura 4.2) ha sido empotrada en G_2 (figura 4.3), con ello podemos construir la tabla de búsqueda 4.1, en donde se indican los rangos semánticos contenidos en cada almacén.

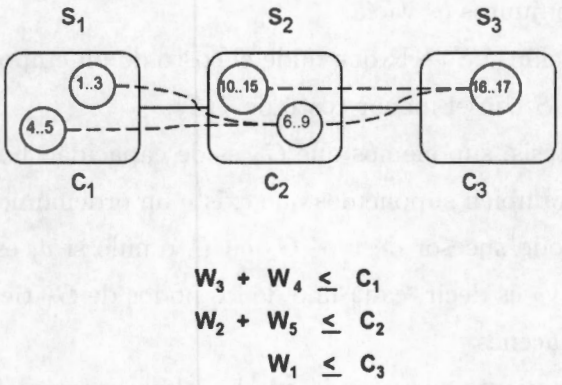


Figura 4.4: G_1 empotrada en G_2

Rango semántico	Almacén
1...5	s_1
6...15	s_2
16...17	s_3

Tabla 4.1: Tabla de búsqueda

Sea G_1=(V_1, E_1) un grafo que representa una Web Semántica Corporativa (WSC) y G_2=(V_2, E_2) un grafo que representa una red de almacenamiento. Un empotrado de G_1 dentro de G_2 es un par de mapas $S : (\nu, \epsilon)$, donde ν mapea cada uno de los nodos de G_1 a un almacen en G_2, mientras ϵ mapea aristas de G_1 a trayectorias en G_2, bajo la siguiente restricción:

Tenemos una función $w : V_1 \mapsto \mathbb{R}$, llamada peso. También, tenemos una función $c : V_2 \mapsto \mathbb{R}$, llamada capacidad.

Sea $N_j = [v_i \mid v_i \in V_1 \;\; \nu(v_i) = v_j]$ el conjunto de nodos V_1 mapeados a $v_j \in V_2$, donde:

$$\sum_{v_i \in N_j} w(v_i) \leq c(v_j), \forall v_j \in V_2$$

$$\bigcup_j N_j = V_1$$

$$N_i \cap N_j = \emptyset$$

El peso total de los nodos de V_1 almacenados en v_j ($v_j \in V_2$), no debe exceder su capacidad correspondiente. La unión de todos los conjuntos N_j forma el conjunto V_1 y la intersección de cualesquiera dos de estos conjuntos es vacía.

Finalmente hay una función $f{:}S \mapsto \mathbb{R}$ que mide el costo de un empotrado. El problema consiste en encontrar un empotrado S con el menor costo de $f(S)$.

Para las metas de esta tesis, suponemos que G_2 es de capacidad homogénea, i.e. cada almacén tiene la misma capacidad. También suponemos que existe un ordenamiento lineal L de los nodos de G_2, tal que succ(v_j) es el nodo sucesor de $v_j \in G_2$ en L, o nulo si v_j es el último. Otra suposición es que $w(v_i) \leq c(v_j)$, $\forall v_j \in V_2$ es decir, cada uno de los nodos de G_1 tiene un peso w menor o igual que la capacidad de los almacenes.

Para la red de almacenamiento suponemos que los almacenes son *entidades* estáticas. Esto se puede sustentar a través de un diseño modular o por capas, en el que exista una capa encargada de ocultar la presencia o ausencia de nodos, mediante técnicas de replicación de contenidos. De esta forma, nuestra propuesta puede integrarse en una capa superior que percibe almacenes lógicos, gestionados por una capa inferior. El nivel superior trabaja con almacenes lógicos provistos por un nivel inferior que administra nodos (*peers*) de almacenamiento de naturaleza dinámica que ofrecen servicios sin garantías.

Según Gutjhar [53], un *sistema de hormigas* se compone de un conjunto de $A_1, A_2, \ldots, A_Z$ de agentes. El *ciclo* del sistema es el período de tiempo durante el cual cada agente completa un camino sobre la gráfica G_1.

Sea $u = (u_0, u_1, \ldots, u_{t-1} = k)$ el camino que llega hasta un nodo k, recorrido por una hormiga en el ciclo m. Para un nodo l de V_1, se denota como $l \in u$ si el nodo está contenido en el camino y como $l \notin u$, en otro caso.

Cada hormiga elige el próximo nodo a visitar realizando un cálculo de probabilidad. La probabilidad de que la hormiga vaya del nodo k al nodo l, en el ciclo m, se denomina *probabilidad de transición* y se determina de la siguiente forma:

$$p_{kl}(m,u) = \frac{[\tau_{kl}(m)^\alpha][\eta_{kl}(m)^\beta]}{\sum_{r \notin u,(k,r) \in E_1} [\tau_{kr}(m)^\alpha][\eta_{kr}(m)^\beta]}$$

si $l \notin u$ y $(k,l) \in E_1$, o bien

$$p_{kl}(m,u) = 0$$

en otro caso.

Donde:

- τ_{kl} es la cantidad de feromona sobre la arista k, l
- α es un parámetro de control de la influencia de τ_{kl}
- η_{kl} es un parámetro denominado "deseabilidad"
- β es un parámetro de control de la influencia de η_{kl}

En nuestro caso, η_{kl} y β toman el valor de 1, y α de 0.2. Los números $\tau_{kl}(m)$ especifican la intensidad del valor de la feromona en la arista (k, l), para el ciclo m, y se actualizan según la fórmula:

$$\tau_{k,l}(m+1) = (1-\rho)\tau_{k,l}(m) + \rho\Delta\tau_{k,l}(m+1)$$

donde ρ se conoce como *factor de evaporación* de la feromona, así $\rho \approx 0$ significa que no hay evaporación; mientras que $\Delta\tau_{k,l}(m+1)$ es:

$$\Delta\tau_{k,l}(m+1) = \sum_{z=1}^{Z} \Delta\tau_{k,l}^{z}(m)$$

Con $\Delta\tau_{k,l}^{z}(m)$ que representa la cantidad de feromona depositada en la arista (k, l) por la *z-ésima* hormiga (A_Z), en el ciclo m, lo cual es $\phi(f_z)$, si A_Z atravesó (k, l) y 0 en otro caso.

Finalmente ϕ es una función no creciente que depende de los caminos completados durante los ciclos $1, \ldots, m-1$, y mide cuánto mejora la solución más reciente, comparada con resultados previos.

Gutjhar en [53] también muestra que el sistema de hormigas converge a una solución óptima después de un número finito de ciclos, debido a que $\Delta\tau_{k,l}^{z}(m)$ va incrementándose con valores positivos, hasta que la probabilidad de que cualquier hormiga z eligа la arista (k, l) que pertenece a un paso en el camino de la solución óptima, sea muy cercana a 1, en un ciclo m dado.

4.6. Algoritmo *Hormiga*

El algoritmo HORMIGA desarrollado en este trabajo de tesis es una adaptación del algoritmo distribuido para búsqueda en profundidad DFS (*Depth First Search*) distribuido.

Se propone lanzar z hormigas, cada una de las cuales realiza un recorrido aleatorio en profundidad (*random depth first search*) sobre la gráfica G_1 que representa a la WSC. A medida que visita los nodos de la gráfica, la hormiga suma el peso de los nodos por los que pasa al tiempo que los asigna a un almacén $v_j \in G_2$. Cuando el peso acumulado de los nodos que lleva guardados excede

la capacidad de éste almacén, entonces reasigna al último nodo de G_1 con succ(v_j) y reinicia este procedimiento de llenado en tanto continua su recorrido.

Cada hormiga visita exhaustivamente los nodos de G_1 y reporta su *trayectoria-solución* al *hormiguero* o agente concentrador. Por su parte, el *hormiguero* se encarga de analizar el costo de cada una de las z soluciones que recibe y luego reparte *feromona* sobre cada trayectoria, en función de su calidad. Enseguida, arranca una nueva ronda en la que vuelve a lanzar z hormigas cuyos recorridos, aunque aleatorios, estarán orientados por el último rastro de *feromona*. Este procedimiento se repite durante un número determinado de ciclos o hasta que cierto porcentaje del número inicial de hormigas siguen el mismo camino.

El costo computacional requerido para resolver el problema antes mencionado, está medido en términos de espacio y tiempo. La *complejidad en el espacio* esta dada por el número de hormigas utilizadas en el algoritmo para resolver el problema. La *complejidad en el tiempo* es el máximo número de ciclos que pueden transcurrir durante la ejecución del algoritmo hasta llegar a que cierto porcentaje de las hormigas sigan el mismo camino.

Los pasos que sigue el sistema de hormigas son los siguientes:

- Se crean y se lanzan z hormigas(agentes), desde un nodo fijo de la gráfica, denominado *hormiguero*. Cada hormiga realizará un recorrido aleatorio de búsqueda en profundidad sobre la gráfica G_1, visitando cada uno de los nodos de G_1.
- A medida que una hormiga visita los nodos de la gráfica, la hormiga suma el peso de los nodos por los que pasa, al tiempo que los asigna a un almacén en G_2.
- Cuando el peso acumulado en un almacén rebasa la capacidad de este, entonces reasigna el último nodo de G_1 al siguiente almacén.
- La hormiga reinicia este procedimiento de llenado en tanto continua su recorrido.
- Una vez que todas las hormigas regresan al sitio desde donde fueron lanzadas, se evalua su recorrido. La mejor hormiga es aquella que logró asignar los nodos de G_1 en el menor número de almacenes.
- Se asignan nuevas probabilidades en los arcos del grafo G_1, a los recorridos premiados.
- Se repite este proceso hasta que más del 75 % de las hormigas lanzadas inicialmente siguen el mismo recorrido.

A continuación se muestran los mensajes que se intercambian las hormigas(agentes) durante su ejecución.

DESCUBRE: Un nodo envía este mensaje a un vecino al que aún no se ha explorado, para indicarle que por aqui seguirá su recorrido

AVISO: Un nodo envía este mensaje a sus vecinos inmediatos, para indicarles que ha iniciado su ejecución.

NEWLISTA: Un nodo envía este mensaje al nuevo nodo que será incluido en la lista.

La creación de las particiones se basa en la capacidad fija del almacén. En el pseudocódigo del Algoritmo 1 se presenta el procedimiento para la asignación de un nodo a un almacén, este procedimiento lo realiza cada hormiga que visita un nodo de la gráfica G_1.

Algoritmo 1 Procedimiento para asignar un nodo a un almacén

Entrada: mensaje $NEWLISTA$, *listadenodos*, *tamParcial*

Salida: mensaje $ENVIA$ o mensaje $NEWLISTA$

1: **si** $NEWLISTA$ **entonces**
2: $p \leftarrow getPeso(miPid)$
3: agregar $miPid$ a *listadenodos*
4: $tamParcial \leftarrow tamParcial + p$
5: $NEWLISTA \leftarrow$ **falso**
6: envia(NEWLISTA, lista, tamParcial)
7: **si no**
8: $p \leftarrow getPeso(miPid)$
9: **si** $(tamParcial \leftarrow tamParcial + p) > CAPACIDADALMACEN$ **entonces**
10: agregar *lista* a *almacenes*
11: $NEWLISTA \leftarrow$ **cierto**
12: $tamParcial \leftarrow 0$
13: envia(NEWLISTA, lista, tamParcial)
14: **si no**
15: agregar $miPid$ a *listadenodos*
16: $tamParcial \leftarrow tamParcial + p$
17: envia(NEWLISTA, lista, tamParcial)
18: **fin si**
19: **fin si**

En este proyecto hemos usado un simulador de eventos discretos [48], desarrollado para estudiar algoritmos distribuidos. La simulación de eventos discretos (DES), es la construcción de modelos que cambian su estado en puntos discretos del tiempo. Usando esta herramienta se pueden ejecutar

uno o varios algoritmos en forma simultanea, lo cual es ideal para la implementación de nuestro *algoritmo hormiga* distribuido.

El simulador DES, es una plataforma de software que ofrece un ambiente para la implantación, simulación y análisis de algoritmos distribuidos. Con esta herramienta, el programador desarrolla un algoritmo codificado en C++ (ver Anexo B) y lo compila usando las librerías del simulador, y así, generar el archivo ejecutable del algoritmo. Nosotros desarrollamos nuestro algoritmo HORMIGA usando este simulador DES.

4.7. Experimentos

Una vez que tuvimos nuestro programa ejecutable, diseñamos una agenda de experimentos donde consideramos los siguientes parámetros:

- El número inicial de hormigas (agentes)
- El tamaño y la granularidad de G_1 con la capacidad de los nodos de G_2
- El factor de evaporación (premiación)
- El número de ciclos (iteraciones del algoritmo)

Nuestro problema se considera resuelto, cuando más del 75 % de las z hormigas siguen el mismo recorrido, usando la menor cantidad de almacenes para guardar los nodos de G_1. Ejecutamos una serie de simulaciones con nuestro programa para tener una perspectiva de qué valores son los adecuados para cada uno de los parámetros antes mencionados, y obtuvimos los resultados descritos a continuación.

Número inicial de hormigas

Primero tratamos de determinar el número z de hormigas adecuado para conseguir mayor variabilidad en las soluciones de la primera iteración (ciclo) del algoritmo. Esto con el fin, de que en la primera iteración se exploren la mayor cantidad posible de caminos aleatorios sobre G_1 y así acotar el número de recursos, como memoria, utilizados para alcanzar una solución óptima.

Ejecutamos 15 simulaciones cada una con 5 semillas (generador de números pseudoaleatorios), para un número variable de hormigas. Al finalizar la primera iteración del algoritmo medimos la variabilidad en las soluciones, una solución esta dada por el número de almacenes que se necesitan para empotrar G_1 (WSC) en G_2 (red de almacenamiento). La tabla 4.2 muestra los valores obtenidos de media, desviación estándar y varianza para las soluciones de la primera iteración del algoritmo. En estos experimentos el número de nodos (tamaño) de G_1 se fijo en 100, 200, 300 y 600 y para el número inicial de hormigas se tomaron los valores de 5, 10, 15, 20 y 25. La capacidad de los

Hormigas	5	10	15	20	25
Nodos de $G_1 = 100$					
Media	68.4	66.7	67.46	67.35	67.12
Desviación estándar	2.07	2.45	1.51	1.18	1.83
Varianza	4.28	6.01	2.27	1.39	3.36
Nodos de $G_1 = 200$					
Media	131.60	129.20	130.00	130.25	130.68
Desviación estándar	1.82	2.66	2.72	1.74	1.93
Varianza	3.30	7.06	7.43	3.04	3.73
Nodos de $G_1 = 300$					
Media	194.40	191.10	193.67	192.20	191.80
Desviación estándar	2.51	2.99	3.15	2.28	2.23
Varianza	6.30	8.98	9.95	5.22	4.99
Nodos de $G_1 = 600$					
Media	332.30	335.40	332.80	336.70	330.20
Desviación estándar	2.36	2.72	2.92	3.10	3.20
Varianza	5.60	7.41	8.56	9.63	10.25

Tabla 4.2: Tabla de varianza para las soluciones encontradas en la primera iteración

almacenes se fijo en 500 unidades de almacenamiento y para los pesos de los nodos de G_1 una distribución uniforme entre 0 y 500.

En las gráficas mostradas en las figuras 4.5, 4.6, 4.8 y 4.7, podemos ver que existe un número óptimo de hormigas iniciales, que producen mayor varianza en las soluciones. Por ejemplo, para un grafo $G1$ de tamaño 100, encontramos mayor variabilidad en las soluciones cuando el número de hormigas es igual a 10 (ver figura 4.5). Con los resultados obtenidos podemos decir que el número de hormigas depende del tamaño del grafo G_1, que es aproximadamente $\sqrt{n}$, donde n es el número de nodos de G_1.

Tamaño y granularidad de G_1 con capacidad de los nodos de G_2

Después realizamos experimentos, para determinar el número de almacenes requerido para empotrar G_1 en G_2, tomando en cuenta el tamaño de G_1, la distribución de pesos de los nodos de G_1 y la capacidad del almacén.

En la tabla 4.3 se muestran los promedios de la mejor solución, es decir cuántos almacenes se necesitan para guardar G_1 en G_2, considerando una distribución uniforme para los nodos de G_1,

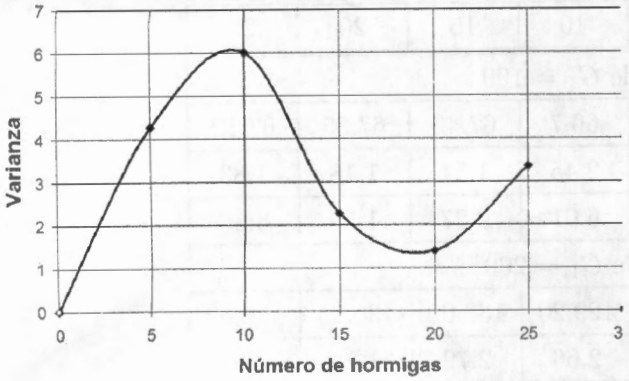


Figura 4.5: Varianza con G_1 de tamaño 100

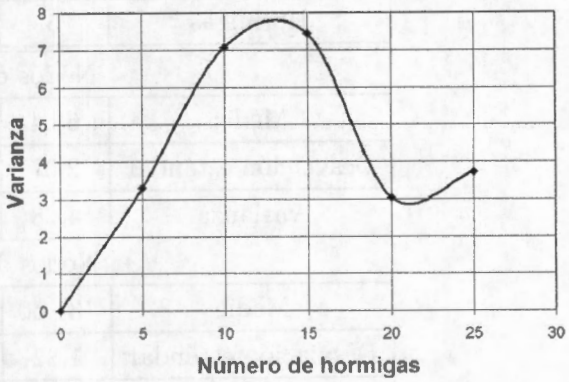


Figura 4.6: Varianza con G_1 de tamaño 200

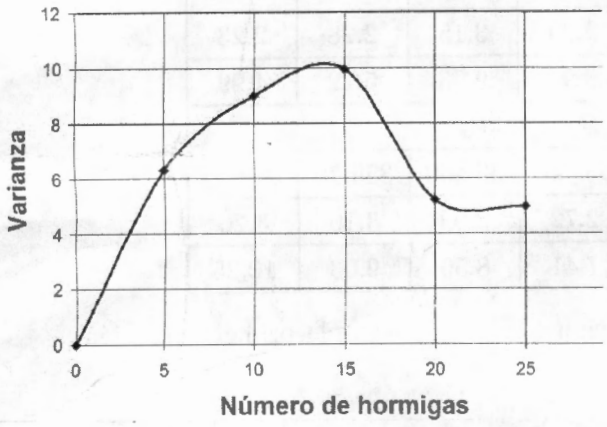


Figura 4.7: Varianza con G_1 de tamaño 300

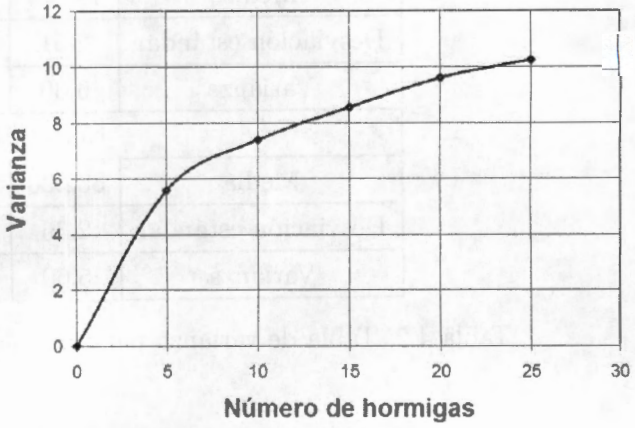


Figura 4.8: Varianza con G_1 de tamaño 600

entre distintos rangos de valores. Se considerá una solución cuando el 75 % o más del número inicial de hormigas reportan la misma trayectoria y el mismo número de almacenes. También podemos observar que no se tienen soluciones cuando los pesos son mayores a la capacidad del almacén, tal es el caso de capacidad del almacén igual a 100 y distribución de pesos de G_1 entre 0-180, 0-450 y 0-900.

Ejecutamos simulaciones en donde el tamaño de G_1 es de 100 nodos, los pesos de los nodos de G_1 tienen una distribución aleatoria uniforme entre 0-20, 0-50, 0-100, y la capacidad de los almacenes de G_2 varia en 100, 300 y 900. En la figura 4.9 podemos observar que cuando el limite superior del rango de la distribución de pesos es igual o cercano a la capacidad del almacén se necesitan más almacenes para guardar los nodos de G_1. Los resultados obtenidos muestran que existe una relación entre la cantidad de almacenes, la suma total de los pesos w_i y la capacidad del almacén. Podemos decir que el número de almacenes obtenido en los experimentos se acerca al resultado de

la expresión:

$$almacenes = \frac{\sum_{i=1}^{n} w(v_i)}{c}, \forall v_i \in V_1$$

Nodos de $G_1 = 100$			Nodos de $G_1 = 300$			Nodos de $G_1 = 900$		
c	Pesos	Almacenes	c	Pesos	Almacenes	c	Pesos	Almacenes
100	0-20	11.3	100	0-60	97.3	100	0-180	-
	0-50	28.2		0-150	-		0-450	-
	0-100	55.6		0-300	-		0-900	-
300	0-20	4.1	300	0-60	33.8	300	0-180	283.7
	0-50	9.8		0-150	81.7		0-450	-
	0-100	17.6		0-300	167.4		0-900	-
900	0-20	2.2	900	0-60	12.8	900	0-180	93.7
	0-50	3.2		0-150	26.4		0-450	245.2
	0-100	6.5		0-300	54.2		0-900	468.3

Tabla 4.3: Relación entre el tamaño de G_1 y la capacidad de los almacenes de G_2

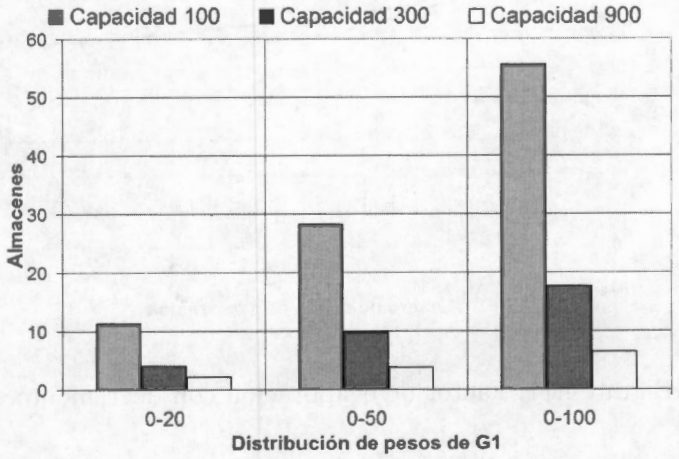


Figura 4.9: Número de almacenes para G_1 de tamaño 100

Factor de evaporación

En un tercer grupo de experimentos, estudiamos la influencia del factor de evaporación. Consideramos los siguientes parámetros: 1) tamaño de $G_1 = 100$, 2) distribución aleatoria uniforme de los pesos de los nodos de G_1, entre 0-20 y 3) la capacidad del almacén de 100.

Ejecutamos 15 simulaciones con 5 semillas diferentes, variando el factor de evaporación con valores de 0.1, 0.5 y 0.9. Graficamos entonces el mínimo número de almacenes, encontrado por cada ciclo dado por el sistema de hormigas.

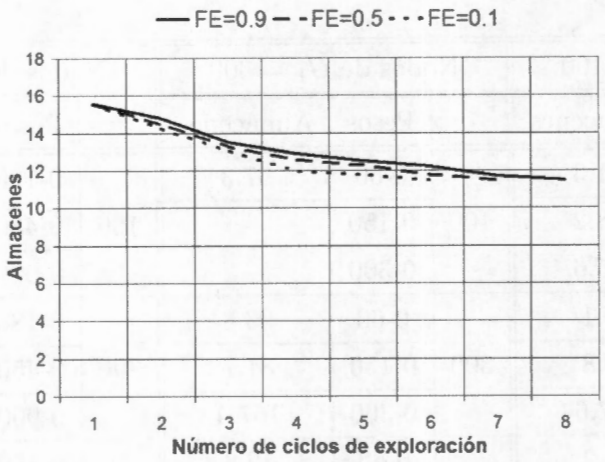


Figura 4.10: Factor de evaporación

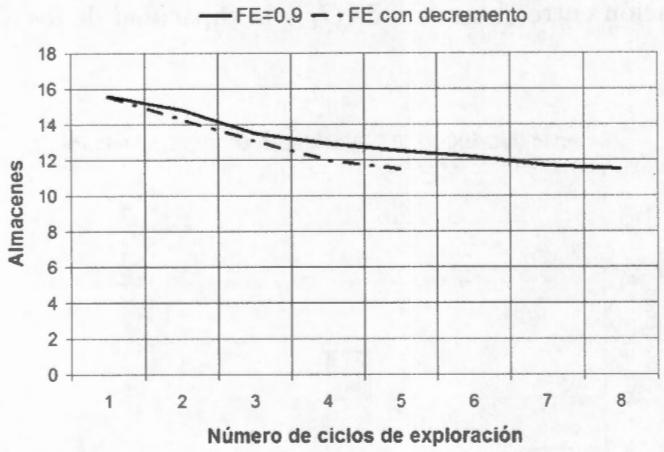


Figura 4.11: Factor de evaporación con decremento

En la gráfica de la figura 4.10 se puede observar que cuando es menor el factor de evaporación es menor el número de ciclos que se necesitan para encontrar la mejor solución. En particular en este problema queremos favorecer siempre la solución con menor número de almacenes encontrada.

Proponemos iniciar con un factor de evaporación alto en los primeros ciclos, pues en principio queremos que las hormigas no tengan *memoria* y exploren caminos diferentes. Para los ciclos posteriores proponemos decrementar el factor de evaporación para que las hormigas vayan reforzando el mejor camino encontrado en un ciclo dado.

Para un valor fijo inicial de 0.9 del factor de evaporación y un decremento de 0.1 por ciclo de ejecución, ejecutamos simulaciones con 1) tamaño de $G_1 = 100$, 2) distribución aleatoria uniforme de los pesos de los nodos de G_1, entre 0-20 y 3) la capacidad del almacén igual a 100. En la figura 4.11 se puede observar que se requieren menos ciclos de ejecución para alcanzar la misma solución.

Número de ciclos (iteraciones del algoritmo)

Número de hormigas	Pesos	Nodos de G_1	Capacidad del almacén	Número de ciclos
10	0 - 20	100	100	5.1
			300	3.5
			900	2.9
17	0 - 60	300	100	9.5
			300	7.8
			900	5.1
30	0 - 180	900	100	-
			300	14.6
			900	8.2

Tabla 4.4: Número de ciclos necesarios para alcanzar la solución

Una vez que establecimos el número de hormigas inicial, la capacidad de los almacenes en relación con los pesos de la WSC, y un factor de evaporación que cambia por ciclo de ejecución, en estas últimas simulaciones, queremos encontrar el número de ciclos (iteraciones) máximo que nos permite alcanzar la solución a nuestro problema.

En la figura 4.12, para G_1 de tamaño 300 con distribución de pesos entre 0-60 y una capacidad para el almacén de 100, podemos observar que el porcentaje de hormigas que reportan el menor número de almacenes encontrado en un ciclo, se va incrementando a lo largo de los ciclos del algoritmo.

Lanzamos 10 simulaciones cada una con parámetros fijos para el número de hormigas y distribución uniforme de los pesos para los nodos de G_1. Los parámetros y los resultados se encuentran en la tabla 4.4. El número de ciclos corresponde al promedio de los ciclos a partir del cual más del 75 % de las hormigas siguen la misma solución. Los resultados obtenidos muestran que existe una relación entre el número de nodos de G_1 y la capacidad de almacén, cuando G_1 es pequeña y el almacén es grande se necesitan menos ciclos para obtener la mejor solución, por el contrario cuando G_1 es grande y el almacen es pequeño se necesitan más ciclos.

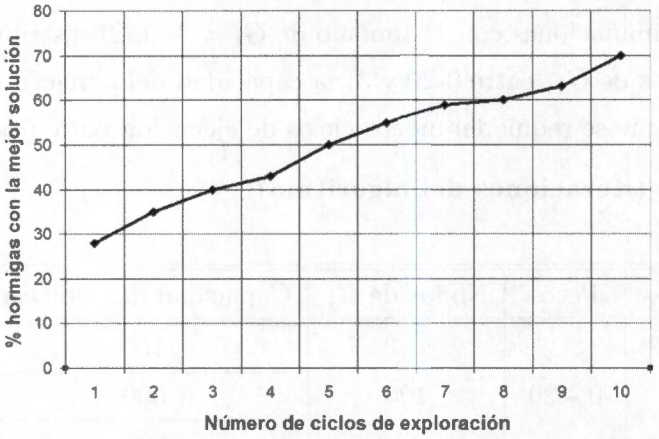


Figura 4.12: Porcentaje de hormigas que siguen la mejor solución

4.8. Recuperación semántica

Para la recuperación semántica, la solución encontrada por el algoritmo de almacenamiento, es replicada en cada uno de los nodos que pertenecen a la red, esta solución es representada a través de una estructura de datos con la dupla: (rango semántico, nodo), donde el nodo se identifica por el rango de llaves que tiene asignado.

Se tienen dos escenarios de recuperación de los recursos:

- La consulta del usuario en un nodo del sistema pertenece al mismo rango semántico que tiene a cargo ese nodo.
- La consulta del usuario en un nodo del sistema pertenece a un rango semántico diferente del que tiene a cargo ese nodo.

En el primer escenario se busca el recurso localmente y se despliega al usuario el(los) recurso(s) correspondiente a la consulta. En el segundo caso, como el propio nodo que recibe la consulta conoce, cual es el nodo que almacena el rango semántico al que pertenece la consulta del usuario, reenvía la petición al nodo del rango semántico correspondiente. Este nodo que recibe la consulta busca el recurso localmente y regresa el(los) recurso(s) asociado a la consulta. Mediante este proceso se garantiza la localización rápida de los recursos, asimismo la descentralización, pues cada nodo, asume el rol de cliente o de servidor según la consulta del usuario.

4.9. Conclusiones y trabajo futuro

La propuesta presentada para la localización de los documentos que constituyen la memoria corporativa es distribuida y descentralizada. La localización consiste en empotrar G_1 que representa la Web Semántica Corporativa (WSC) dentro de G_2, la red de almacenamiento; como resultado de este proceso obtenemos una tabla de búsqueda que es enviada a cada uno de los peers de la red. Esta tabla localiza los documentos de la WSC usando sus indices semánticos.

El algoritmo de almacenamiento propuesto encuentra: 1) cuál es la mínima cantidad de almacenes necesaria, para alojar una WSC con un tamaño y un peso asociado para cada uno de sus rangos semánticos y 2) la distribución de los índices semánticos y sus documentos dentro de los nodos de la red de almacenamiento, lo cual construye la tabla de búsqueda. La complejidad del algoritmo puede medirse en espacio y tiempo, en nuestro caso depende del tamaño de G_1. La complejidad en espacio, es decir, el costo en memoria del algoritmo, corresponde al número de hormigas, el cual es del orden de $O(\sqrt(n))$ y la complejidad en tiempo, es determinada a partir del número de ciclos que se ejecutará el algoritmo el cual depende del número de nodos de G_1 y de la capacidad del almacén. Por otro lado, la cantidad de almacenes necesaria para empotrar G_1 en G_2, depende del tamaño de G_1, de los pesos que tienen cada uno de los nodos de G_1 y de la capacidad de los almacenes de G_2.

El modelo para el proceso de recuperación de información es más rápido y tiene una complejidad constante, esto debido a que cada nodo conoce los índices que le corresponden a cada uno dentro de la red, la respuesta a la consulta de un usuario puede estar dada por los siguientes dos casos: 1) el nodo que recibe la consulta tiene la respuesta y 2) el nodo que recibe la consulta conoce que nodo tiene la respuesta y redirecciona hacia este la consulta. Lo anterior, evita que los nodos que no tienen la respuesta reciban peticiones de otros nodos. En contraste, en esta propuesta no se ha considerado el hecho de que un nodo pueda ser saturado con peticiones si las consultas de los usuarios, corresponden al rango semántico que guarda un sólo nodo.

En esta segunda fase, hay elementos a considerar para un trabajo futuro, tal es el caso, de que la capacidad de los almacenes no sea homogénea. También considerar que el peso de los nodos de la WSC aumente dinámicamente, es decir, que la cantidad de documentos asociados a un rango semántico aumenten, lo anterior implicaría tomar en cuenta un rango de peso extra al peso inicial, para que se sigan guardando los nuevos documentos en su almacén correspondiente. Además se debe considerar qué pasaría si un documento esta asociado a más de un rango semántico.

Capítulo 5

Conclusiones generales y perspectivas

Tomando en consideración que dentro de una organización existe heterogeneidad y multiplicidad en sus recursos, la propuesta de integración de elementos de la Web semántica con sistemas de almacenamiento P2P, parece ser una buena aproximación para realizar actividades de uso e intercambio de información más rápida y eficiente. El uso de una ontología hace más eficiente la búsqueda en un sistema distribuido, ya que no se limita a la búsqueda por palabras clave y la consulta de un usuario tiene asociado un contexto.

El conocimiento nace de una actividad intrínsicamente social y el paradigma P2P es el que mejor modela las interacciones sociales. Los usuarios pueden compartir sus archivos, sin depositarlos en un servidor centralizado. Las consultas de los usuarios se encaminan en la red P2P hacia los nodos donde estan los recursos asociados al rango semántico de su consulta.

Entre los objetivos de esta tesis, está la representación del conocimiento de un grupo de trabajo a través de la adaptación y/o construcción de una ontología de dominio. La *Ontología del Área de Redes y Telecomunicaciones (ODARyT)* es un resultado de este trabajo de tesis. ODARyT es una ontología de dominio, que representa el conocimiento del área de Redes y Telecomunicaciones. Además ODARyT contiene conceptos que permiten la gestión de los recursos documentales de una memoria corporativa. ODARyT puede ser reutilizada en función de los intereses y objetivos que se tengan en otros grupos de trabajo del mismo dominio. Como otro resultado se propuso una metodología para la construcción de ODARyT que puede extenderse y ser reutlizada por otras organizaciones o comunidades científicas. La metodología propuesta incorpora las etapas de refinamiento y validación de la metodología IDEF5 a las etapas de especificación, adquisición del conocimiento, conceptualización, implementación y mantenimiento de la metodología METHONTOLOGY, completando el ciclo de desarrollo para una ontología de dominio.

En cuanto a la descripción de los recursos documentales pertencientes a un grupo de trabajo, hemos construido una interfaz de usuario, que permite la edición de anotaciones (descripciones semánticas de los recursos), usando los conceptos de la ontología. En particular nuestra interfaz

permite a un usuario común, realizar las anotaciones a su recursos. La interfaz propuesta es sencilla y fácil de usar para usuarios que no están familiarizados con los elementos y herramientas de la Web semántica. La interfaz también permite editar una ontología y navegar a través de la misma. Así mismo, el usuario puede consultar las anotaciones realizadas a sus recursos.

El segundo objetivo de esta tesis, fué proponer un modelo semántico de almacenamiento y recuperación de los recursos que pertencen a una memoria corporativa, comparado con trabajos anteriores, nuestra propuesta difiere en la utilización de indices semánticos basados en el contenido de una Web Semántica Corporativa (WSC). A partir de estos índices, nosotros construimos una tabla de búsqueda, que permite una recuperación rápida de los recursos. La construcción de la tabla de búsqueda implicó la propuesta de un algoritmo para resolver el problema de *gráficas empotradas*. Dado que podemos modelar una WSC como un grafo G_1 y una red de almacenamiento como otro grafo G_2, la tabla de búsqueda se construye a través de la localización de los nodos de G_1 dentro de los nodos de G_2. El problema de *gráficas empotradas* es del tipo NP-Completo y la solución a estos problemas es planteada a través del uso de una heurística. En nuestro caso usamos la heurística de la *Colonia de Hormigas*.

El procedimiento de recuperación es transparente para el usuario final, pues cada nodo de la red de almacenamiento, guarda una copia de la tabla y es capaz de recuperar los recursos solicitados por un usuario con una complejidad constante.

Durante el desarrollo de este proyecto se identificaron varios problemas abiertos. Uno de estos problemas es considerar la creación de anotaciones de manera automática, dentro de nuestra interfaz, lo que permitiría al usuario ahorrar tiempo en la descripción de sus recursos. En cuanto al almacenamiento y recuperación se deben considerar capacidades no homogéneas en los nodos de almacenamiento, así como también, el aumento de recursos asociados a un rango semántico y mejorar el algoritmo HORMIGA, para que sea más eficiente y considere los parámetros antes mencionados.

Anexo A

Conceptos de ODARyT

A continuación se presentan los conceptos de ODARyT agrupados en las 3 áreas principales del grupo de RyT y un grupo para Documento.

1. Networks and Telecommunications Services

Concepto	Definición	Ref
1.1 Reference Models		
1.1.1 The OSI Reference Model	The Open Systems Interconnection (OSI) Reference Model is a model of communications between cooperating devices. It defines a seven-layer architecture of communication functions.	[29]
1.1.2 The TCP/IP Reference Model	The complete suite of protocols including IP, TCP, and the associated application protocols.	[30]
1.1.3 Protocol Verification		
1.1.3.1 Finite State Machine Models		
1.1.3.2 Petri Net Models		
1.2 The Physical Layer	Concerned with transmission of unstructured bit stream over physical medium; deals with the mechanical, electrical, functional, and procedural characteristics to access the physical medium.	[29]
1.2.1 Theory for Data Communication	Model the behavior of the signal and analyze it mathematically.	[54]
1.2.1.1 Analog Data	Data represented by a physical quantity that is considered to be continuously variable and whose magnitude is made directly proportional to the data or to a suitable function of the data.	[29]
1.2.1.2 Analog Signal	A continuously varying electromagnetic wave that may be propagated over a variety of media.	[29]
1.2.1.3 Digital Data	Data consisting of a sequence of discrete elements.	[29]
1.2.1.4 Digital Signal	A discrete or discontinuous signal, such as voltage pulses.	[29]
1.2.1.5 Digital Transmission	The transmission of digital data, using either an analog or digital signal, in which the digital data are recovered and repeated at intermediate points to reduce the effects of noise.	[29]
1.2.1.6 Amplitude	The size or magnitude of a voltage or current waveform.	[29]
1.2.1.7 Frequency	Rate of signal oscillation in hertz.	[29]

1.2.1.8 Bandwidth	The difference between the limiting frequencies of a continuous frequency spectrum.	[29]
1.2.1.9 Attenuation	A decrease in magnitude of current, voltage, or power of a signal in transmission between points.	[29]
1.2.1.10 Noise	Unwanted signals that combine with and, hence, distort the signal intended for transmission and reception.	[29]
1.2.1.11 Crosstalk	The phenomenon in which a signal transmitted on one circuit or channel of a transmission system creates an undesired effect in another circuit or channel.	[29]
1.2.1.12 Maximun Data Rate of a Channel	Transmission capacity of a channel.	[29]
1.2.1.13 Fourier Series	A descomposition of any reasonably behaved periodic function, g(t) with period T as the sum of a (possibly infinite) number of sines and cosines.	[54]
1.2.2 Guided Transmission Media	Physical media can be used for the transmission.	[54]
1.2.2.1 Magnetic Media	One of the most common ways to transport data from one computer to another is to write them onto magnetic tape or removable media.	[54]
1.2.2.2 Coaxial Cable	A cable consisting of one conductor, usually a small copper tube or wire, within and insulated from another conductor of larger diameter, usually copper tubing or copper braid.	[29]
1.2.2.3 Twisted Pair	A transmission medium consisting of two insulated wires arranged in a regular spiral pattern.	[29]
1.2.2.4 Optical fiber	A thin filament of glass or other transparent material, through which a signal-encoded light beam may be transmitted by means of total internal reflection.	[29]
1.2.3 Wireless Transmission	Transmission without terrestrial communication infrastructure.	[54]
1.2.3.1 The Electromagnetic Spectrum	Refers to the full range of electromagnetic frequencies, which include Radio Frequency (RF).	[28]
1.2.3.2 Radio	Radio is generally refered to as electromagnetic waves whose frequencies are between 10 kHz and 300 GHz. Radio waves can travel long distances, and can penetrate buildings easily, so they are widely used for communication, both indoors, and outdoors.	[28]
1.2.3.3 Microwave	Electromagnetic waves in the frequency range of about 2 to 40 GHz.	[29]
1.2.3.4 Infrared	Infrared is the electromagnetic waves whose frequency range is above that of microwaves, but below that of the visible spectrum.	[28]
1.2.4 Communication Satellite	Refers to a space vehicle launched into orbit to relay audio, data or video signals as part of a telecommunications network. Signals are transmitted to the satellite from earth station antennas, amplified and sent back to earth for reception by other earth station antennas. Satellites are capable of linking two points, one point with many others, or multiple locations with other multiple locations.	[28]

1.2.4.1 Geostationary Earth Orbit Systems	Geo-stationary Earth Orbit Systems (GEOS) is a communications system with satellites in geosynchronous orbits – 22,300 miles above the Earth.	[28]
1.2.4.2 Medium Earth Orbit Satellites	Satellites at much lower altitudes. They have a smaller footprint on the ground and require less powerful transmitters to reach them.	[54]
1.2.4.3 Low Earth Orbit Satellites	Satellites are so close to the Earth, the ground stations do not need much power, and the round-trip delay is only a few miliseconds.	[54]
1.3 The Data Link Layer	Provides for the reliable transfer of information across the physical link; sends blocks of data (frames) with the necessary synchronization, error control, and flow control.	[29]
1.3.1 Flow Control	The function performed by a receiving entity to limit the amount or rate of data that is sent by a transmitting entity.	[29]
1.3.1.1 Stop-and-Wait Protocol	A flow control protocol in which the sender transmits a block of data and then awaits an acknowledgment before transmitting the next block.	[29]
1.3.1.2 Sliding Window Protocol	A method of flow control in which a transmitting station may send numbered packets within a window of numbers. The window changes dynamically to allow additional packets to be sent.	[29]
1.3.2 Error Control	Refers to the techniques for detecting and correcting errors in data transmissions.	[28]
1.3.2.1 Error-Detecting Code	A code in which each expression conforms to specific rules of construction, so that if certain errors occur in an expression, the resulting expression will not conform to the rules of construction, and thus the presence of the errors is detected.	[29]
1.3.2.2 Error-Correcting Code	Error-Correcting Code (ECC) is a code in which each data signal conforms to specific rules of construction so that departures from this construction in the received signal can generally be automatically detected and corrected. It is used in computer data storage, for example in dynamic RAM, and in data transmission. Examples include Hamming code, Reed-Solomon code, Reed-Muller code, Binary Golay code, convolutional code, turbo code and others. The simplest error correcting codes can correct single-bit errors (single error correction) and detect double-bit errors (double error detection). Other codes can detect or correct multi-bit errors.	[28]
1.3.2.2.1 Hamming Code	Hamming code, sometimes referred to as an Error Correction Code (ECC), is an algorithm that can be used to detect errors in individual bits of transmitted data, and sometimes (dependent on the exact code used) correct that error. Although not particularly powerful, they are one of the "perfectçodes in that its standard array has all of the error patterns that can exist for single errors.	[28]

1.3.2.2.2 Reed-Solomon Code	Reed-Solomon codes are block-based error correcting codes with a wide range of applications in digital communications and storage. Reed-Solomon codes are used to correct errors in many systems including: (1) Storage devices (including tape, Compact Disk, DVD, barcodes, etc) (2) Wireless or mobile communications (including cellular telephones, microwave links, etc) (3) Satellite communications. (4) Digital television / DVB. (5) Highspeed modems such as ADSL, cDSL, etc.	[28]
1.3.3 High Level Data Link Control (HDLC)	Very common, bit-oriented data link protocol - (layer 2) issued by ISO. Similar protocols are LAPB, LAPD, LLC.	[29]
1.3.4 Point-to-Point Protocol (PPP)	The Point-to-Point Protocol (PPP) suite provides a standard method for transporting multi-protocol datagrams over point-to-point links.	[28]
1.3.5 Medium Access Control	For broadcast networks, the method of determining which device has access to the transmission medium at any time.	[29]
1.3.5.1 ALOHA	A medium access control technique for multipleaccess transmission media. A station transmits whenever it has data to send. Unacknowledged transmissions are repeated.	[29]
1.3.5.2 CSMA	(Carrier Sense Multiple Access) A medium access control technique for multiple-access transmission media. A station wishing to transmit first senses the medium and transmits only if the medium is idle.	[29]
1.3.5.3 Token Bus	A medium access control technique for busltree. Stations form a logical ring, around which a token is passed. A station receiving the token may transmit data and then must pass the token on to the next station in the ring.	[29]
1.3.5.4 Token Ring	A medium access control technique for rings. A token circulates around the ring. A station may transmit by seizing the token, inserting a packet onto the ring, and then retransmitting the token.	[29]
1.3.6 Local Area Networks (LAN)	A communication network that provides interconnection of a variety of data communicating devices within a small area.	[29]
1.3.6.1 Logical Link Control (LLC)	Logic Link Control (LLC) is the IEEE 802.2 LAN protocol that specifies an implementation of the LLC sublayer of the data link layer. IEEE 802.2 LLC is used in IEEE802.3 (Ethernet) and IEEE802.5 (Token Ring) LANs to perform some functions.	[28]
1.3.6.2 FDDI	Fiber Distributed Data Interface (FDDI) is a set of ANSI protocols for sending digital data over fiber optic cable. FDDI networks are token-passing (similar to IEEE 802.5 Token Ring protocol) and dual-ring networks, and support data rates of up to 100 Mbps. FDDI networks are typically used as backbones technology because of its support for high bandwidth and great distance.	[28]
1.3.6.3 Ethernet	The name of the LAN invented at the Xerox Corporation Palo Alto Research Center. It operates using the CSMA/CD medium access control method.	[30]

1.3.6.4 Topology	The structure, consisting of paths and switches, that provides the communications interconnection among nodes of a network.	[29]
1.3.6.4.1 Ring	A local-network topology in which stations are attached to repeaters connected in a closed loop. Data are transmitted in one direction around the ring, and can be read by all attached stations.	[29]
1.3.6.4.2 Star	A topology in which all stations are connected to a central switch. Two stations communicate via circuit switching.	[29]
1.3.6.4.3 Bus	All nodes on the LAN are connected by one linear cable, which is called the shared medium. Every node on this cable segment sees transmissions from every other station on the same segment. At each end of the bus is a terminator, which absorbs any signal, removing it from the bus.	[28]
1.3.6.4.4 Tree	The tree topology is a logical extension of the bus topology. The transmission medium is a branching cable with no closed loops. The tree layout begins at a point called the head-end, where one or more cables start, and each of these may have branches. The branches in turn may have additional branches to allow quite complex layouts.	[28]
1.3.6.5 Wireless LAN	A LAN that uses either radio or infrared as the transmission medium. Requires different MAC	[30]
1.3.7 Metropolitan Area Network (MAN)	Metropolitan Area Network (MAN) is a computer networks usually spanning a campus or a city, which typically connect a few local area networks using high speed backbone technologies. A MAN often provides efficient connections to a wide area network (WAN). Generally, a MAN spans a larger geographic area than a LAN, but a smaller geographic area than a WAN.	[28]
1.3.8 Wide Area Network (WAN)	A Wide Area Network (WAN) is a computer network covering multiple distance areas, which may spread across the entire world. WANs often connect multiple smaller networks, such as local area networks (LANs) or metro area networks (MANs). The world's most popular WAN is the Internet. Some segments of the Internet are also WANs in themselves. A Wide Area Network may be privately owned or rented from a service provider, but the term usually connotes the inclusion of public (shared user) networks.	[28]
1.3.9 Bluetooth	Bluetooth becomes widely used specification for wireless communications among portable digital devices including notebook computers, peripherals, cellular telephones, beepers, and consumer electronic devices.	[28]
1.3.10 Frame Relay	A form of packet switching based on the use of variable-length, linklayer frames. There is no network layer, and many of the basic functions have been streamlined or eliminated to provide for greater throughput. Frequency Rate of signal oscillation in hertz.	[29]

1.3.11 Asynchronous Transfer Mode (ATM)	The proposed mode of operation of the emerging broadband integrated services digital network. All information to be transmitted -voice, data, image, video- ia first fragmented into small, fixed-sized frames known as cells. These are switching and routed using packet switching principles.	[30]
1.3.12 Data Link Layer Switching	Examine the data link layer addresses to do routing.	[54]
1.3.12.1 Bridge	A functional unit that interconnects two local area networks (LANs) that use the same logical link control protocol but may use different mediumaccess control protocols.	[29]
1.3.12.2 Hub	The Hub, also called repeater, is a device that accepts Ethernet connections from network devices and cross-connects them. Data arriving via the receive pair of one connection is regenerated and sent out on the transmit pair to all connected devices except for the device who originated the transmission.	[28]
1.3.12.3 Switch	A switch is a networking device that connects network segments. Technically, network switches operate at layer two (Data Link Layer) of the OSI model.A switch is similar to a hub in that it provides a single broadcast domain, but differs in that each port on a switch is its own collision domain.	[28]
1.4 The Network Layer	Provides upper layers with independence from the data transmission and switching technologies used to connect systems; responsible for establishing, maintaining, and terminating connections.	[29]
1.4.1 Router	A router is a device or a piece of software in a computer that forwards and routes data packets along networks. A router connects at least two networks, commonly two LANs or WANs or a LAN and its ISP network. A router is often included as part of a network switch. A router is located at any gateway where one network meets another, including each point-of-presence on the Internet.	[28]
1.4.2 Routing Algorithms	Algorithms selection of a suitable path through a network	[55]
1.4.2.1 Shortest Path Routing	The idea is to build a graph of the subnet, with each node of the graph representing a router and each arc of the graph representing a communication line (often called a link). To choose a route between a given pair of routers, the algorithm just finds the shortest path between them on the graph.	[54]
1.4.2.2 The Distance Vector Routing Algorithm	Dynamic algorithm. Each router maintains a routing table indexed by, and containing one entry for, each router in the subnet. Operate by having each router maintain a table giving the best know distanceto each destination and hich line to use to get there. These tables are updated by exchanging information with the neighbors.	[54]

1.4.3 Congestion Control	Congestion is when too many packets are present in (a part of) the subnet, performance degrades. When the number of packets dumped into the subnet by the hosts in within its carrying capacity, they are all derivered and the number delivered is proportional to the number sent.	[54]
1.4.4 Virtual-Circuit Subnet	A packet-switching service in which a connection (virtual circuit) is established between two stations at the start of transmission. All packets follow the same route; they need not carry a complete address, and they arrive in sequence.	[29]
1.4.5 Store and Forward Switching	Store And Forward Switching refers to a switching technique in which frames are completely processed before being forwarded out the appropriate port. This processing includes calculating the Cyclic Redundancy Check (CRC) and checking the destination address. In addition, frames must be temporarily stored until network resources are available to forward the message.	[28]
1.4.6 The Internet Protocol (IP)	An internetworking protocol that provides connectionless service across multiple packet-switching networks.	[29]
1.4.6.1 Datagram	In packet switching, a packet, independent of other packets, that carries information sufficient for routing from the originating data terminal equipment (DTE) to the destination DTE without the necessity of establishing a connection between the DTEs and the network.	[29]
1.4.6.2 Address	Address is a data structure or logical convention used to identify a unique entity, such as a particular process or a network device. Important addresses include IP address for network routing and MAC address for a hardware device.	[28]
1.4.6.3 Address Resolution	Address Resolution refers to the process of translating or expressing the address of an entity on one system to the equivalent address of the same entity in the second system, when two addressing systems refer to the same entity. For instance, translating an IP address to its given DNS name, or translating an IP address to its MAC address.	[28]
1.4.6.4 Internet Control Message Protocol(ICMP)	Internet control message protocol. A component part of the internet protocol in the TCP/IP suite that handles error and other control messages that are returned by internet gateways and hosts.	[30]
1.4.7 IPv6	IPv6 is the new version of Internet Protocol (IP) based on IPv4, a network-layer (Layer 3) protocol that contains addressing information and some control information enabling packets to be routed in the network. IPv6 increases the IP address size from 32 bits to 128 bits, to support more levels of addressing hierarchy, a much greater number of addressable nodes, and simpler auto-configuration of addresses.	[28]
1.4.8 Routing in the Internet		

1.4.8.1 The Interior Gateway Protocol (IGP)	Interior Gateway Protocol. The routing protocol used in the gateways of a TCP/IP internetwork to obtain the shortest path routes through the internet.	[30]
1.4.8.2 The Exterior Gateway Protocol (BGP)	Exterior Gateway Protocol. A protocol used in relation to large internetworks that comprise multiple smaller internetworks interconnected together. The interconnection devices used are know as exterior gateways and the EGP is the protocol they use to advertise the IP addresses of the networks present in each of the smaller internetworks.	[30]
1.4.9 Internetworking	Communication among devices across multiple networks.	[29]
1.4.9.1 End System	A device attached to one of the subnetworks of an internet that is used to support end-user applications or services.	[29]
1.4.9.2 Intermediate System	A device used to connect two subnetworks and permit communication between end systems attached to different subnetworks.	[29]
1.4.9.3 Subnetwork	Refers to a constituent network of an internet. This avoids ambiguity since the entire internet, from a user's point of view, is a single network.	[29]
1.5 The Transport Layer	Provides reliable, transparent transfer of data between end points; provides end-to-end error recovery and flow control.	[29]
1.5.1 The Transport Services	Providing communication services directly to the application processes running on different hosts.	[56]
1.5.2 Multiplexing	In data transmission, a function that permits two or more data sources to share a common transmission medium such that each data source has its own channel.	[29]
1.5.3 Connectionless Transport: UDP	In a unreliable, connectionless protocol for applications that do not want TCP's sequencing or flow control and wish to provide their own.	[54]
1.5.3.1 UDP Segment Structure	The UDP header has only four fields, each consisting of four bytes: Source Port, Destination Port, Length and UDP Checksum.	[56]
1.5.3.2 UDP Checksum	The UDP checksum provides for error detection. UDP at the sender side performs the one's complement of the sum of all the 16-bit words in the segment. This result is put in the checksum field of the UDP segment.When the segment arrives (if it arrives!) at the receiving host, all 16-bit words are added together, including the checksum. If this sum equals 1111111111111111, then the segment has no detected errors. If one of the bits is a zero, then we know that errors have been introduced into the segment.	[56]
1.5.4 Connection-Oriented Transport: TCP	Is a reliable connection-oriented protocol that allows a byte stream originating on one machine to be delivered without error on any other machine in the internet.	[54]

1.5.4.1 TCP Segment Structure	The TCP segment consists of header fields and a data field. The32-bit sequence number field, and the 32-bit acknowledgment number field are used by the TCP sender and receiver in implementing a reliable data transfer service. The 16-bit window size field is used for the purposes of flow control. The 4-bit length field specifies the length of the TCP header in 32-bit words. The optional and variable length options field is used when a sender and receiver negotiate the maximum segment size (MSS) or as a window scaling factor for use in high-speed networks. The flag field contains 6 bits.	[56]
1.5.4.2 Reliable Data Transfer	TCP's reliable data transfer service ensures that the data stream that a process reads out of its TCP receive buffer is uncorrupted, without gaps, without duplication, and in sequence, i.e., the byte stream is exactly the same byte stream that was sent by the end system on the other side of the connection.	[56]
1.6 The Session Layer	Provides the control structure for communication between applications; establishes, manages, and terminates connections (sessions) between cooperating applications.	[29]
1.7 The Presentation Layer	Provides independence to the application processes from differences in data representation (syntax).	[29]
1.8 The Application Layer	Provides access to the OSI environment for users and also provides distributed information services.	[29]
1.8.1 Web: HTTP	The Hypertext Transfer Protocol (HTTP) is an application-level protocol with the lightness and speed necessary for distributed, collaborative, hypermedia information systems. HTTP allows an open-ended set of methods to be used to indicate the purpose of a request.	[28]
1.8.2 File Transfer: FTP	File Transfer Protocol. The application protocol in the TCP/IP suite that provides acess to a networked file server.	[30]
1.8.3 Electronic Mail: SMTP	Simple Mail Transfer Protocol (SMTP) is a protocol designed to transfer electronic mail reliably and efficiently. SMTP is a mail service modeled on the FTP file transfer service. SMTP transfers mail messages between systems and provides notification regarding incoming mail.	[28]
1.8.4 Name Translation: DNS	DNS (Domain Name System or Service) is a distributed Internet directory service. DNS is used mostly to translate between domain names (www.domainname.com) and IP addresses (123.123.123.123), and to control Internet email delivery. Most Internet services rely on DNS to work, and if DNS fails, web sites cannot be located and email delivery stalls.	[28]

1.8.5 Remote Terminal Access: Telnet	TELNET is the terminal emulation protocol in TCP/IP environment. TELNET uses the TCP as the transport protocol to establish connection between server and client. After connecting, TELNET server and client enter a phase of option negotiation that determines the options that each side can support for the connection. Each connected system can negotiate new options or renegotiate old options at any time. In general, each end of the TELNET connection attempts to implement all options that maximize performance for the systems involved.	[28]
1.8.6 Network Management: SNMP	Simple Network Management Protocol (SNMP) is the protocol developed to manage nodes (servers, workstations, routers, switches and hubs etc.) on an IP network. SNMP enables network administrators to manage network performance, find and solve network problems, and plan for network growth. Network management systems learn of problems by receiving traps or change notices from network devices implementing SNMP.	[28]
1.8.7 Remote File Server: NFS	NFS was designed for remote file access and sharing over network with various types of machines, operating systems, network architecture and transport protocols. NFS uses a client/server architecture and consists of a client program and a server program.	[28]
1.8.8 Content Distribution		
1.8.9 Multimedia	Generally mean the combination of two or more continuous media, that is, media that have to be played during some well-defined time interval, usually with some user interaction.	[54]
1.8.9.1 Audio Compression	Audio compression is a form of data compression designed to reduce the size of audio files.	[28]
1.8.9.2 Streaming Audio	Listening to sound over the Internet. This is also called music on demand.	[54]
1.8.9.3 Voice over IP	Voice over IP (VOIP) refers to using a group of technologies to transmit voice, as well as video signals, as packets over an IP network. VOIP is replacing the traditional PBX and PSTN technologies to become the main stream in corporate and public telecommunications.	[28]
1.8.9.4 Video Compression	Video compression is a form of encoding and decoding to reduce the size of video files.	[54]
1.8.9.5 Video on Demand	Video on Demand(VoD) is a service using video compression to supply video programs to viewers when requested via ISDN or cable.	[28]
1.9 Network Security	Collection of tools designed to protect data. Network security measures are needed to protect data during their transmission, and to guarantee that data transmissions are authentic	[29]
1.9.1 Cryptography	The branch of criptology dealing with the design of algorithms for encryption and decryption, intended to ensure the secrecy and/or authenticity of messages.	[29]

1.9.1.1 Symmetric Cryptography	Symmetric Cryptography is a branch of cryptography involving algorithms that use the same key for two different steps of the algorithm (such as encryption and decryption, or signature creation and signature verification). Symmetric cryptography is sometimes called "secret-key cryptography" (versus public-key cryptography) because of the entities that share the key.	[28]
1.9.1.1.1 Symetric-Key Algorithm	Symmetric key algorithm, also known as secret key algorithm, is a mathematical algorithm used in secret key encryption.	[28]
1.9.1.1.1.1 DES	Data Encryption Standard (DES) is a long-standing US encryption standard with symmetric-key encryption method standardized by ANSI in 1981 as ANSI X.3.92. DES uses a 56-bit key and uses the block cipher method, which breaks text into 64-bit blocks and then encrypts them. There are 72 quadrillion or more possible encryption keys that can be used in this algorithm. Like other private key cryptographic methods, both the sender and the receiver must know and use the same private key.	[28]
1.9.1.1.1.2 AES	The Advanced Encryption Standard (AES), also known as Rijndael, is a block cipher adopted as an encryption standard developed by NIST. AES is intended to specify an unclassified, publicly-disclosed, symmetric encryption algorithm. AES has a fixed block size of 128 bits and a key size of 128, 192 or 256 bits.	[28]
1.9.1.2 Public-Key Cryptography	Public Key Cryptography is also known as asymmetric cryptography which is based on the mathematic scheme developed by Diffie and Hellman. Asymmetric cryptography uses different (but related) keys for encryption and decryption. It is also called public key cryptography because the encryption key is made public while the decryption key is kept private. The public key cryptography process allows any person to encrypt a message and send it to another person without prior key exchange.	[28]
1.9.1.2.1 Public-Key Algorithm	One of the two keys use in a symmetric encryption system. The public key is made public, to be used in conjunction with a corresponding private key.	[29]
1.9.1.2.1.1 RSA Algorithm	A public-key encryption algorithm based on exponentiation in modular arithmetic. It is the only algorithm generally accepted as practical and secure for public-key encryption.	[29]
1.9.2 Digital Signatures	An authentication mechanism that enables the creator of a message to attach a code that acts as a signature. The signature guarantees the source and integrity of the message.	[29]
1.9.3 Communication Security		
1.9.3.1 Firewalls		
1.9.3.2 Virtual Private Networks		
1.9.3.3 Wireless Security		

1.9.4 Authentication	A process uses to verify the integrity of transmitted data,especially a message.	[29]
1.10 Performance Evaluation		
1.10.1 Poisson Process		
1.10.1.1 Little's Theorem		
1.10.2 Queueing Models		
1.10.2.1 The M/M/1 Queueing System		
1.10.2.2 The Markov Systems		
1.10.2.3 The M/G/1 System		
1.11 Standards and Specification Bodies		
1.11.1 International Telecommunications Union (ITU)	International Telecommunications Union (ITU) is the leading United Nations agency for information and communication technologies. As the global focal point for governments and the private sector, ITU's role in helping the world communicate spans 3 core sectors: radiocommunication, standardization and development. ITU also organizes TELECOM events and was the lead organizing agency of the World Summit on the Information Society.	[57]
1.11.2 American National Standards Institute (ANSI)	American National Standards Institute. A national standards organization comprising members from computers manufactures and users in the United States. Its members are involved in the development of standards at all levels in the ISO reference model.	[30]
1.11.3 European Telecommunications Standards Institute (ETSI)	European Telecommunications Standards Institute. A European standards body that produces standards for regulatory purposes by the EU and EFTA countries. It produces standards relating to telecommunication services, public data networks, videotex, and digital cellular services which are issued as European Telecommunication Standards (ETSs), also know as NET's.	[30]
1.11.4 Institute for Electrical and Electronics Engineers (IEEE)	Institute of Electrical and Electronics Engineers. A US professional institute and standardization.	[55]
1.11.5 International Organization for Standarization (ISO)	International Organization for Standardization. An international standards organization comprising designated standards bodies of the participating contries. It is concerned with a wide range of standards, each of which is controlled by a separate technical committee.	[30]

2. Distributed Systems

Concepto	Definición	Ref
2.1 Architecture Model	The division of responsibilities between system components (applications, servers and other processes) and the placement of the components on computers in the network	[58]

2.1.1 Client/server	A service is owned by a particular machine at which a server administers that service.Clients run on the user machines and facilitate user consumption of that service.	[59]
2.1.2 Peer to Peer	In this architecture all of the processes involved in a task or activity play similar roles, interacting cooperatively as peers without any distinction between client and server processes or the computers that they run on.	[58]
2.1.3 Middleware	The term middleware applies to a software layer that provides a programming abstraction as well as masking the heterogeneity of the underlying networks, hardware, operating systems and programming languages.	[58]
2.2 Distributed Objects	Extends the model supported by object-oriented programming languages to make it apply to distributed objects.	[58]
2.2.1 Remote procedure call (RPC)	A remote procedure call is very similar to a remote method invocation in that a client program calls a procedure in another program running in a server process.	[58]
2.2.2 Events	The actions where one object can react to a change occurring in another object.	[58]
2.2.3 Notifications	Objects that represent events.	[58]
2.3 Operating System	Software that controls the management and the execution of programs.	[58]
2.3.1 Processes	A process consists of an execution environment together with one or more threads. An execution environment is the unit of resource management: a collection of local kernel-managed resources to which its threads have access.	[58]
2.3.2 Threads	A thread is the operating system abstraction of an activity (the term derives from the phrase "thread of execution")	[58]
2.3.3 Invocation		
2.3.4 Process Management		
2.3.4.1 Synchronization	The programming of different activities so that they operate in step with each other	[59]
2.3.4.1.1 Clocks	Electronic devices that count oscillations ocurring in a crystal at a definite frequency, and that typically divide this count and store the result in a counter register.	[58]
2.3.4.1.2 Logical Time		
2.3.4.1.3 Logical Clock	A Lamport logical clock is a monotonically increasing software counter, whose value need bear no particular relationship to any physical clock.	[58]
2.3.4.2 Concurrency	The facility to accommodate many of the same type of activity at the same time. For example, a number of transactions may access the same file at the same time.	[59]
2.3.4.2.1 Locks		

2.3.4.2.2 Timestamp ordering	Each operation in a transaction is validated when it is carried out. If the operation cannot be validated, the transaction is aborted inmediately and can then be restarted by the client. Each transaction is assigned a unique timestamp value when it starts. The timestamp defines its position in the time sequence of transactions.	[58]
2.3.4.3 Coordination and Fault Tolerance		
2.3.4.3.1 Distributed mutual exclusion	If a collection of processes share a resource or collection of resources, then often mutual exclusion is required to prevent interference and ensure consistency when accessing the resources. In a distributed system, the distributed mutual exclusion: one that is based solely on message passing.	[58]
2.3.4.3.2 Elections	An algorithm for choosing a unique process to play a particular role.	[58]
2.3.4.3.3 Multicast	Operation to send a message to each of a group of process.	[58]
2.3.4.3.4 Consensus	The problem is for processes to agree on a value after one or more of the processes has proposed what value should be.	[58]
2.3.4.3.5 Fault Tolerance	Dependable applications should continue to function correctly in the presence of faults in hardware, software and networks.	[58]
2.4 Distributed File Systems	Is a network file system where a single file system can be distributed across several physical computer nodes. Separate nodes have direct access to only a part of the entire file system, in contrast to shared disk file systems where all nodes have uniform direct access to the entire storage.	[58]
2.5 Distributed databases	A collection of cooperating database systems each at a separate site.	[59]
2.6 Peer to Peer Systems	This are systems have no centralized control or hierarchical organization, where the software running at each node is equivalent in functionality.	[60]
2.7 Global state	The set of attributes of an entity, described at a particular time, when that set is extended to every occurrence of that entity within a prescribed boundary. The complete set of attributes necessary to describe an entity at a particular time.	[61]
2.8 Distributed Transactions		
2.8.1 Atomic commit protocols	The atomicity of transactions requires that when a distributed transaction comes to an end, either all of its operations are carried out or none of them.	[58]
2.8.2 Distributed deadlocks		
2.9 Replication	To hold a copy of a file at another site.	[59]
2.10 Mobile Computing	Mobile computing is the performance of computing tasks while the user is on the move, or visiting places other than their usual environment. Users who are away from their 'home' intranet are still provide with access to resources via the devices they carry with them.	[58]

2.11 Ubiquitous Computing	Ubiquitous computing is the harnessing of many small, cheap computational devices that are present in users' physical environments, including the home, office and even natural settings. The term 'ubiquitous' is intended to suggest that small computing devices will eventually become so pervasive in everyday objects that they are scarcely noticed.	[58]
2.12 Distributed Multimedia System	Applications can handle streams of continuous, timebased data such as digital audio and video.	[58]
2.12.1 Multimedia Data	It's referred to video and audio data as continuous and time-based.	[58]
2.12.2 Quality of service management	The planned allocation and scheduling of resources to meet the needs of multimedia and other applications.	
2.13 Distributed Shared Memory		
2.13.1 Consistency	All the copies of data areas reflect the same state.	[59]
2.14 Web Services	A web service interface generally consists of a collection of operations that can be used by a client over the Internet. The operations in a web service may be provided by a variety of different resources, for example, programs, objects or databases.	[58]

3. Digital Communication Systems

Concepto	Definición	Ref
3.1 Information source	The user providing the information to be transferred to a destination user during a particular information transfer transaction. Synonym information source.	[61]
3.2 Source coding		
3.3 Source decoding		
3.4 Channel coding	A single path provided by a transmission medium via either (a) physical separation, such as by multipair cable or (b) electrical separation, such as by frequency- or time-division multiplexing.	[61]
3.4.1 Line code	A code chosen for use within a communications system for transmission purposes. Note 1: A line code may differ from the code generated at a user terminal, and thus may require translation. Note 2: A line code may, for example, reflect a requirement of the transmission medium, e.g., optical fiber versus shielded twisted pair.	[61]
3.4.2 Block code	An error detection and/or correction code in which the encoded block consists of N symbols, containing K information symbols (K¡N) and N-K redundant check symbols, such that most naturally occurring errors can be detected and/or corrected.	[61]
3.4.3 Cyclic code	An error checking mechanism that checks data integrity by computing a polynomial algorithm based checksum.	[61]

3.4.4 Convolutional code	A type of error-correction code in which (a) each m -bit information symbol (each m-bit string) to be encoded is transformed into an n -bit symbol, where n ¿m and (b) the transformation is a function of the last k information symbols, where k is the constraint length of the code. Note: Convolutional codes are often used to improve the performance of radio and satellite links.	[61]
3.4.5 Treillis code		
3.4.6 Fire code		
3.4.7 Reed-Muller code		
3.5 Channel decoding		
3.6 Communication channel	A connection between initiating and terminating nodes of a circuit.	[61]
3.6.1 Telephone lines	The branch of science devoted to the transmission, reception, and reproduction of sounds, such as speech and tones that represent digits for signaling.	[61]
3.6.2 Radio links	A transmitter, receiver, or transceiver used for communication via electromagnetic waves. A general term applied to the use of radio waves.	[61]
3.6.3 Microwaves links	An electromagnetic wave having a wavelength from 300 mm to 10 mm (1 GHz to 30 GHz). Note: Microwaves exhibit many of the properties usually associated with waves in the optical regime, e.g. , they are easily concentrated into a beam.	[61]
3.6.4 Satellite links	A radio link between a transmitting Earth station and a receiving Earth station through one satellite. A satellite link comprises one uplink and one downlink.	[61]
3.6.5 Hard disk	A flat, circular, rigid plate with a magnetizable surface on one or both sides of which data can be stored.	[61]
3.6.6 Magnetic tapes	A tape with a magnetizable surface on which data can be stored and retrieved. 2. A tape or ribbon of any material impregnated or coated with magnetic or other material on which information may be placed in the form of magnetically polarized spots.	[61]
3.7 Modulation	The process, or result of the process, of varying a characteristic of a carrier, in accordance with an information-bearing signal.	[61]
3.7.1 Single Side Band (SSB)	An amplitude modulated emission with one sideband only	[61]
3.7.2 Double Side Band (DSB)	AM transmission in which both sidebands and the carrier are transmitted.	[61]
3.7.3 Baseband	Baseband is a type of a network technology where only one carrier frequency is used. In a baseband network, information is carried in digital form on a single unmultiplexed signal channel on the transmission medium. Ethernet and Token Ring are examples of a baseband network. Baseband in signal transmisson may mean a signal's bandwidth before modulation and multiplexing, or after demultiplexing and demodulation.	[28]

3.7.4 Frecuency band	Electromagnetic spectrum. The range of frequencies of electromagnetic radiation from zero to infinity. Note: The electromagnetic spectrum was, by custom and practice, formerly divided into 26 alphabetically designated bands. This usage still prevails to some degree. However, the ITU formally recognizes 12 bands, from 30 Hz to 3000 GHz. New bands, from 3 THz to 3000 THz, are under active consideration for recognition.	[61]
3.7.5 Amplitude band	Modulation in which the amplitude of a carrier wave is varied in accordance with some characteristic of the modulating signal. Note: Amplitude modulation implies the modulation of a coherent carrier wave by mixing it in a nonlinear device with the modulating signal to produce discrete upper and lower sidebands, which are the sum and difference frequencies of the carrier and signal. The envelope of the resultant modulated wave is an analog of the modulating signal. The instantaneous value of the resultant modulated wave is the vector sum of the corresponding instantaneous values of the carrier wave, upper sideband, and lower sideband. Recovery of the modulating signal may be by direct detection or by heterodyning	[61]
3.7.6 Digital		
3.7.6.1 Pulse Code Modulation (PCM)	Pulse-code Modulation (PCM) is a sampling technique for digitizing analog signals, especially voice/audio signals. It samples the analog signals 8000 times per second; each sample is represented by 8 bits for a total of 64 Kbps. There are two standards for coding the sample level. The Mu-law is used in North America and Japan while the A-law is used in Europe and most other countries.	[28]
3.7.6.2 Differencial Pulse Code Modulation (DPCM)	Differential Pulse-Code Modulation (DPCM) is a PCM technique that codes the difference between sample points to compress the digital data. Because audio waves propagate in predictable patterns, DPCM predicts the next sample and codes the difference between the prediction and the actual point. The differences are smaller numbers than the numerical value of each sample on the full scale and thereby reduce the resulting bitstream	[28]
3.7.6.3 PSK	In digital transmission, angle modulation in which the phase of the carrier is discretely varied in relation either to a reference phase or to the phase of the immediately preceding signal element, in accordance with data being transmitted.	[61]
3.7.6.4 DPSK	Differential Phase Shift Keying (DPSK) is a digital modulation format where information is conveyed in phase difference of a carrier signal between consecutive symbols.	[28]
3.8. Demodulation	The recovery, from a modulated carrier, of a signal having substantially the same characteristics as the original modulating signal.	[61]
3.8.1 Single Side Band (SSB)		

3.8.2 Double Side Band (DSB)		
3.8.3 Baseband		
3.8.4 Frecuency band		
3.8.5 Amplitude band		
3.8.6 Digital		
3.8.6.1 Pulse Code Modulation (PCM)		
3.8.6.2 Differencial Pulse Code Modulation (DPCM)		
3.8.6.3 PSK		
3.8.6.4 DPSK		
3.9 Filters	A device which transmits only part of the incident energy and may thereby change the spectral distribution of energy	[61]
3.9.1 Analogical filters		
3.9.1.1 Low-pass filter	A filter network that passes all frequencies below a specified frequency with little or no loss, but strongly attenuates higher frequencies.	[61]
3.9.1.2 High-pass filter	A filter that passes frequencies above a given frequency and attenuates all others.	[61]
3.9.1.3 Band-pass filter	A filter that ideally passes all frequencies between two non-zero finite limits and bars all frequencies not within the limits.	[61]
3.9.2. Digital filters	A filter (usually linear), in discrete time, that is normally implemented through digital electronic computation. Note: Digital filters differ from continuous time filters only in application. The parameters of digital filters are generally more stable than the parameters of commonly used analog (continuous) filters. Digital filters can be applied as optimal estimators. Commonly used forms are finite impulse response (FIR) and infinite impulse response (IIR).	[61]
3.9.2.1 FIR		
3.9.2.2 IRR		
3.9.2.2.1. Adaptatives		
3.9.2.2.2. Non Adaptative		
3.10. Noise	An undesired disturbance within the frequency band of interest; the summation of unwanted or disturbing energy introduced into a communications system from man-made and natural sources. A disturbance that affects a signal and that may distort the information carried by the signal.	[61]
3.10.1. Gaussian		
3.10.2 Interference	In general, extraneous energy, from natural or manmade sources, that impedes the reception of desired signals.	[61]
3.10.3 Hidden terminal		
3.10.4 Thermal		
3.11 Connectivity Type		

3.11.1 WirelineConnectivity Type	Fixed, wireline connectivity type (e.g., PSTN)	[33]
3.11.2 WirelessConnectivity Type	Wireless connectivity type (e.g., GSM connectivity)	[33]
3.11.3 LongRangeConnectivity Type	Long range wireless connectivity, such as WiMAX or GSM	[33]
3.11.3.1 WiMAX Connectivity		
3.11.4 ShortRange Connectivity	Short range wireless connectivity type (WiFi, Bluetooth)	[33]
3.11.4.1 IrDA Connectivity		
3.11.4.2 BluetoothConnectivity		
3.11.4.3 WiFiConnectivity		
3.11.5 Cellular Connectivity type	Cellular connectivity, e.g., GSM	[33]
3.11.5.1 GSMConnectivity		
3.11.5.2 UMTSConnectivity		
3.11.6 User Terminal	Class of user terminals.	[33]
3.11.7 OneWayCommunication	One way communication, i.e., one participant sends data (or information), another one receives data. An example for this is a SMS message exchange. This one covers connection as well as connectionless communication.	[33]
3.11.8 TwoWayCommunication	Two way communication. It covers all phone calls, instant messaging and (most of) data sessions.	[33]
3.11.9 SessionCommunication	Two way session communication, i.e. the two way stateful communication.	[33]
3.11.10 ChatSession	Chat session (e.g., IRC)	[33]
3.11.11 InstantMessaging Session	Instant messaging session (e.g., ICQ)	[33]
3.11.12 CallSession	Telephone connection (e.g., PSNT, VoIP or similar connection)	[33]
3.11.12.1 VoiceCall	Telephone call session between two or more parties that includes voice-only communication. Notice, that this may include also a conference call	[33]
3.11.12.2 MultimediaCall	Telephone call session between two or more parties that exchange of multimedia content	[33]
3.11.12.3 CallParticipant	Call participant role is a person (or an entity) which takes part in a (phone) call	[33]
3.11.12.4 Caller	Call participant that initiates the call	[33]
3.11.12.5 Callee	Call participant that answers a call	[33]
3.11.12.6 Call Initiation	Act of initiating of a (telephone) call session	[33]

4. Document

Concepto	Definición	Ref
4.1 Abstract	A summarization of another document.	[34]
4.2 Comment		
4.3 Correspondence	A letter, an e-mail or any other text-based communication	[62]
4.3.1 Discussion	Oral, and sometimes written, exchange of opinions - usually to analyze, clarify, or reach conclusions about issues, questions, or problems.	

4.3.2 Email	E-mail	[63]
4.3.3 Letter		
4.3.4 Postcard		
4.4 Form	A template to be filled in by an applicant for a service or other resource	
4.5 Guideline		
4.6 Homepage	Web site main page	
4.6.1 OrganizationHomepage	A organization homepage of some organization	
4.6.2 PersonalHomepage		
4.7 Index		
4.8 Lecture	Teaching method in which information is presented orally to a class with a minimal amount of class participation.	
4.9 Manuscript	For unpublished texts not described elsewhere.	[34]
4.10 Minutes	A summary of a meeting.	[34]
4.11 Preprint		
4.12 Promotion		
4.13 Publication	A printed work offered for distribution.	[34]
4.13.1 Article	Objects where the contents are of such length and/or self-contained in subject matter that the author would consider this as an article. The purpose of the contents is to fully realize a particular objective in a relatively concise form. This class includes essays, stories, preprints, and other short written forms.	[34]
4.13.1.1 BookArticle	The book where the article is published	[63]
4.13.1.2 ConferencePaper		
4.13.1.3 JournalArticle	Article in Journal	[63]
4.13.1.4 WorkshopPaper	Article in Workshop	[63]
4.13.2. Book		
4.13.3 Dictionary	An organized list of terms and their definitions.	[34]
4.13.4 Editorial		
4.13.5 Manual	A reference book for giving instructions.	[34]
4.13.6 Periodical		
4.13.6.1 Journal	Publishing vehicle for formal papers - often scientific or technical, or relating to a trade or profession	
4.13.6.2 Magazine	A periodical publication for general interest such as news, current events, and popular material.	
4.13.6.3 Newsletter		
4.13.6.4 Newspaper		
4.13.7 Proceedings	Conference proceedings.	
4.13.8 Regulation		
4.13.9 Specification		
4.13.10 TechnicalReport		
4.13.11 Thesis		

4.13.11.1 DoctoralThesis	A PhD thesis.	
4.13.11.2 MastersThesis	A masters thesis.	
4.14. Review	A piece of writing giving one persons opinion of the book.	
4.15. PhoneCall		
4.16. Software	A set of computer programs, procedures, and associated documentation concerned with the operation of a data processing system; e.g., compilers, library routines, manuals, and circuit diagrams.	[61]
4.17. Speech		
4.18. DocumentRepresentation		
4.18.1. PaperDocument		
4.18.2. ElectronicDocument		
4.19. Author	A person or organisation who is solely or partly responsible for the creation of a book.	

Anexo B

Código del algoritmo HORMIGA

Algoritmo HORMIGA en código C++:

```
#include "simulation.h"
#include "random.h"
#include "graph_pesos.cc"

#define ARCHIVOPESOS "PesosOnto100.txt"
#define TAM_ONTO 100

const int HORMIGAS = 10;
const int CAP_ALMACEN = 100;
const int ITERACIONES = 50;
const float RHO = 0.1;

/*Inicializa el modelo de Ant de recorrido aleatorio en profundidad*/
void AntRDFSModel::init(int _pid){
   int i,*n,*p;
   float w;
   pid = _pid;
   r.setRange(0.0,1.0);
   visitado = 0;
   padre = me;
   contador = 0;

   while (n = neighbors->next())
      sin_visitar.insert(w=0.1, i = *n);

   /*Lee y guarda los pesos de los nodos de G1*/
   GraphPesos *graficaPesos = new GraphPesos(ARCHIVOPESOS);
   vector = graficaPesos->getVectorPesos();
   vector_pesos = *vector;
};
```

```
/*Obtiene el peso de un nodo de G1, de acuerdo a su identificador*/
int AntRDFSModel::peso(int j){
   int d, *ij;

   if (ij = vector_pesos.find(j))
      return (d = *ij);
   else
      return (100000);
};

/* Ajusta los pesos para que sumen 1.0 */
void AntRDFSModel::normalize(){

   float *w, total,temp;

   total=0;
   while (w = sin_visitar.next())
      total+= *w;
   while (w = sin_visitar.next()){
      temp = *w;
      *w = temp/total;
   }
};

/*Permite continuar la exploración de los caminos sobre G1*/
void AntRDFSModel::continua_exploracion(){
   int j;
   float w;

   if (!sin_visitar.isEmpty()){
      w = (float) r.getRealVal();
      j = sin_visitar.ifind(w);
      sin_visitar.remove(j);
      normalize();
      send(DESCUBRE, j, pid);
   } else {
      if (padre != me)
         send(REGRESA, padre, pid, &particion);
      else //soy el padre {
         cout << "soy el padre\n";
         particion.inqueue(me);
         contador++;
```

```
            termina(contador);
        };
    };
};

/*Atiende el mensaje recibido por el agente actual*/
void AntRDFSModel::atiende(AntRDFSMessage *e){

    int j,c,t;
    LinkList<int> *l;
    int *p;

    j = e->getSource();
    pid = e->getPid();
    c = e->getContador();
    l = e->getRecorrido();

    switch(e->getName()) {
        /*Mensaje que recibe un nuevo nodo visitado*/
        case DESCUBRE:
            sin_visitar.remove(j);
            normalize();
            if (!visitado) {
                visitado = 1;
                padre = j;
                while (int *k = neighbors->next()) {
                    if (padre != *k)
                        send(AVISO, *k, pid);
                };
                continua_exploracion();
            };
            break;

        /*Mensaje que recibe un nodo para regresar del recorrido*/
        case REGRESA:
            if (l->getLength() != 0) {
                while( p = l->next())
                    particion.inqueue(*p);
            }
            particion.inqueue(j);
            hijos.inqueue(j);
            continua_exploracion();
            break;
```

```
      /*Mensaje para avisar a los vecinos del nodo*/
      case AVISO:
         sin_visitar.remove(j);
         normalize();
         break;

      /*Mensaje para generar una nueva lista*/
      case NEWLISTA:
         evalua_solucion();
         break;
   };
};


/*Evalua la solución para crear los grupos de nodos para el almacen*/
int AntRDFSModel::evalua_solucion(){
   int suma=0, k, numlista=1, tam=0, *p, i, *m;
   LinkList<int> lis, *aux, *aux2;
   int x=0;

   while ( m = particion.next()){
      x++;
      p = vector_pesos.find(i=*m);
      suma= suma+(*p);
      if (suma <= CAP_ALMACEN){
         lis.inqueue(*m,x);
      } else {
         grupos.insert(lis, numlista);
         //vaciar la lista
         while(!(lis.isEmpty())){
            k = lis.dequeue();
         };
         //se crea nueva lista
         suma=0;
         numlista++;
         lis.inqueue(*m);
         suma=*p;
         send(NEWLISTA, pid);
      };
   };
   //insertar ultima lista formada, me incluyo
   grupos.insert(lis, numlista);
```

```
   numlista++;
   if (me == 1)
      tam = grupos.getLength();
   else
      return tam;
};

/*Termina el recorrido de una hormiga y envía la solución al hormiguero
void AntRDFSModel::termina(int cont) {
   int tam;
   NestAntModel **maux, *m1;

   //imprime los elementos del recorrido
   while (int *n = particion.next()){
      cout << " / " << *n;
   }
   cout << "\n";

   tam = evalua_solucion();

   if (maux=(NestAntModel **) partners.find(1)) {
      m1=*maux;
      m1->ver_solucion(pid, tam);
   }
   else
      cout << "No esta NestAntModel\n";
};
```

Bibliografía

[1] Berners-Lee, T., Hendler, J., Lassila, O.: "The semantic web". *Scientific American* 2001

[2] Gandon, Fabien. "ONTOLOGY ENGINEERING: A SURVEY AND A RETURN ON EXPERIENCE", *Reporte de investigación INRIA, equipo ACACIA*. 181 p. March 2002

[3] Medina-Ramírez, Reyna Carolina. "Semantic Information Retrieval: a return on experience". *Journal of Engineering Letters, Special issue on Artificial Intelligence and Computer Science, Vol 15, issue 2. IAENG International Association of Engineers.* 2007

[4] Dejan S. Milojicic, Vana Kalogeraki, Rajan Lukose, Kiran Nagaraja, Jim Pruyne, Bruno Richard, Sami Rollins and Zhichen Xu. (2002). "Peer-to-peer computing". *Hewlett Packard. Technical Report.* HPL-2002-57R1.

[5] Bonifacio, M.; Bouquet, P.; Traverso, P. "Enabling Distributed Knowledge Management. Managerial and Technological Implications". *Novatica and Informatik/Informatique*, vol. III, n. 1 (2002).

[6] Bonifacio, M.; Bouquet, P.; Traverso, P. "Knowledge Management Knowledge Management Enabling Distributed Knowledge Management. Managerial and Technological Implications". *Novatica and Informatik/Informatique*, vol. III, n. 1 (2002).

[7] Risson, J, Moors, T. "Survey of research towards robust peer-to-peer networks: search methods", *Computer Networks: The International Journal of Computer and Telecommunications Networking*, Volume 50 , Issue 17, pp 3485-3521, 2006, ISSN:1389-1286

[8] Latent Semantic Indexing.
`http://www.hirank.com/semantic-indexing-project/lsi/`, 2007

[9] Tang, C., Z. Xu, and M. Mahalingam.: PeerSearch: Efficient Information retrieval in Peer-Peer Networks. Hewlett-Packard Labs: Palo Alto.(2002)

[10] Hassan, R. Anwar, Z. Yurcik, W. Brumbaugh, L. Campbell, R.. "A Survey of Peer-to-Peer Storage Techniques for Distributed File Systems". *In Proceedings of IEEE International Conference on Information Technology (ITCC)*, April 2005.

[11] A. Y. Halevy, Z. G. Ives, P. Mork, and I. Tatarinov. "Piazza: Data management infrastructure for semantic web applications". *In Proceedings of the Twelfth International World Wide Web Conference* (WWW2003), Budapest, Hungary, May 2003.

[12] Nejdl, W. Wolf,B. Qu, C. Decker, S. Sintek, M. Naeve, A. Nilsson, M. Palmer, M. Risch, T. "EDUTELLA: a P2P Networking Infrastructure based on RDF". *In Proceedings of the 11th International World Wide Web Conference*, Hawaii, USA, May 2002. `http://edutella.jxta.org/reports/edutellawhitepaper.pdf`

[13] Cai, M., Frank, M., "RDFPeers: A Scalable Distributed RDF Repository Based on a Structured Peer-to-Peer Network". *In Proceedings of the 13th international conference on the World Wide Web* (New York, USA, May 2004), 650-657.

[14] Owens Alisdair, "Semantic Storage: Overview and Assessment". Technical Report IRP Report, Electronics and Computer Science, U of Southampton, 2005

[15] Kjetil, N., Christos, D., Michalis, V.: The SOWES Approach to P2P Web Search Using Semantic Overlays. WWW '06: Proceedings of the 15th international conference on World Wide Web, (2006) pp.1027-1028

[16] Crespo, A. Garcia-Molina, H.: Semantic Overlay Networks for P2P Systems. Technical report, Stanford University, (2002)

[17] Medina-Ramírez, Reyna Carolina. "Contribution a la recherche d'informations sémantiques : Capitalisation de connaissances dans une mémoire d'interactions géniques". Tesis doctoral, INRIA Sophia Antipolis - Université de Nice-Sophia Antipolis, France.

[18] Medina-Ramírez, Reyna Carolina, Corby, Olivier y Dieng-Kuntz, Rose. "A Conceptual Graph and RDF(S) approach for representing and querying document content". In Francisco J. Garijo, José Cristóbal Riquelme Santos, Miguel Toro *(Eds.): Advances in Artificial Intelligence - IBERAMIA 2002, 8th Ibero-American Conference on AI*, Seville, Spain, November 12-15, 2002. Pp: 121-130. Lecture Notes in Computer Science (LNCS) 2527. Springer-Verlag. ISBN 3-540-00131-X.

[19] Gruber,T. "A translation approach to portable ontology specifications", *Proc. of JKAW' 92*, pp. 89-108, 1992.

[20] Studer R, Benjamins VR, Fensel D. "Knowledge Engineering: Principles and Methods". *IEEE Transactions on Data and Knowledge Engineering* 1998

[21] Consorcio W3C, `http://www.w3c.es/Traducciones/es/SW/2005/owlfaq`, 2008

[22] M. Uschold and M. King. "Towards a Methodology for Building Ontologies". *In Workshop on Basic Ontological Issues in Knowledge Sharing, held in conjunction with IJCAI-95*, Montreal, Canada, 1995

[23] Corcho, Oscar, Fernández-López Mariano, Gómez-Pérez Asunción. "Methodologies, tools and languages for building ontologies. where is their meeting point?" *Data and Knowledge Engineering*, vol. 46, no. 1. 2003

[24] Mizoguchi, Riichiro. "Towards Ontology Engineering" *J. Jpn. Soc. for Artificial intelligence*, Vol. 13,No. 1, pp. 9-10, 1998

[25] M. Fernandez, A. Gomez-Perez, and N. Juristo. "METHONTOLOGY: From Ontological Arts Towards Ontological Engineering". *In Proceedings of the AAAI97 Spring Symposium Series on Ontological Engineering*, Stanford, USA, pages 33–40, March 1997.

[26] KBSI "The IDEF5 Ontology Description Capture Method Overview", KBSI Report, Texas, 1994

[27] Taxonomía ACM **http://www.acm.org/class/1998/overview.html**, Septiembre 2007

[28] Diccionario de redes **http://www.networkdictionary.com/**, 2007

[29] Stallings, William. "Comunicaciones y Redes de Computadoras". 6ta. Edición. España. 2002

[30] Halsall, Fred. "Data Communications", Computer Networks and Open Systems", 4th Edition. Addison-Wesley. USA. 1996. 907 p.

[31] Tesauro de Redes de Computadoras **http://www.um.es/gtiweb/fjmm/tesauro/**, Septiembre 2007

[32] Vocabulario de Infraestructura de Telecomunicaciones **http://www.multites.com/**, Septiembre 2007

[33] Ontología de Telecomunicaciones
http://meag.tele.pw.edu.pl/sims/sims-ontology/index.html, Agosto 2007

[34] Ontología de Documento
http://www.cs.umd.edu/projects/plus/SHOE/onts/docmnt1.0.html, 4 de Noviembre de 2007

[35] Preguntas frecuentes sobre el Lenguaje de Ontologías Web (OWL) del W3C **http://www.w3c.es/Traducciones/es/SW/2005/owlfaq**, 2007

[36] OWL Web Ontological Language Overview **http://www.w3.org/TR/owl-features/**, 2007

[37] Resource Description Framework (RDF) **http://www.w3.org/RDF/**, 2007

[38] K. Wilkinson, C. Sayers, H. Kuno, D. Reynolds, "Efficient RDF Storage and Retrieval in Jena2", HP Laboratories Technical Report HPL-2003-266

[39] Sesame: A Generic Architecture for Storing and Querying RDF and RDF Schema. **http://sesame.aidministrator.nl/publications/del10.pdf**, 2007

[40] Protege:Editor de ontologías **http://protege.stanford.edu/**, 2007

[41] Sewese **http://www-sop.inria.fr/acacia/soft/sewese/**, 2007

[42] Corese **http://www-sop.inria.fr/edelweiss/wiki/wakka.php?wiki=Corese**, 2007

[43] Apache Tomcat **http://tomcat.apache.org/**, 2007

[44] SPARQL Query Language for RDF, http://www.w3.org/TR/rdf-sparql-query/, 2007

[45] Marcelín Jiménez, R. Almacenamiento distribuido tolerante a fallas. Tesis de doctorado, UNAM, 2004.

[46] Marcelín-Jiménez, Ricardo, S. Rajsbaum and B. Stevens. "Cyclic Storage for Fault-tolerant Distributed Executions". *IEEE Transactions on Parallel and Distributed Systems*, Vol. 17, No. 9, 2006, pp. 1028-1036

[47] Marcelin-Jiménez, Ricardo. "Distributed Site Partitioning", *10th International Colloquium on Structural Information and Communication Complexity (SIROCCO 10), Jop F. Sibeyn (ed.)*, pp.249-257. 2003

[48] Marcelin-Jiménez, Ricardo. "A Flexible Simulator for Distributed Algorithms", Proceedings of the ENC'03 by the IEEE Computer Society Press, (2003) pp.176-181

[49] Quezada-Naquid, Moisés, Marcelín-Jiménez, Ricardo, López-Guerrero, Miguel. "Service Policies for Storage Service Dispatcher in a Distributed Fault-Tolerant Storage Network and their Performance Evaluation". *In Proceedings of the 20th IEEE Canadian Conference in Electrical and Computer Engineering CCECE'07*, pp231-234, Vancouver, Canada, Abril 22-26, 2007.

[50] Reeves, Colin R. "Modern Heuristic Techniques for Combinatorial Problems". McGrawHill Uk. 1995. 320 p.

[51] Horowitz, E.; Sahni, S. "Fundamentals of Computer Algorithms"; Computer Science Press; 1978

[52] Montresor, A.: Anthill: a Framework for the Design and the Analysis of Peer-to-Peer Systems. 4th European Research Seminar on Advances in Distributed Systems. (2001)

[53] W. Gutjahr. "A generalized convergence result for the graph-based ant system metaheuristic". Technical Report 99-09, University of Vienna, 1999.

[54] A. S. Tanenbaum. "Redes de Computadoras". Prentice Hall Hispanoamericana S. A., México, 1997.

[55] Freer, John. "Computer Communications and Networks", Taylor & Francis, 1996, 394 p.

[56] J. Kurose and K. Ross, "Computer Networking: A Top-Down Approach Featuring the Internet", Addison-Wesley, 2nd edition, July 2002.

[57] International Telecommunication Union, `www.itu.ch`, 2007

[58] Coulouris G., Dollimore J., Kindberg, "Distributed systems: Concepts and Design", 4ta. Edición, Addison-Wesley, USA, 2005

[59] Joel M. Crichlow, "An Introduction to distributed and parallel computing", 2da. Edición, Prentice Hall Europe, 1997

[60] Balakrishnan, Kaashoek, Karger,Morris, Stoica. "Looking up data in P2P systems". In COMMUNICATIONS OF THE ACM February 2003/Vol. 46, No. 2. February 2003

[61] Alliance for Telecommunications Industry Solutions `http://www.atis.org/glossary/`, Octubre 2007

[62] Vocabulario de Documento
`http://www.e.govt.nz/archive/standards/nzgls/standard/usage-guide-2-1/chapter33.html`, Noviembre de 2007

[63] Ontología de Documento
`http://knowledgeweb.semanticweb.org/semanticportal/OWL/DocumentationOntology.owl`, Novimebre de 2007