



Universidad de San Carlos de Guatemala

Facultad de Ingeniería

Escuela de Ingeniería en Ciencias y Sistemas

**DESARROLLO DE UNA APLICACIÓN PEER-TO-PEER ORIENTADA A
PEQUEÑAS Y MEDIANAS EMPRESAS (PYMES) PARA LA OFERTA Y
DEMANDA DE PRODUCTOS Y SERVICIOS.**

Raúl del Cid López

Asesorado por el Ing. Freiry Javier Gramajo López

Guatemala, octubre de 2009

UNIVERSIDAD DE SAN CARLOS DE GUATEMALA



FACULTAD DE INGENIERÍA

**DESARROLLO DE UNA APLICACIÓN PEER-TO-PEER ORIENTADA A
PEQUEÑAS Y MEDIANAS EMPRESAS (PYMES), PARA LA OFERTA Y
DEMANDA DE PRODUCTOS Y SERVICIOS.**

TRABAJO DE GRADUACIÓN

PRESENTADO A LA JUNTA DIRECTIVA DE LA

FACULTAD DE INGENIERÍA

POR:

RAÚL DEL CID LÓPEZ

ASESORADO POR EL ING. FREIRY JAVIER GRAMAJO LOPEZ

AL CONFERÍRSELE EL TÍTULO DE

INGENIERO EN CIENCIAS Y SISTEMAS

GUATEMALA, OCTUBRE DE 2009

UNIVERSIDAD DE SAN CARLOS DE GUATEMALA
FACULTAD DE INGENIERIA



NÓMINA DE JUNTA DIRECTIVA

DECANO	Ing. Murphy Olympo Paiz Recinos
VOCAL I	Inga. Glenda Patricia García Soria
VOCAL II	Inga. Alba Maritza Guerrero de López
VOCAL III	Ing. Miguel Ángel Dávila Calderón
VOCAL IV	Br. José Milton De León Bran
VOCAL V	Br. Isaac Sultán Mejía
SECRETARIA	Inga. Marcia Ivónne Véliz Vargas

TRIBUNAL QUE PRACTICÓ EL EXAMEN GENERAL PRIVADO

DECANO	Ing. Murphy Olympo Paiz Recinos
EXAMINADOR	Ing. Alvarado Díaz Ardavín
EXAMINADOR	Ing. Pedro Pablo Hernández
EXAMINADOR	Ing. Oscar Alejandro Paz Campos
SECRETARIA	Inga. Marcia Ivónne Véliz Vargas

HONORABLE TRIBUNAL EXAMINADOR

Cumpliendo con los preceptos que establece la ley de la Universidad de San Carlos de Guatemala, presento a su consideración mi trabajo de graduación titulado:

DESARROLLO DE UNA APLICACIÓN PEER-TO-PEER ORIENTADA A PEQUEÑAS Y MEDIANAS EMPRESAS (PYMES), PARA LA OFERTA Y DEMANDA DE PRODUCTOS Y SERVICIOS,

tema que me fuera asignado por la Dirección de la Escuela de Ingeniería en Ciencias y Sistemas, en agosto 2008.

Raúl del Cid López

Guatemala, 16 de julio de 2009

Ingeniero
Carlos Alfredo Azurdia Morales
Coordinador
Comisión de Aprobación y Revisión de Tesis
Escuela de Ciencias y Sistemas
Facultad de Ingeniería
Universidad de San Carlos de Guatemala

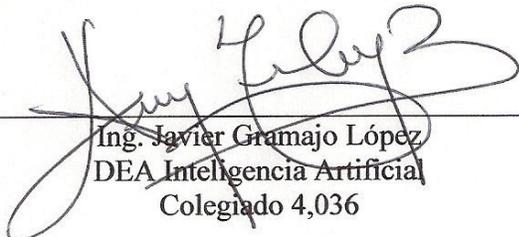
Estimado Ingeniero:

Por medio de la presente hago de su conocimiento que he considerado el trabajo de tesis titulado **Desarrollo de una aplicación Peer-to-Peer orientada a Pequeñas y Medianas Empresas (PYMES) para la oferta y demanda de productos y servicios**, que el estudiante Raúl del Cid López ha desarrollado como proyecto de graduación para optar al título de Ingeniero en Ciencias y Sistemas. Este trabajo ha sido de mi interés, por la experiencia con que cuento en el área de investigación que el estudiante ha elegido, y a mi consideración el contenido del trabajo es satisfactorio y está finalizado en un 100%.

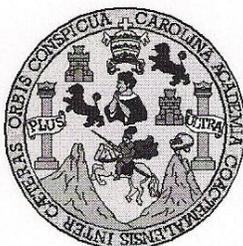
Como Asesor de este trabajo me hago responsable de la calidad y del contenido que la tesis ha alcanzado.

Sin otro particular y agradeciendo la oportunidad de colaboración a la educación e investigación universitaria que me da, me despido.

Atentamente,



Ing. Javier Gramajo López
DEA Inteligencia Artificial
Colegiado 4,036



Universidad San Carlos de Guatemala
Facultad de Ingeniería
Escuela de Ingeniería en Ciencias y Sistemas

Guatemala, 12 de Agosto de 2009

Ingeniero
Marlon Antonio Pérez Turk
Director de la Escuela de Ingeniería
En Ciencias y Sistemas

Respetable Ingeniero Pérez:

Por este medio hago de su conocimiento que he revisado el trabajo de graduación del estudiante **RAÚL DE CID LÓPEZ**, titulado: **“DESARROLLO DE UNA APLICACIÓN PEER-TO-PEER ORIENTADA A PEQUEÑAS Y MEDIANAS EMPRESAS (PYMES) PARA LA OFERTA Y DEMANDA DE PRODUCTOS Y SERVICIOS”**, y a mi criterio el mismo cumple con los objetivos propuestos para su desarrollo, según el protocolo.

Al agradecer su atención a la presente, aprovecho la oportunidad para suscribirme,

Atentamente,


Ing. Carlos Alfredo Azurdia
Coordinador de Privados
y Revisión de Trabajos de Graduación



E
S
C
U
E
L
A

D
E

C
I
E
N
C
I
A
S

Y

S
I
S
T
E
M
A
S

UNIVERSIDAD DE SAN CARLOS
DE GUATEMALA



FACULTAD DE INGENIERÍA
ESCUELA DE CIENCIAS Y SISTEMAS
TEL: 24767644

*El Director de la Escuela de Ingeniería en Ciencias y Sistemas de la Facultad de Ingeniería de la Universidad de San Carlos de Guatemala, luego de conocer el dictamen del asesor con el visto bueno del revisor y del Licenciado en Letras, de trabajo de graduación titulado **“DESARROLLO DE UNA APLICACIÓN PEER-TO-PEER ORIENTADA A PEQUEÑAS Y MEDIANAS EMPRESAS (PYMES) PARA LA OFERTA Y DEMANDA DE PRODUCTOS Y SERVICIOS”**, presentado por el estudiante RAÚL DEL CID LÓPEZ, aprueba el presente trabajo y solicita la autorización del mismo.*

“ID Y ENSEÑAD A TODOS”



Ing. Marlon Antonio Perez Turk
Director, Escuela de Ingeniería Ciencias y Sistemas

Guatemala, 19 de octubre 2009

Universidad de San Carlos
de Guatemala



Facultad de Ingeniería
Decanato

Ref. DTG.422.09

El Decano de la Facultad de Ingeniería de la Universidad de San Carlos de Guatemala, luego de conocer la aprobación por parte del Director de la Escuela de Ingeniería en Ciencias y Sistemas, al trabajo de graduación titulado: **DESARROLLO DE UNA APLICACIÓN PEER-TO-PEER ORIENTADA A PEQUEÑAS Y MEDIANAS EMPRESAS (PYMES), PARA LA OFERTA Y DEMANDA DE PRODUCTOS Y SERVICIOS**, presentado por el estudiante universitario Raúl Del Cid López, procede a la autorización para la impresión del mismo.

IMPRÍMASE.


Ing. Murphy Olimpo Paiz Recinos
DECANO

Guatemala, octubre de 2009



/cc

ACTO QUE DEDICO A:

Dios

Por acompañarme en todos los momentos de mi vida y rodearme de las personas indicadas.

Mi madre

Telma López Perdomo, por todo su amor y sacrificio. Con su ejemplo me ha hecho el hombre que soy, este logro es especialmente para ella.

Mi hijo

Juan Pablo, por enseñarme que la grandeza de la vida está en los detalles más pequeños.

Mi esposa

Ana Lucrecia, por todo su amor y apoyo, gracias por ser mi aliento y ayudarme a alcanzar este logro tan importante.

Mi abuelita

Erica Perdomo, por todo su cariño, sus regaños y por siempre estar pendiente de mí.

Mis primos

Daniel, Damian y Danilo López, por los momentos que compartimos juntos y formar parte de mi vida.

Mi tío	Aroldo López, por sus buenos consejos, su amistad y por motivarme a terminar mi carrera.
Familia Castillo Coronado	Por aceptarme como miembro de su familia y apoyarme en los últimos años.
Familia Santizo Castillo	Por los momentos compartidos, por su amistad y por motivarme siempre a seguir adelante.
Mi asesor	Javier Gramajo, por contagiarme con su entusiasmo y brindarme su amistad y apoyo durante estos últimos años.
Universidad San Carlos de Guatemala	Por haberme exigido y a la vez enseñado tanto en las aulas durante los años de estudio.

ÍNDICE GENERAL

ÍNDICE DE ILUSTRACIONES.....	VII
GLOSARIO.....	XI
RESUMEN.....	XV
OBJETIVOS	XVII
INTRODUCCIÓN	XIX
1. LOS NEGOCIOS ELECTRÓNICOS	1
1.1 Introducción.....	1
1.2 Los negocios electrónicos y el comercio electrónico.....	3
1.2.1 ¿Qué son los negocios electrónicos?	3
1.2.2 ¿Qué es comercio electrónico?	4
1.3 Tipos de negocios electrónicos	5
1.3.1 <i>B2C</i>	6
1.3.2 <i>B2B</i>	7
1.3.3 <i>B2G</i>	8
1.3.4 <i>P2P</i>	8
1.4 Modelos de negocio	9
1.4.1 Elementos de un modelo de negocio.....	9
1.5 Usos de los negocios electrónicos	12
1.5.1 Lidiar con el Gobierno.....	13
1.5.2 Correo electrónico.....	14
1.5.3 Banca en línea.....	16
1.5.4 Compras en línea	18
1.5.5 Investigación de mercados	20
1.6 Beneficios.....	21

1.6.1	Administración	21
1.6.2	Producción, intercambio y distribución	22
1.6.3	Recursos humanos.....	23
1.6.4	Servicio al cliente.....	23
1.6.5	Mercadeo y comunicaciones	24
1.6.6	Colaboración	25
1.6.7	Investigación de mercados.....	26
2.	LA WEB SEMÁNTICA	27
2.1	Introducción	27
2.1.1	Visión.....	28
2.2	La Web Semántica en los negocios	29
2.2.1	Soporte a la toma de decisiones	30
2.2.2	Desarrollo de negocios.....	31
2.2.3	Compartir y descubrir información.....	31
2.2.4	Administración y automatización	32
2.3	Tecnologías	33
2.3.1	<i>URI</i>	33
2.3.2	<i>XML</i> y <i>XML Schema</i>	34
2.3.3	<i>RDF</i> y <i>RDF Schema</i>	35
2.4	Las Ontologías.....	36
2.4.1	Estructura	38
2.4.2	¿Para qué crear una Ontología?	40
2.4.3	Tipos de Ontologías	42
2.4.4	Aplicaciones comunes.....	43
2.4.5	Las Ontologías en la Web	44
2.5	Lenguajes para representar Ontologías.....	52
2.5.1	Lenguajes tradicionales.....	52
2.5.2	Lenguajes basados en la Web	55

2.6 Herramientas.....	59
2.6.1 <i>Protégé</i>	59
2.6.2 <i>OILEd</i>	61
2.6.3 <i>OntoEdit</i>	62
2.7 Pasos para crear una Ontología.....	63
2.7.1 Determinar el dominio y alcance de la Ontología	64
2.7.2 Reutilizar Ontologías existentes	65
2.7.3 Enumerar los términos de la Ontología.....	66
2.7.4 Definir las clases y su jerarquía	66
2.7.5 Definir las propiedades de las clases (slots).....	67
2.7.6 Definir las facetas de los slots	67
2.7.7 Crear instancias.....	67
2.8 Aplicaciones semánticas actuales.....	68
2.8.1 <i>Freebase</i>	69
2.8.2 <i>Powerset</i>	70
2.8.3 <i>Twine</i>	71
2.8.4 <i>Hakia</i>	72
2.8.5 <i>AdaptiveBlue</i>	73
2.8.6 <i>TrueKnowledge</i>	74
2.8.7 <i>Triplt</i>	74
2.8.8 <i>Spock</i>	76
3. LAS REDES PEER TO PEER.....	77
3.1 Introducción.....	77
3.2 Elementos	80
3.2.1 Nodos	81
3.2.2 Grupos de nodos	83
3.2.3 Transporte de red	84
3.2.4 Servicios	85

3.2.5	Anuncios.....	85
3.2.6	Protocolos	86
3.2.7	Identificación	86
3.3	Objetivos del <i>P2P</i>	87
3.3.1	Reducir y compartir costos	87
3.3.2	Agregar recursos (mejorar rendimiento).....	87
3.3.3	Mejorar escalabilidad y confiabilidad	88
3.3.4	Incrementar la autonomía.....	88
3.3.5	Mejorar la privacidad y anonimato.....	88
3.3.6	Facilitar el dinamismo.....	89
3.3.7	Habilitar comunicaciones <i>ad-hoc</i>	89
3.4	Clasificación del <i>P2P</i>	90
3.4.1	Puro.....	91
3.4.2	Híbrido.....	92
3.4.3	Super nodos	93
3.5	Aplicaciones del <i>P2P</i>	94
3.5.1	Paralelas	94
3.5.2	Gestión de contenido.....	95
3.5.3	Colaborativas.....	95
3.6	El <i>P2P</i> y los negocios	96
3.7	Características	97
3.7.1	Descentralización	97
3.7.2	Escalabilidad	97
3.7.3	Anonimato	98
3.7.4	Organización autónoma	101
3.7.5	Costo compartido	102
3.7.6	Conectividad ad-hoc.....	102
3.7.7	Rendimiento	103
3.7.8	Seguridad	105

3.7.9	Transparencia	106
3.7.10	Tolerancia a fallos	106
3.7.11	Interoperabilidad	107
4.	LA PLATAFORMA JXTA	109
4.1	Introducción	109
4.2	Capas lógicas	110
4.2.1	Capa base	110
4.2.2	Capa de servicios	111
4.2.3	Capa de aplicaciones	111
4.3	Protocolos	112
4.3.1	<i>Peer Discovery</i>	113
4.3.2	<i>Peer Resolver</i>	116
4.3.3	<i>Rendezvous</i>	118
4.3.4	<i>Peer Information</i>	121
4.3.5	<i>Pipe Binding</i>	122
4.3.6	<i>Endpoint Routing</i>	124
4.4	Ventajas y desventajas	127
4.4.1	Ventajas	127
4.4.2	Desventajas	127
4.5	Comparación con <i>Jini</i> y <i>.NET</i>	128
5.	IMPLEMENTACIÓN DE LA APLICACIÓN P2P ORIENTADA A PYMES, PARA LA OFERTA Y DEMANDA DE PRODUCTOS Y SERVICIOS	129
5.1	Introducción	129
5.2	Definición del problema	129
5.3	Análisis	130
5.3.1	Objetivos	130
5.3.2	Beneficios	131

5.3.3 Funciones de la aplicación cliente	132
5.3.4 Funciones de la aplicación servidor.....	134
5.4 Diseño.....	136
5.4.1 Entorno.....	137
5.4.1 Herramientas de desarrollo	138
5.4.2 Elementos usados de <i>JXTA</i>	139
CONCLUSIONES.....	141
RECOMENDACIONES	143
REFERENCIAS	145
BIBLIOGRAFÍA.....	149
APÉNDICE	151

ÍNDICE DE ILUSTRACIONES

FIGURAS

1. Crecimiento de Internet, medido por el número de Hosts en el mundo.....	2
2. Crecimiento del comercio electrónico <i>B2B</i> en Estados Unidos de América ..	7
3. Uso de la Web Semántica en la Organización	30
4. Tecnologías que forman la Web Semántica.....	33
5. Partes que forman el <i>RDF</i>	35
6. Ejemplo de una Ontología sobre una Empresa.....	37
7. Portal Web mejorado mediante el uso de Ontologías	45
8. Funcionamiento de un Agente Web de Servicios.....	50
9. Estructura en capas del lenguaje <i>OIL</i>	57
10. Interfaz gráfica de la herramienta <i>Protégé</i>	60
11. Pantalla principal de la herramienta <i>Oiled v3.5.7</i>	62
12. Ontología de productos y servicios visualizada en <i>OntoEdit</i>	63
13. Formas de implementar la Web Semántica.....	68
14. Página principal de la base de datos <i>Freebase</i>	69
15. Ejemplo de una búsqueda realizada en <i>Powerset</i>	70
16. Página de introducción a la aplicación <i>Twine</i>	71
17. Pantalla principal del motor de búsqueda <i>hakia</i>	72
18. Ejemplo de un hipervínculo mejorado con <i>SmartLinks</i>	73
19. Encabezado del sitio Web oficial de <i>TrueKnowledge</i>	74
20. Pasos básicos para utilizar el agente <i>Triplt</i>	75
21. Versión beta del motor de búsqueda <i>Spock</i>	76
22. Comparación de la computación cliente/servidor y el <i>P2P</i>	80

23. Clasificación de los sistemas de cómputo.....	90
24. Funcionamiento del modelo <i>P2P</i> Puro.....	91
25. Modelo de directorio centralizado	92
26. Funcionamiento del modelo <i>P2P super nodos</i>	93
27. Clasificación de aplicaciones <i>P2P</i>	94
28. Capas lógicas de la plataforma <i>JXTA</i>	110
29. Protocolos de la plataforma <i>JXTA</i>	112
30. Funcionamiento del protocolo <i>Peer Discovery</i>	113
31. Interacción de mensajes <i>rendezvous</i>	119
32. Funcionamiento del protocolo <i>Peer Information</i>	122
33. Funcionamiento del protocolo <i>Pipe Binding</i>	123
34. Funcionamiento del protocolo <i>Endpoint Routing</i>	125
35. Componentes del sistema <i>OntoBiz</i>	136
36. Diagrama de la red virtual de <i>OntoBiz</i>	137
37. Pantalla del instalador web de <i>Ontobiz</i>	152
38. Pantalla inicial del instalador <i>Ontobiz</i>	153
39. Pantalla de introducción al instalador <i>Ontobiz</i>	153
40. Pantalla de aceptación de la licencia.	154
41. Pantalla de registro de la empresa.....	155
42. Navegador de actividades económicas.....	156
43. Pantalla principal de la aplicación <i>Ontobiz</i>	157
44. Pantalla para administración de ofertas.	158
45. Pantalla para realizar búsquedas.....	159
46. Pantalla de detalles de oferta.....	160
47. Dialogo de confirmación de compra.....	160
48. Pantalla de visualización de compras.	161
49. Pantalla de visualización de ventas.	162
50. Pantalla de visualización detallada de ventas.	163
51. Estructura en paquetes del código fuente.....	164

TABLAS

I. Estadísticas de población y uso de Internet en Centro América.....	3
II. Modelos de ganancia en los negocios electrónicos.....	10
III. Sitios Web relacionados con el Gobierno de Guatemala.....	13
IV. Bancos guatemaltecos que ofrecen servicios en línea.....	16
V. Tipos de Ontologías	42
VI. Listado de Ontologías reutilizables disponibles en la Web.....	65
VII. Partes de un transporte de red.....	84
VIII. Parámetros del mensaje <i>Discovery Query</i>	114
IX. Parámetros del mensaje <i>Discovery Response</i>	115
X. Beneficios de la aplicación <i>OntoBiz</i>	131
XI. Herramientas usadas para el desarrollo de la aplicación <i>P2P</i>	138

GLOSARIO

- Ad hoc** Palabra latina que significa “para este propósito”. Generalmente significa una solución que fue diseñada para un problema específico y que no se puede generalizar ni adaptar para otro propósito.
En las redes de computadoras, *ad hoc* es una conexión de red asociada a dispositivos inalámbricos. La conexión se establece para la duración de una sesión y no requiere una estación central.
- Cliente** Formalmente definido como una entidad, ya sea un nodo, un programa o un módulo, que inicia peticiones pero que no es capaz de atenderlas.
- Cliente/servidor** Representa la ejecución de entidades que poseen los roles de cliente y servidor al mismo tiempo.
- Computación distribuida** Un sistema de computadora en el cual varias computadoras interconectadas comparten la tarea de computación asignada al sistema.
- Computación Grid** Es una infraestructura para compartir globalmente recursos intensivos de cómputo, como las supercomputadoras o clusters.

Dominio	Un área de temática específica o un área de conocimiento, tales como medicina, fabricación de herramientas, bienes inmuebles, reparación automovilística, gestión financiera, etc.
Esquema	Es una descripción genérica de una colección de metadatos.
Lógica descriptiva	Es una familia de lenguajes de representación del conocimiento que pueden ser usadas para representar el dominio de una aplicación de una forma estructurada y con una comprensión bien establecida.
Modelo P2P	Modelo de computación distribuido en el cual todas las computadoras que conforman la red pueden compartir sus recursos, y cada una puede jugar los papeles de cliente y servidor.
Peer	Se define como una entidad con capacidades similares a las demás entidades del sistema.
Servidor	Se define como una entidad que atiende las peticiones de otras entidades, pero no las inicia.
Sistema centralizado	Representa soluciones de una sola unidad, incluyendo máquinas de procesador simple/múltiple, como supercomputadoras o mainframes.

Sistema distribuido	Es el sistema en el que sus componentes localizados en computadoras de la red, logran comunicar y coordinar sus acciones a través del paso de mensajes.
Taxonomía	Es una colección de términos explícitos y organizados en una estructura jerárquica. Cada término en la taxonomía tiene una o más relaciones de padre-hijo con otros términos.
Ubicuo	Que se encuentra al mismo tiempo en todas partes.

RESUMEN

El presente trabajo presenta el desarrollo de una aplicación de software que utiliza el modelo *Peer-to-Peer* para facilitar negocios electrónicos entre Pequeñas y Medianas Empresas (PYMES), para la oferta y demanda de sus productos y servicios.

El primer capítulo aborda el tema de los negocios electrónicos, se describen sus conceptos básicos, en que se diferencia con el comercio electrónico, sus aplicaciones, sus ventajas y desventajas.

En el segundo capítulo se hace una introducción a la Web Semántica, se describe como se logra integrar ésta con los negocios, se describen sus elementos, sus aplicaciones, la importancia de las Ontologías y herramientas para crear Ontologías.

En el capítulo tres, se introduce al lector a las redes *Peer-to-Peer*, se da una breve historia de cómo se popularizaron gracias al programa Napster, se describen los elementos de este modelo y las aplicaciones potenciales de esta tecnología.

En el capítulo cuatro, se describe la plataforma JXTA, esta es la tecnología elegida para el desarrollo de la aplicación *Peer-to-Peer*. Se describen sus elementos y la funcionalidad que permite conectar computadoras en redes *Peer-to-Peer*.

Finalmente en el capítulo cinco, se hace un análisis, diseño e implementación de la aplicación *Peer-to-Peer*, utilizando los conceptos y tecnologías descritas en los capítulos anteriores.

OBJETIVOS

General:

Desarrollar una aplicación *Peer-to-Peer* para facilitar negocios electrónicos entre Pequeñas y Medianas Empresas (PYMES), haciendo uso de una ontología que permita clasificar los productos y servicios que ofrecen.

Específicos:

1. Aplicar el comercio electrónico orientado a negocios.
2. Utilización de una ontología para clasificar los productos y servicios de las PYMES.
3. Identificar los tipos de redes *Peer-to-Peer* que se utilizan actualmente en Internet.
4. Investigar los protocolos y herramientas de uso reciente para el desarrollo de aplicaciones *Peer-to-Peer*.
5. Desarrollar una aplicación *Peer-to-Peer* que permita facilitar negocios electrónicos entre PYMES, usando una ontología para clasificar los productos y servicios.

INTRODUCCIÓN

El continuo avance de las Tecnologías de Información y Comunicación (TICS), están cambiando constantemente la forma en que las personas desarrollan sus actividades diarias, tanto dentro de la oficina como en el hogar, esto se ha dado en parte a los nuevos desarrollos tecnológicos que permiten a las empresas reducir sus costos y a la gran competencia a la que se deben enfrentar las empresas hoy en día.

Las empresas han adoptado la tecnología de la computación y la han incorporado dentro de sus procesos de negocio con el objetivo de optimizar sus recursos, incrementar su rendimiento, agilizar sus procesos, y han descubierto que esto produce más ganancias, al mismo tiempo que mejoran la relación con sus socios, con sus proveedores y con sus clientes.

Este trabajo describe el desarrollo de una aplicación de software que permitirá a Pequeñas y Medianas Empresas (PYMES) ofertar sus productos y servicios a través de Internet utilizando la tecnología *Peer-to-Peer* (P2P) y la Web Semántica. La tecnología P2P fue controversial, porque en el pasado se ha utilizado para compartir material intelectual entre sus usuarios. Sin embargo, es tan fascinante que en conjunto con la Web Semántica, permitirán el desarrollo futuro de una nueva generación de aplicaciones y servicios inteligentes que permitirán facilitar la vida de las personas y organizaciones alrededor del mundo al integrar capacidades automatizadas que de otra forma no serían posibles.

1. LOS NEGOCIOS ELECTRÓNICOS

1.1 Introducción

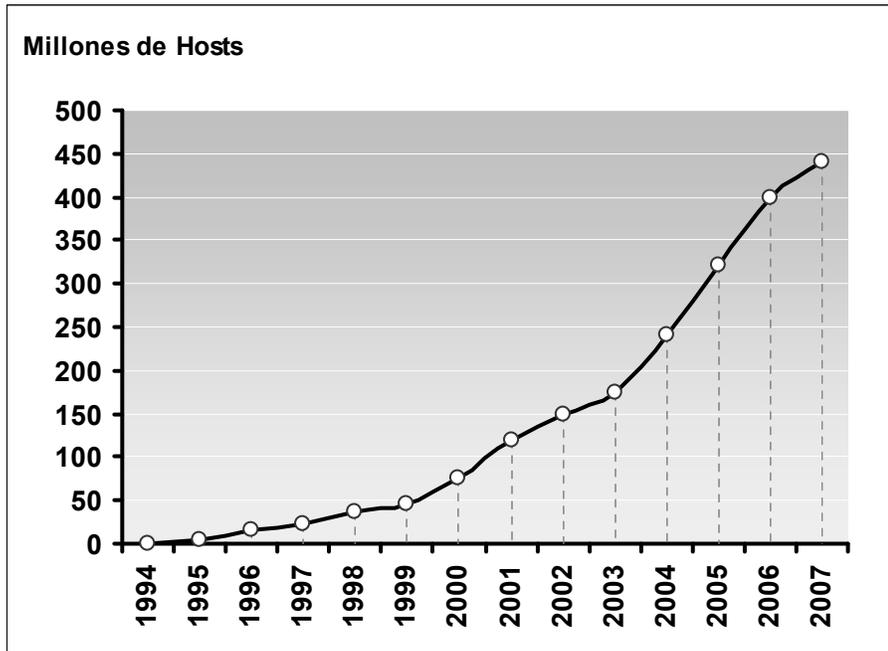
El crecimiento de Internet y la continua reducción del costo del poder de procesamiento de las computadoras, tienen un impacto profundo en la forma en que se realizan negocios alrededor del mundo. Además de su importancia como fuente de información y medio de comunicación, Internet se ha convertido en una herramienta práctica y vital para los negocios.

El término negocios electrónicos, también conocido como *eBusiness*, “se refiere ampliamente al uso de tecnologías de información y comunicación (TICs) para manejar un negocio de una forma más efectiva” [1]. Para un negocio, esto puede significar desde sistematizar cuentas de clientes o instalar Internet para el personal, hasta proveer a los clientes con acceso a los productos y servicios a través de un sitio Web. También puede significar el uso de correo electrónico para mejorar las comunicaciones con los socios, proveedores y clientes.

Los sistemas electrónicos, Internet y las tecnologías relacionadas pueden cambiar la forma en que se desarrollan y conducen los procesos de un negocio, haciéndolos más eficientes en el manejo de costos y tiempos.

Las Pequeñas y Medianas Empresas (PYMES), son las principales beneficiarias de lo que la tecnología puede ofrecer, liberando a los negocios de algunas limitantes como la ubicación y tamaño.

Figura 1. Crecimiento de Internet, medido por el número de *Hosts* en el mundo



Fuente: *Internet Systems Consortium*. www.isc.org

Como se puede observar en la figura 1, el número de huéspedes de páginas Web en Internet alrededor del mundo ha crecido exponencialmente.

En la Tabla I, se muestra como Guatemala es el segundo país de Centro América con mayor número de usuarios en Internet, cuyo total asciende a un millón trescientos veinte mil, éste crecimiento del año 2000 al 2008 fue en total de 1,930.8%. El porcentaje de penetración de Internet es actualmente de 10.2%, lo cual indica la importancia que Internet fue adquiriendo en la vida diaria de los Guatemaltecos. En comparación Costa Rica posee el mayor porcentaje de penetración de Internet de toda Centro América, mientras que Nicaragua es el país más rezagado en adoptar la nueva tecnología.

Tabla I. Estadísticas de población y uso de Internet en Centro América

PAÍS	Población	Usuarios de Internet	% de penetración	Crecimiento (2000-2008)
Belice	301,270	32,000	10.60%	113.30%
Costa Rica	4,195,914	1,500,000	35.70%	500.00%
El Salvador	7,066,403	700,000	9.90%	1650.00%
Guatemala	13,002,206	1,320,000	10.20%	1930.80%
Honduras	7,639,327	344,100	4.50%	760.30%
Nicaragua	5,785,846	155,000	2.70%	210.00%
Panamá	3,292,693	264,316	8.00%	487.40%

Fuente: *Internet World Stats, Usage and Population Statistics*
<http://www.internetworldstats.com>

1.2 Los negocios electrónicos y el comercio electrónico

A medida que las tecnologías de Internet se han desarrollado, el uso y significado de los términos negocios electrónicos y comercio electrónico se han confundido. Y aunque ambos términos están íntimamente relacionados y regularmente son usados indistintamente, existen algunas diferencias importantes en la referencia de su alcance.

1.2.1 ¿Qué son los negocios electrónicos?

Los negocios electrónicos se pueden definir como “la conducción de negocios de forma electrónica, de forma interna y externa de la organización, con clientes, comunidades y socios” [1]. Otra definición de negocios electrónicos más específica es “cualquier proceso que una organización de

negocios conduce sobre una red de computadoras. Las organizaciones de negocios incluyen cualquier entidad lucrativa, no lucrativa y gubernamental. Sus procesos incluyen producción orientada al cliente, y procesos internos o administrativos del negocio” [2].

El término de negocios electrónicos se utiliza implícitamente con el de comercio electrónico, pero además de las transacciones comerciales, los negocios electrónicos también pueden incluir otros servicios al cliente, como soporte, mercadeo y comunicaciones, así como colaboración con socios de negocios.

El concepto de servicios electrónicos, o la disposición de servicios y tareas a través de Internet, mediante proveedores de servicios también se consideran dentro del alcance de los negocios electrónicos.

En Guatemala, ya se han empezado a ver cambios en relación a este tema, y prueba de ello es que en el año 2005 el *Banco Interamericano de Desarrollo (BID)* invirtió 1.5 millones de dólares junto con la *Cámara de Comercio de Guatemala (CCG)*, para impulsar proyectos de tecnologías de información, con el objetivo de que las Pequeñas y Medianas Empresas (PYMES) logren incrementar su productividad y eficiencia [3].

1.2.2 ¿Qué es comercio electrónico?

El comercio electrónico es el resultado de realizar negocios electrónicos. Se refiere más específicamente a las transacciones comerciales involucradas al vender y comprar bienes y servicios de forma electrónica. Los sistemas en línea son comúnmente usados en el comercio electrónico, y las transacciones

pueden ser conducidas utilizando cualquiera de las aplicaciones que se basan en Internet, como correo electrónico, mensajería instantánea, tiendas de compra, servicios Web o el FTP.

Según la Cámara de Comercio de Guatemala, el comercio electrónico puede definirse como:

- “La aplicación de la avanzada tecnología de información para incrementar la eficiencia de las relaciones empresariales entre socios comerciales” [4].
- “Es el uso de la tecnología computacional y de telecomunicaciones que se realiza entre empresas o bien entre vendedores y compradores, para apoyar el comercio de bienes y servicios” [4].

1.3 Tipos de negocios electrónicos

El uso de Internet se está incrementando para lograr el lanzamiento, manejo y mejora de los negocios tradicionales, desde comunicaciones simples hasta llegar a realizar análisis complejos.

Cada vez que Internet es usada dentro de un negocio, se puede considerar como una actividad de los negocios electrónicos. Algunos componentes de los negocios electrónicos han existido desde hace varios años y todavía se utilizan frecuentemente, como el fax, banca telefónica y video conferencia. Si se ha realizado alguna de las siguientes actividades, entonces ya ha conducido negocios electrónicos:

- Comunicarse con clientes y proveedores mediante correo electrónico.

- Vender y promover productos y servicios mediante la Web o email.
- Usar Internet para realizar banca en línea y pagar las cuentas.
- Investigar información acerca de los clientes y competidores utilizando sitios Web.
- Manejar y distribuir documentos internos de la compañía mediante una intranet.

Los negocios electrónicos resultan más exitosos cuando se utilizan para mejorar las funciones básicas del negocio, por ejemplo:

- Vender productos y servicios.
- Manejo de la relación con el cliente.
- Inteligencia competitiva e investigación.
- Recursos humanos.
- Producción y distribución.
- Administración y estrategia.

Los negocios electrónicos se pueden dividir según el tipo de usuarios involucrados en la comunicación en: B2C, B2B, B2G. También se pueden clasificar por el tipo de tecnología involucrada en: P2P y *mCommerce*.

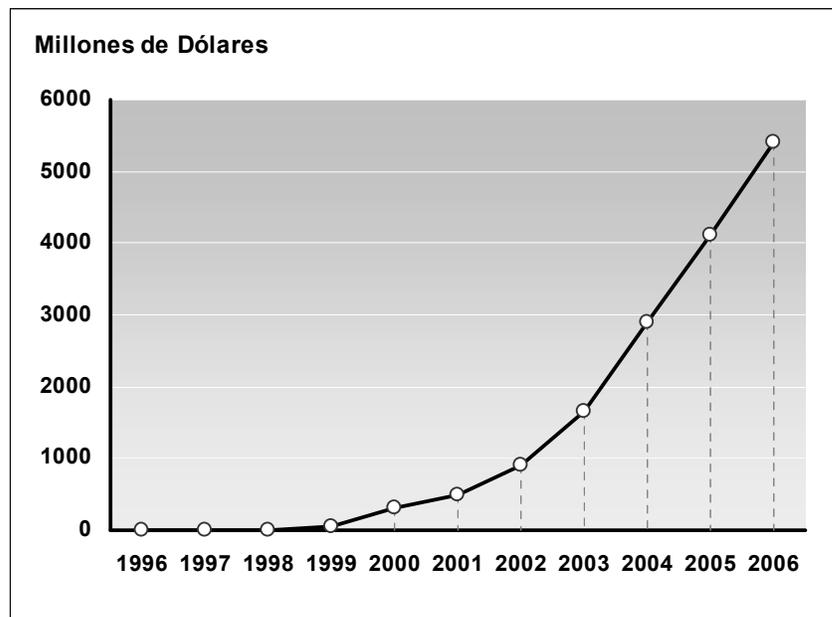
1.3.1 B2C

Business-to-Consumer (de negocio a consumidor). Es el tipo de negocio electrónico más popular. Se caracteriza en que los negocios tratan de vender sus productos y servicios a consumidores individuales. Aunque el *B2C* es pequeño, ha crecido exponencialmente desde 1995, y es el tipo de negocio electrónico que la mayoría de consumidores van a encontrar.

1.3.2 B2B

Business-to-Business (de negocio a negocio). Se caracteriza en que los negocios venden sus productos y servicios a otros negocios, y es el tipo de negocio electrónico más grande que existe. La figura 2 muestra el crecimiento que ha tenido en los últimos años, sugiriendo su potencial futuro.

Figura 2. Crecimiento del comercio electrónico *B2B* en Estados Unidos de América



Fuente: *Júpiter Media Metrix*

1.3.3 B2G

Business-to-Government (de negocio a gobierno). Este tipo de negocio electrónico se refiere a las comunicaciones en línea que ocurren entre los negocios y el gobierno. Puede incluir el pago de impuestos y cualquier otra transacción relacionada con el gobierno. En Guatemala tenemos dos ejemplos claros de este tipo de negocios electrónicos:

- **Bancasat:** es un sistema electrónico de pago de impuestos, que la Superintendencia de Administración Tributaria (SAT) pone a su disposición por medio de Internet a través de varios bancos del sistema [5].
- **Guatecompras:** es el sistema de información de Contrataciones y Adquisiciones que el Estado de Guatemala utiliza para comprar y contratar bienes y servicios, es un mercado electrónico operado a través de Internet [6].

1.3.4 P2P

Peer-to-Peer (de punto a punto). Existen varios modelos de *P2P*: el puro, el híbrido y los super-nodos. Por ejemplo, en el modelo puro, los usuarios pueden comunicarse directamente de una computadora a otra sin que la comunicación sea intermediada por un servidor centralizado. Este modelo fue popularizado por la aplicación *Napster*, que en su momento permitía a sus usuarios compartir archivos mp3, sin embargo, este modelo de comunicación ofrece nuevas formas de hacer negocios electrónicos. Estos modelos se discuten más profundamente en el capítulo 3.

1.4 Modelos de negocio

Existe una gran variedad de modelos de negocio que son utilizados antes, durante y después de realizar una transacción de negocio. Un modelo de negocio es un conjunto de actividades planeadas, algunas veces denominado procesos de negocio, diseñados para producir una ganancia en un mercado.

1.4.1 Elementos de un modelo de negocio

1.4.1.1 Proposición de valor

Define la forma en que los productos o servicios de una organización llenan las necesidades de los clientes. Para desarrollar y analizar una proposición de valor, hay que responder las siguientes preguntas: ¿Por qué los clientes escogen hacer negocio en nuestra compañía en lugar de hacerlo en otra?, ¿Qué provee nuestra compañía que no provee la competencia?

1.4.1.2 Modelo de ganancia

Describe como una compañía obtiene ganancias y logra producir un retorno superior al capital invertido. Para considerar a una compañía exitosa debe producir ganancias mayores a las inversiones alternativas. Existen varios modelos de ganancia que han sido desarrollados en el comercio electrónico, muchas compañías dependen de uno o la combinación de varios. Los modelos más importantes se presentan en la Tabla II.

Tabla II. Modelos de ganancia en los negocios electrónicos.

Modelo	Descripción
Publicidad	Un sitio Web que ofrece a sus usuarios contenidos, servicios y productos, también provee un espacio para anuncios publicitarios y recibe comisiones de los anunciantes que publican sus anuncios.
Suscripción	Un sitio Web cobra una comisión por suscripción para que sus usuarios puedan tener acceso al contenido o servicios que ofrecen.
Transacción	En este modelo una compañía recibe una comisión por iniciar o ejecutar una transacción. Por ejemplo: <i>gubiz.com</i> recibe una pequeña comisión cada vez que se vende un producto a través de su sitio.
Ventas	Las compañías obtienen ganancias vendiendo bienes, información o servicios a los clientes. Por ejemplo: <i>Amazon.com</i>
Afiliación	Son sitios que facilitan negocios a un afiliado y reciben una comisión o porcentaje por cualquier venta realizada

Fuente: Kenneth C. Laudon, Carol G. Traver *E-commerce: business. Technology. Society* (2002)

1.4.1.3 Oportunidad de mercado

Se refiere al mercado objetivo al que se quiere dirigir una compañía y a las oportunidades financieras potenciales disponibles para la compañía en ese espacio de mercado.

1.4.1.4 Ambiente competitivo

Se refiere a las otras compañías que están operando en el mismo espacio de mercado vendiendo productos similares. Este ambiente es influenciado por varios factores: cuantos competidores hay, que tan grandes son sus operaciones, el segmento de mercado de cada competidor, que tan lucrativas son estas compañías y cuál es el precio de sus productos.

1.4.1.5 Ventaja competitiva

Las compañías adquieren una ventaja competitiva cuando pueden producir un producto superior y/o colocar el producto en el mercado a un precio más bajo que la mayoría o todos sus competidores. Una compañía obtiene ventaja competitiva porque de alguna manera ha tenido acceso a factores de producción que han sido negados o que no han sido posibles para sus competidores.

1.4.1.6 Estrategia de mercado

La estrategia de mercado es todo lo que una compañía hace para promover sus productos y servicios a clientes potenciales, es un plan que se arma y detalla exactamente como se pretende entrar al mercado y atraer nuevos clientes.

1.4.1.7 Desarrollo organizacional

Todas las compañías que quieren crecer, necesitan tener un plan de desarrollo organizacional, el cual describe como la compañía organizará el trabajo que debe realizarse. Por lo regular, el trabajo es organizado en departamentos, se definen los puestos y se contrata el personal que realizará las tareas necesarias.

1.4.1.8 Equipo de administración

Este es el elemento más importante del modelo de negocio y es el encargado de hacer que el modelo funcione. Un equipo de administración fuerte le da al modelo credibilidad para atraer inversionistas, conocimiento específico del mercado y experiencia en implementar planes de negocio.

1.5 Usos de los negocios electrónicos

Las tecnologías de información pueden aplicarse a través de todo el alcance del negocio o aplicarse a un aspecto específico relacionado con las compras, ventas, promoción o administración. Los negocios electrónicos no se aplican solamente a las grandes empresas. Las Pequeñas y Medianas Empresas (PYMES) también pueden encontrar sistemas y procesos de negocios electrónicos que puedan ayudar a reducir los costos, mejorar la comunicación, administrar recursos humanos, mejorar la productividad, incrementar las ganancias o encontrar nuevas oportunidades en el mercado.

Una porción creciente de negocios utilizan las tecnologías de Internet como un aspecto central de sus operaciones. La mayoría de usos de los negocios electrónicos caen en ciertas categorías: lidiar con el gobierno, correo electrónico, banca en línea, compras en línea e investigación de mercados.

1.5.1 Lidiar con el Gobierno

A medida que las personas se han convertido en usuarios de Internet, los negocios y el gobierno han ido cambiando sus prácticas y procesos para cumplir con la demanda de información y prestar mejores servicios. En la Tabla III se listan varios sitios Web relacionados con el Gobierno de Guatemala [6].

Internet ofrece oportunidades a los negocios para interactuar con agencias de gobiernos locales, nacionales e internacionales, sin importar en donde se encuentran situadas. Los siguientes son ejemplos de operaciones que los negocios pueden realizar con el gobierno:

- Proveer servicios de gobierno por teléfono o fax.
- Uso de teléfonos móviles y agendas personales digitales.
- Sistemas de tecnologías de información internos del gobierno.

Tabla III. Sitios Web relacionados con el Gobierno de Guatemala

Descripción	Dirección URL
Cámara Guatemalteca de la Construcción	http://www.simop.info
Ministerio de Finanzas Públicas	http://www.minfin.gob.gt/
Portal del Gobierno de Guatemala	http://www.guatemala.gob.gt
Portal de Transparencia Fiscal	http://transparencia.minfin.gob.gt
Superintendencia de Administración	http://www.sat.gob.gt

Fuente: Guatecompras. www.guatecompras.gt

1.5.2 Correo electrónico

El correo electrónico o *email*, se refiere a la transmisión de mensajes sobre una red de computadoras como intranet o Internet. El correo electrónico permite una comunicación más eficiente dentro del negocio y fuera de él, con los clientes, proveedores y socios.

Según una encuesta realizada por la Superintendencia de Telecomunicaciones (SIT) de mayo a diciembre del 2006, alrededor de 258 mil estudiantes en el país ya utilizan computadoras, y casi 39 mil de estas máquinas fueron instaladas en 967 centros educativos. Este mismo estudio reveló que al menos 28 mil 684 estudiantes tienen computadora en casa y 25 mil 352 han abierto una cuenta de correo electrónico [7].

Estos son sólo datos preliminares, y a nivel regional se espera que existan más usuarios que posean por lo menos una cuenta de correo electrónico, dado que el correo electrónico es la aplicación de Internet más utilizada.

El correo electrónico puede ser utilizado para:

- Circular información de noticias, actualizaciones y cartas a los clientes.
- Colocar órdenes con otros proveedores.
- Arreglar reuniones.
- Mantener relaciones de negocio.
- Actualizar al personal con noticias internas.

1.5.2.1 Ventajas

- **Velocidad:** el correo electrónico viaja grandes distancias en segundos, dependiendo del tamaño del mensaje y la conexión de red.
- **Conveniencia:** dependiendo de la ubicación y conexión del equipo, se puede recibir y enviar correo sin importar la ubicación.
- **Flexibilidad:** el software de correo electrónico permite organizar, archivar y buscar en los mensajes, permitiendo acceder la información de forma fácil.
- **Costo:** en comparación con los cargos postales, los costos de instalación y mantenimiento del correo electrónico, son mínimos.
- **Adjuntos:** cualquier tipo de documentos pueden ser adjuntados y enviados de forma electrónica a cualquier dirección de correo.

1.5.2.2 Desventajas

- **Spam:** ser dueño de una cuenta de correo, permite el ingreso de mensajes no solicitados que son enviados en grandes volúmenes.
- **Seguridad:** aunque hay disponible encriptación y opciones de seguridad, los mensajes de correo se pueden interceptar por una tercera persona sin que el emisor o receptor se den cuenta.

1.5.3 Banca en línea

La banca en línea es un servicio que permite a los cuenta habientes de un banco realizar transacciones bancarias a través del sitio Web del banco. La banca en línea puede utilizarse para uso personal o de negocios, en cada caso el banco establece sus términos y condiciones de uso.

El número de clientes que prefieren utilizar banca en línea para enfrentar sus finanzas está creciendo rápidamente, debido a la gran conveniencia que ofrece con sus características para administrar cuentas y pagar facturas. En la tabla IV se pueden apreciar los bancos de Guatemala que ofrecen este servicio.

Tabla IV. Bancos guatemaltecos que ofrecen servicios en línea

Nombre del banco	Dirección URL
Banco Agromercantil	http://www.agromercantil.com.gt
CITIBANK	http://www.citibank.com/guatemala
Banco Corporativo	http://www.corpobanco.com.gt
Crédito Hipotecario Nacional (CHN)	http://www.chn.com.gt
Banco de América Central (BAC)	http://www.bac.com.gt
Banco de Desarrollo Rural	http://www.banrural.com.gt
Banco de Exportación (Banex)	http://www.banex.net.gt
Banco de la República	http://www.larepublica.com
Banco de Occidente	http://www.occidente.com.gt
Banco del Quetzal (Banquetzal)	http://www.banquetzal.com.gt
Banco G&T Continental	http://www.gytcontinental.com.gt
Banco Industrial	http://www.bi.com.gt
Banco Internacional	http://www.bancointernacional.com.gt
Banco Privado para el Desarrollo	http://www.bancasol.com.gt
Banco Reformador (Bancor)	http://www.bancoreformador.com.gt
Banco SCI	http://www.sci.com.gt
Banco Uno	http://www.bancouno.com.gt
Banco Vivibanco	http://www.vivibanco.com.gt
Banco Inmobiliario	http://www.vivibanco.com.gt
Banco de Crédito	http://www.bancredit.com.gt

Fuente: Bancasat

Existen varios factores a considerar antes de iniciar a utilizar los servicios de la banca en línea, como lo son: manejar el acceso, seguridad, software, servicios y pagos.

1.5.3.1 Manejar el acceso

Aunque esto no es necesario al tener una computadora en casa u oficina, por razones de seguridad si se recomienda. Si se utiliza una computadora pública, por ejemplo en una librería o café de Internet, al terminar, hay que asegurarse que ha salido completamente del sistema bancario antes de abandonar la computadora.

1.5.3.2 Seguridad

Los sitios Web bancarios son sitios seguros que encriptan la información y solicitan una clave de acceso. Si no conoce la seguridad del sitio, pregunte a la entidad bancaria las medidas que toma para evitar el fraude en línea.

1.5.3.3 Software

A diferencia de la banca en línea personal, los negocios algunas veces tendrán que pagar una cuota y requerir la instalación de software especial. Además de esto, pueden existir medidas de seguridad adicionales para realizar transacciones de negocios.

1.5.3.4 Servicios

Cada entidad bancaria pondrá a disposición un conjunto distinto de servicios y términos de uso. A continuación se listan los servicios que son comúnmente encontrados en línea:

- Consulta de saldo y estados de cuenta.
- Vincular personal y cuentas de negocios.
- Transferir dinero entre sus propias cuentas.
- Programar pagos automáticos de renta y préstamos
- Transferir fondos a una cuenta de otra institución.
- Realizar pagos de cuentas que requieren más de una firma.

1.5.3.5 Pagos

Pagar a los proveedores en línea, puede reducir costos, al no requerir cheques y evitar la visita al banco para completar la transacción. El banco también puede generar y enviar cheques de papel a los que no estén configurados para recibir pagos electrónicos.

1.5.4 Compras en línea

Las compras son la actividad esencial de casi cualquier negocio. Internet y las tecnologías móviles e inalámbricas pueden ser usadas en varias etapas del proceso de compra, y permiten:

- Buscar información de productos y proveedores.

- Ver y descargar el catálogo de productos.
- Comparar productos y precios.
- Enviar correo a un proveedor.
- Realizar órdenes de compra electrónicas.
- Realizar pagos electrónicos.

En Guatemala existen muchas empresas que ofrecen sus productos y servicios a través de Internet, he aquí unos ejemplos populares:

- **Gubiz.com:** compra y venta de productos de persona a persona.
- **MiSuper.com:** compra y envío de abarrotes.
- **Quickphotoonline.com:** impresión de fotografías digitales.

Existen muchos beneficios de usar las tecnologías de información y comunicación en el B2B:

- Reducir el papeleo.
- Mejorar la productividad y eficiencia.
- Procesar las órdenes más rápidamente.
- Incrementar el éxito de localizar elementos difíciles de conseguir.
- Agilizar los pagos.
- Reducir el tiempo de viajar para cotizar y comprar productos.
- Localizar proveedores locales y regionales.

Para habilitar la integración del B2B, es necesario que los procesos internos de la organización soporten el intercambio en línea. Hay que motivar al personal y encontrar o desarrollar un sistema que beneficie a todos los involucrados en la organización.

1.5.5 Investigación de mercados

Internet es una herramienta poderosa para encontrar información de proveedores, clientes y competidores. Puede asistir en tareas simples, como encontrar los detalles de contacto de un proveedor, así como investigación de mercado más compleja.

1.5.5.1 Motores de búsqueda

Los motores de búsqueda son populares para encontrar una página Web, pero sin una estrategia clara de búsqueda, los motores de búsqueda pueden retornar una cantidad increíble de resultados. Al encontrar un sitio relevante, hay que tomar nota de las palabras clave y volver a realizar una búsqueda con éstas.

1.5.5.2 Directorios y portales

Un directorio en línea es un catálogo de sitios Web que han sido localizados y organizados en temas y subtemas por editores. Son útiles para encontrar información general, porque se puede explorar por área de interés. Un ejemplo popular de un directorio guatemalteco es: www.deguate.com [8].

Como es relativamente fácil publicar información en Internet, es importante asegurarse que la fuente de información sea confiable. De lo contrario, estamos arriesgándonos en tomar decisiones equivocadas al usar información que no es correcta.

1.6 Beneficios

Los negocios electrónicos apuntan a reducir costos y mejorar el rendimiento al integrar los sistemas del negocio y eliminar la duplicidad de esfuerzos. Al tomar ventaja de las TICs, las organizaciones pueden usar las soluciones de negocios electrónicos para hacer los procesos existentes más eficientes [1].

Aspectos como mercadeo, acceso a Internet, correo electrónico y un sitio Web, pueden ayudar a mejorar la imagen del negocio, identificando prospectos futuros, reforzando relaciones con los clientes y mejorando procesos internos.

Los negocios electrónicos pueden proveer beneficios estratégicos así como beneficios operacionales. La estrategia permite reforzar las relaciones entre el negocio, los clientes y proveedores, identificando nuevas oportunidades de mercado. Operacionalmente, un beneficio clave es la reducción de tiempo y trabajo requerido para completar procesos de negocio estándar.

1.6.1 Administración

Los negocios electrónicos pueden crear varias eficiencias administrativas que pueden ahorrar costos y tiempo de trabajo [1]. Entre esas eficiencias tenemos:

- Preparación de documentos, copias y publicación.
- Ingreso de datos y procesamiento de formularios.
- Reservaciones y administración de órdenes.
- Reuniones y comunicaciones.
- Proceso de las nóminas de pago.

1.6.2 Producción, intercambio y distribución

Para algunas organizaciones, los negocios electrónicos pueden cambiar fundamentalmente la forma en que operan, ofreciendo beneficios a través de toda la cadena de valor [1]. Por ejemplo:

- Seguimiento en línea: procesamiento de órdenes integrado y otros recursos en línea pueden llevar a eficiencias mejoradas en las operaciones de compra.
- Espacios de mercado: al quitar el intermediario, los negocios pueden vender directamente a los clientes, como el caso de Dell [9].
- Manejo de la cadena de aprovisionamiento: al integrar sistemas para que trabajen de cerca con los proveedores y clientes.
- Inventario: un inventario en línea, puede maximizar la disponibilidad de los productos al permitir análisis más rápido y reconocimiento de tendencias en la demanda.
- Manejo de órdenes: reduce la cantidad de tiempo entre el alojamiento de la orden y la llegada de los bienes.

1.6.3 Recursos humanos

La mejora significativa en los sistemas de recursos humanos y la satisfacción del personal, depende de los sistemas tecnológicos [1]. Por ejemplo:

- Reclutamiento: los servicios en línea pueden agilizar el proceso de contratar personal, reducir los costos y permiten alcanzar segmentos de personal especializados.
- Administración de la carga de trabajo: el acceso mejorado a la información, permite al personal tomar decisiones informadas a tiempo.
- Enfoque estratégico: mover el énfasis lejos de las tareas de rutina, ayuda al personal concentrarse en las áreas centrales del negocio, por ejemplo buscando nuevas oportunidades y mejorando los procesos de negocio.

1.6.4 Servicio al cliente

Desarrollar las operaciones internas, el software y las capacidades de Internet dentro del negocio, puede ayudar a administrar la relación con los clientes de una forma estratégica y sistemática [1]. Por ejemplo:

- Tiempo de respuesta: preguntas de los clientes son respondidas en periodos de tiempo más cortos.

- Ciclos de entrega: se incrementa la satisfacción del cliente al tener tiempos de entrega más rápidos.
- Expectativa del cliente: un sitio Web ofrece servicios a los clientes sin las restricciones de las horas de visita de los negocios tradicionales.
- Información de catálogo detallada: los sistemas de comercio electrónico pueden ofrecer información sobre los productos, órdenes de compra y entrega de una forma más detallada.
- Información después de la venta: reducir las demandas de servicio de los clientes, al proveer respuestas a las preguntas de los clientes en forma de contenido accesible en línea.

1.6.5 Mercadeo y comunicaciones

Internet es un canal poderoso para alcanzar nuevos mercados y llevar información a los clientes, proveedores y socios [1]. Por ejemplo:

- Publicaciones: folletos, noticias, manuales y lanzamientos de prensa pueden distribuirse rápidamente y eficientemente usando sitios Web y el comercio electrónico.
- Retroalimentación del cliente: encuestas en línea permiten obtener información de valor de los clientes.
- Alcance de mercado: la naturaleza global de Internet puede ayudar ampliando la base de clientes relativamente a bajo costo.

- Actualizaciones regulares: información actualizada y relevante de los productos y servicios puede ser enviada por correo a los clientes.
- Reputación: tomar las nuevas tecnologías le dan credibilidad al negocio, catalogándolo como innovador y orientado al futuro.

1.6.6 Colaboración

Los negocios pueden escoger formar parte de un conjunto de negocios colaboradores, haciendo uso de las TICs para mejorar la colaboración y coordinación entre socios de negocios. Se puede mejorar la cadena de aprovisionamiento [1]. Por ejemplo:

- Procesos: al integrar procesos internos y externos a través de múltiples ubicaciones.
- Relaciones: socios, proveedores y el personal pueden trabajar en tareas y compartir documentos a través de Internet mediante el correo electrónico, salas de Chat, intranets y extranets.
- Oportunidades: la comunicación mejorada puede ayudar a los socios a identificar nuevas oportunidades que emergen y rápidamente llevan nuevos productos y servicios al mercado.

1.6.7 Investigación de mercados

El acceso rápido a información actualizada es un beneficio principal de Internet y permite mantener una ventaja competitiva en las siguientes áreas:

- Inteligencia competitiva: reunir datos comparativos sobre los competidores de sus sitios Web y de los datos de la bolsa de valores.
- Desarrollo de productos: identificar oportunidades de negocio emergentes.
- Información de mercado: como tasas de cambio, perfiles de los países, tendencias de los clientes y precios del mercado.
- Información de gobierno: como licencias, transacciones en línea, impuestos y leyes.
- Directorios: como bases de datos de proveedores, portales regionales y directorios telefónicos.

En general las TICs han venido a cambiar permanentemente la forma en que las organizaciones se manejan, al mismo tiempo han cambiado la forma en que las personas se comunican e interactúan entre sí.

2. LA WEB SEMÁNTICA

2.1 Introducción

En la actualidad, la Web es un espacio donde las personas podemos compartir información. Las páginas Web son diseñadas por personas para ser utilizadas por personas. No existe un estándar para mostrar la información, es por eso que cada sitio despliega la información de acuerdo al perfil de los usuarios que desea atraer.

Cada motor de búsqueda indexa de forma diferente las páginas Web y la búsqueda también la realizan de forma diferente. La utilización de palabras clave dentro del código *HTML* es útil pero está lejos de ser perfecta, ya que no hay forma de diferenciar un tipo de información de otra, lo que nos lleva a obtener miles de resultados no relacionados al hacer una búsqueda con este esquema.

Por otra parte, los motores de búsqueda actuales, no están diseñados para comprender la información contenida en las páginas Web. Esto se debe a que es imposible interpretar la información si no se utiliza un estándar para representarla.

Sin duda alguna, Internet tiene sus ventajas al buscar información, sin embargo, no tiene una forma precisa de encontrarla y tampoco pueden realizar deducciones con la información que ya existe, lo que limita realizar búsquedas realmente inteligentes.

2.1.1 Visión

La idea detrás de la Web semántica, es que la Web sea un banco de información bien definida, clasificada y que pueda ser entendida por las computadoras y personas sin supervisión. De esta forma, los motores de búsqueda serán diseñados para manejar la información que se encuentra en los sitios Web de forma casi automatizada [10].

Según el W3C, “La Web Semántica provee un marco de trabajo que permite compartir y reutilizar datos a través de los límites de las aplicaciones, las organizaciones y las comunidades. Es un esfuerzo liderado por el W3C con la participación de una gran cantidad de investigadores y socios industriales” [11].

Esto pretende mejorar la búsqueda de información, además de incentivar la creación de aplicaciones de comercio electrónico, utilizando una forma estándar para representar la información. Los motores de búsqueda podrán encontrar la información de forma precisa, y también podrán realizar deducciones automáticamente obteniendo información relacionada con la encontrada en las páginas y con los parámetros de la consulta realizada por un usuario.

Según Tim Berners-Lee, el padre de la Web Semántica, “El primer paso es colocar los datos en la Web de una forma que las máquinas la puedan entender naturalmente, o convertirla a esa forma. Esto crea lo que yo llamo la Web Semántica, una Web de datos que puedan ser procesados directa o indirectamente por las máquinas” [12].

“La Web Semántica es una extensión de la Web actual en la cual la información se da mediante un significado bien definido, lo que facilita que los ordenadores y la gente trabajen en cooperación” [13].

2.2 La Web Semántica en los negocios

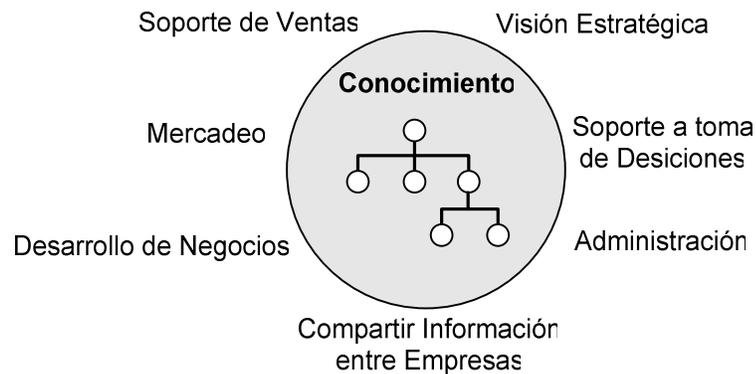
Muchos administradores han dicho que la visión de la Web Semántica suena increíble, pero se han preguntado ¿como la pueden usar y por que deben invertir en ella?” [14].

Es sentido común que la organización que tenga la mejor información, sepa dónde encontrarla y pueda utilizarla más rápidamente, gana. El conocimiento es poder. Antes se decía que la organización con la mayor información ganaba, pero ahora que estamos ahogados literalmente en terabytes de información, nos damos cuenta que necesitamos encontrar la información correcta rápidamente para poder tomar decisiones bien informadas. También nos hemos dado cuenta que el conocimiento (el significado de los datos), y no solo los datos, es lo más importante. La organización que pueda hacer esto, podrá utilizar sus recursos de la mejor forma y obtendrá una ventaja competitiva. La clave es el manejo del conocimiento.

La Web Semántica promete poner orden al caos de información que enfrentamos actualmente. Para adaptarnos, debemos empezar a utilizar las tecnologías discutidas más adelante.

La figura 3 provee una visión de cómo las organizaciones pueden girar alrededor de la Web Semántica, impactando virtualmente cada pieza de la organización.

Figura 3. Uso de la Web Semántica en la Organización



Fuente: Michael C. Daconta y otros. *The Semantic Web. A Guide to the Future of XML, Web Services and Knowledge Management.*

2.2.1 Soporte a la toma de decisiones

Tener el conocimiento y no solo los datos al alcance, permite tomar mejores decisiones. Si nuestra organización tiene mucha información, entonces seguramente existen relaciones implícitas ocultas en los datos que las tecnologías de la Web Semántica nos ayudarán a encontrar.

Actualmente se desarrollan sistemas basados en semánticas que sirven de soporte para la toma de decisiones, estos sistemas se orientan al análisis de agentes de software y la interacción entre el usuario y la computadora con el fin de dar poder al usuario final para que tome decisiones bien informadas.

2.2.2 Desarrollo de negocios

Es importante para los miembros de una organización, tener información actualizada en tiempo real que pueda ayudar a obtener nuevos negocios. Con la Web Semántica se podrán desarrollar sistemas *CRM* más inteligentes que puedan disminuir el tiempo necesario para lograr interoperabilidad con los sistemas antiguos.

Los expertos del comercio electrónico creen que la Web Semántica puede ser utilizada para acoplar negocios electrónicos. Este es un proceso donde los negocios son conectados con socios o clientes potenciales. Actualmente se contrata a un corredor de negocios para hacer el acople, sin embargo las tecnologías de la Web Semántica podrán automatizar dicho proceso a niveles más avanzados.

2.2.3 Compartir y descubrir información

Compartir información y la comunicación son clave en cualquier organización, pero a la medida que las organizaciones crecen y colectan más información, esto se vuelve más difícil. Todos entendemos la importancia de no inventar la rueda, pero cuantas veces de forma no intencional, se han duplicado los esfuerzos. Cuando las organizaciones se vuelven grandes, los orificios de comunicación son inevitables. Pero con un poco de esfuerzo, se puede crear una base de conocimiento que incluya una descripción de cada proyecto en desarrollo y el equipo que lo construye. Con esto, los empleados podrían encontrar fácilmente información relevante. Esto se logra con la Web Semántica.

2.2.4 Administración y automatización

Tener una base de conocimiento, da a los programas de software la capacidad de automatizar tareas administrativas. Por ejemplo, realizar una reservación de viaje, es una tarea donde la Web Semántica y los servicios Web pueden asistir en convertir una tarea difícil en fácil. Cada persona tiene preferencias de viaje, y se deben considerar los siguientes puntos:

- Transporte (carro, tren, bus, avión)
- Hotel y servicios asociados.
- Línea aérea y descuento de viajero frecuente.
- Cercanía del hotel a los lugares visitados.
- Preferencias de la habitación de hotel (fumador, bar, comida)
- Opciones para rentar vehículos.
- Precio

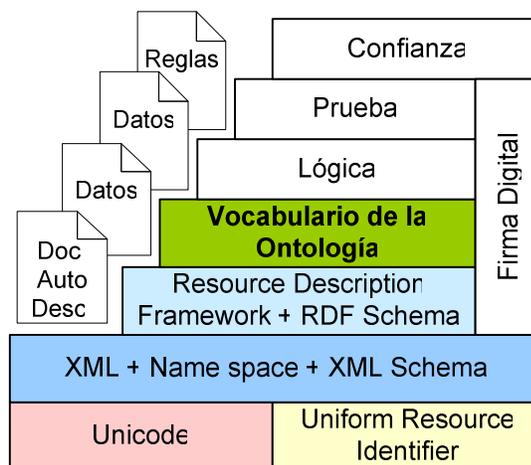
Hacer la reservación manualmente, puede implicar horas de estar revisando información y tratando de que todas las opciones que seleccionemos se ajusten a nuestras preferencias.

Con la Web Semántica se podrá automatizar el proceso, al tener todas las preferencias y reglas almacenadas en una base de conocimiento, un agente de software podría seleccionar o sugerir el viaje por uno, y de hecho ya existe un sitio que ofrece este servicio, *Tripit.com* [36], planifica los lugares y restaurantes a visitar, dados la fecha de llegada, el hotel de estadía y la fecha de partida de un viaje planificado.

2.3 Tecnologías

Para que la visión de la Web Semántica sea una realidad, es necesario que todos participemos en ella, para lograr esto, debemos usar estándares populares que han probado ser un éxito en Internet. En la figura 4, se pueden apreciar las tecnologías que la conforman.

Figura 4. Tecnologías que forman la Web Semántica



Fuente: Miguel Angel Abian, "EL FUTURO DE LA WEB, XML, RDF/RDFS, ontologías y la Web Semántica".

2.3.1 URI

Los *URI* (*Uniform Resource Identifiers*, identificadores uniformes de recursos), cuyo subconjunto más popular, el *URL* (*Uniform Resource Locators*, localizadores uniformes de recursos), proporcionan la forma de identificar inequívocamente cualquier recurso en Internet [15]. En la Web Semántica se utilizarán los *URI* para identificar los objetos del mundo, por ejemplo un proveedor será representado mediante un *URI*, así como la función que una persona desempeñe en una organización se representará mediante un *URI*.

2.3.2 XML y XML Schema

El *XML* (*eXtensible Markup Language*, lenguaje extensible de etiquetado), es un metalenguaje que permite definir lenguajes propios de una forma estándar. *XML* permite presentar información en Internet y a la vez darle significado, para que otros programas de software puedan entender y utilizar los datos. Tiene el potencial de convertirse en el lenguaje común del comercio electrónico y el manejo del conocimiento.

El *XML* forma la base sintáctica para la Web Semántica. Todas las demás tecnologías que proveen características, se desarrollaran sobre *XML*. Al hacer que las demás tecnologías se basen en *XML*, se garantiza un nivel base de interoperabilidad.

Sin embargo *XML* no es suficiente, ya que solo provee interoperabilidad sintáctica. En otras palabras, compartir un documento *XML*, agrega significado al contenido, pero solo cuando ambas partes conocen y entienden ese contenido. Por ejemplo, si alguien coloca la etiqueta `<precio>Q50.00</precio>` a un producto, y alguien más usa la etiqueta `<valor>Q50</valor >`, no existe forma de que una máquina sepa que las dos etiquetas significan lo mismo, al menos que usen tecnologías de la Web Semántica como las Ontologías.

XML Schema es un lenguaje de definición que permite restringir documentos *XML* a un vocabulario específico y a cierta estructura jerárquica. En este sentido juega dos roles importantes:

- Como plantilla para generar instancias del tipo de documento.
- Como validación para asegurar la precisión de los documentos.

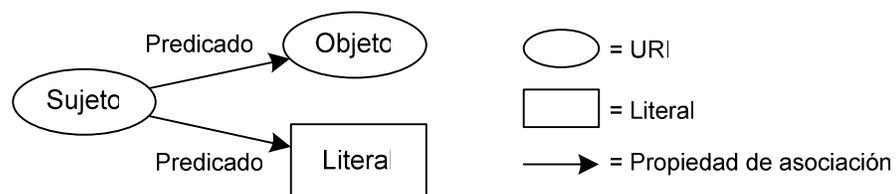
2.3.3 RDF y RDF Schema

Mientras que *XML* estructura de forma estándar los datos, *RDF* (*Resource Description Framework*, marco de descripción de recursos) provee el mecanismo para darle significado a los datos. Es un modelo para representar datos acerca de los recursos de la Web. Un recurso Web es cualquier objeto que posea un identificador *URI* [15].

RDF Schema es un sistema para tipificar el *RDF*. Provee el mecanismo para definir atributos específicos de un dominio y clases de recursos a las cuales se les puede aplicar los atributos. *RDF* y *RDFS*, proveen una forma simple de representar conocimiento para los recursos Web. *RDF* es el estándar propuesto por la *W3C* para la representación e intercambio de meta-datos en la Web Semántica. *RDFS* es el estándar dedicado a representar conocimiento Ontológico, usado en sentencias *RDF*.

El modelo *RDF* también es llamado tripleta, porque posee tres partes que son expresadas en términos de las partes gramaticales de una sentencia: sujeto, predicado y objeto.

Figura 5. Partes que forman el *RDF*



Sin embargo no es suficiente para modelar el conocimiento. Para especificar el significado de los datos de forma más precisa, se necesitan lenguajes más ricos en lógica y semántica.

2.4 Las Ontologías

La palabra Ontología tiene su origen en la filosofía, donde Aristóteles la utilizó para estudiar las categorías de la metafísica, tratando de analizar lo relativo al ser y la existencia. En el campo de las ciencias de la computación, fue utilizada por primera vez en la Inteligencia Artificial como un modelo para representar conocimiento. Una Ontología puede definirse como “una especificación explícita y formal de una conceptualización compartida” [16]. Analicemos esta definición:

- Es *explícita* porque define los conceptos, propiedades, relaciones, funciones, axiomas y restricciones que la componen.
- Es *formal* porque es legible e interpretable por computadoras.
- Es una *conceptualización* porque es un modelo abstracto y una vista simplificada de las entidades que representa.
- Finalmente, es *compartida* porque ha existido un consenso previo sobre la información, que ha sido acordado por un grupo de investigadores.

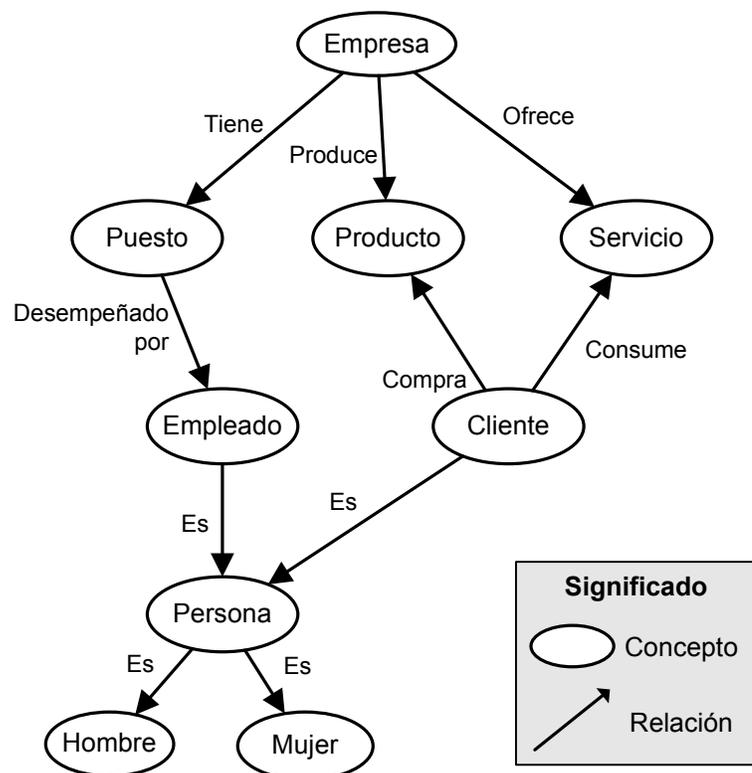
En resumen se puede decir que una Ontología es la definición de un conjunto de conceptos usados para describir y representar un área de conocimientos.

Las Ontologías son el medio para representar el conocimiento de una forma bien definida, clasificada y que pueda ser entendida por las computadoras y personas a la vez. Son necesarias para ayudar a los agentes

de búsqueda y herramientas de análisis, a recuperar información de la Web. Proveen una comprensión compartida y común de un dominio de conocimiento que puede ser comunicado entre personas y sistemas de información.

La Figura 6 ilustra una Ontología simplificada que describe el dominio de una empresa. En la figura se puede observar la relación entre los distintos conceptos que la componen.

Figura 6. Ejemplo de una Ontología sobre una Empresa



2.4.1 Estructura

Las Ontologías tienen un conjunto de componentes que son utilizados para representar el conocimiento de algún dominio [10].

- **Conceptos**

Son las ideas básicas que se quieren formalizar. Los conceptos son representados mediante clases. Que definen las características que comparten un grupo de objetos. Pueden existir subclases que representan conceptos más específicos que la superclase.

- **Atributos**

Describen las propiedades de las clases o instancias. Son representadas mediante *slots*. Por ejemplo podríamos agregar el atributo *fabricante* a la clase *producto*, para describir el responsable de fabricar dicho producto.

- **Relaciones**

Permiten crear un vínculo entre los conceptos del dominio, en conjunto las relaciones forman la taxonomía del dominio. Gran parte del poder de las Ontologías proviene de poder describir las relaciones. Como ejemplo de relaciones tenemos: *subclase de*, *superclase de*, *es un*, *parte de*.

- **Instancias**

Las instancias representan objetos o ejemplos reales de una clase de objetos. Por ejemplo, un vehículo es una *instancia* de la clase producto, y en el atributo *fabricante* podemos colocar el nombre del fabricante, por ejemplo *Citroen*.

- **Funciones**

Las funciones son un tipo especial de relación, que permiten identificar un elemento haciendo uso de una función que toma en cuenta varios elementos de una Ontología. Por ejemplo, pueden aparecer funciones como clasificar clase, asignar fecha.

- **Axiomas**

Son sentencias que son siempre verdaderas y pueden ser utilizadas para limitar la información, verificar la validez de ésta y deducir nueva información. Por ejemplo: “Si X y Y son de la clase Z, entonces X no es subclase de Y”, “Para todo X que cumpla la condición C1, X es Y”.

Los axiomas y la herencia de conceptos, permiten inferir nuevo conocimiento que no está descrito de forma explícita en la Ontología.

2.4.2 ¿Para qué crear una Ontología?

Existen varias razones que llevan al desarrollo de una Ontología. Sin embargo desarrollar una Ontología no es un fin en sí mismo, sino que, es el medio para definir un conjunto de datos y su estructura, para luego ser utilizada por otros programas que tienen como fin resolver un problema. Las razones más importantes para desarrollar una Ontología se describen a continuación [17].

1. Compartir un entendimiento común de la estructura de la información entre personas o agentes de software.

Supongamos que existen diferentes sitios Web que contienen información sobre salud o proveen servicios médicos de comercio electrónico. Si estos sitios Web comparten y usan la misma Ontología de términos que utilizan, entonces los agentes de software pueden extraer y agregar información de estos sitios Web distintos. Los agentes pueden usar la información agregada para responder consultas de usuarios o también puede ser usada como entrada para otra aplicación.

2. Permitir la reutilización del conocimiento de dominios.

Esto nos permite reutilizar Ontologías desarrolladas por otros investigadores e incluirlas en nuestro propio dominio. De esta forma si desarrollamos una gran Ontología, la podemos integrar utilizando distintas Ontologías que describirían una porción del gran dominio. También podemos utilizar una Ontología general y extenderla para describir el dominio de nuestro interés.

3. Hacer las especificaciones de un dominio explícitas.

Esto nos permitirá cambiar las especificaciones fácilmente si nuestro conocimiento del dominio cambia. Además estas son útiles para los nuevos usuarios quienes deben aprender lo que significan los términos en el dominio.

4. Separar el conocimiento del dominio del conocimiento operacional.

Podemos definir una tarea sobre un objeto de acuerdo a una especificación requerida e implementar un programa que realice esta tarea independientemente de las instancias de este objeto. Esto permitirá utilizar un mismo algoritmo para realizar una tarea en Ontologías diferentes.

5. Analizar el conocimiento del dominio.

Analizar el conocimiento es posible siempre y cuando exista una especificación declarativa de los términos disponibles. El análisis formal es valioso cuando uno intenta reutilizar y extender una Ontología existente.

2.4.3 Tipos de Ontologías

Se han identificado diferentes tipos de Ontologías, que se listan en la Tabla V, de acuerdo a la generalidad y el contexto donde se utilicen [18].

Tabla V. Tipos de Ontologías

Tipo	Descripción
Dominio	Capturan el conocimiento valido para un tipo de dominio particular. Por ejemplo: Ingeniería, Informática, Medicina.
Meta-datos	Son usadas para describir el contenido de la información.
Genéricas	Usadas para capturar conocimiento general sobre el mundo y proveen nociones básicas y conceptos para cosas como: espacio, tiempo y eventos.
Representativas	No se refieren a un dominio en particular y proveen entidades de representación que no definen lo que hay que representar.
Tareas o Métodos	Proveen términos específicos para una tarea en particular. Muestran un punto de vista razonable sobre el conocimiento de un dominio.
Aplicación	Encapsulan todo el conocimiento necesario para modelar un dominio en particular.

Fuente: D. Fensel. *Ontologies: Silver bullet for knowledge management and electronic commerce*. Springer-Verlag, Berlin, 2001

2.4.4 Aplicaciones comunes

Las Ontologías son utilizadas en la inteligencia artificial, la Web Semántica, en ingeniería de software, en la educación, en arquitectura de la información y muchas áreas más. También pueden ser utilizadas en diversas tareas para resolver problemas. Definitivamente el campo donde mayor aplicación tiene, es en las organizaciones donde utilizan las tecnologías de información de forma intensiva. Aplicaciones comunes de las Ontologías son los siguientes:

- Repositorios para la organización de conocimientos e información de tipo científico y corporativo.
- Herramienta para la adquisición de información, en situaciones en la que un equipo de trabajo la utiliza como soporte común para la organización del dominio.
- Permitir la reutilización del conocimiento existente, al desarrollar nuevos sistemas.
- Base para la construcción de lenguajes de representación del conocimiento

No existe un límite para utilizar Ontologías. Siempre que manejemos un conjunto de conceptos relacionados acerca de un área de conocimiento y necesitemos compartir esos conceptos con otras personas u organizaciones para trabajar en conjunto, entonces, ese es un punto de partida para utilizar las Ontologías.

2.4.5 Las Ontologías en la Web

Debido a que las Ontologías se están desarrollando sobre estándares comúnmente encontrados en la Web, es natural que la mayoría de aplicaciones que se le han encontrado a las Ontologías, residan en la Web [19].

2.4.5.1 Portales Web

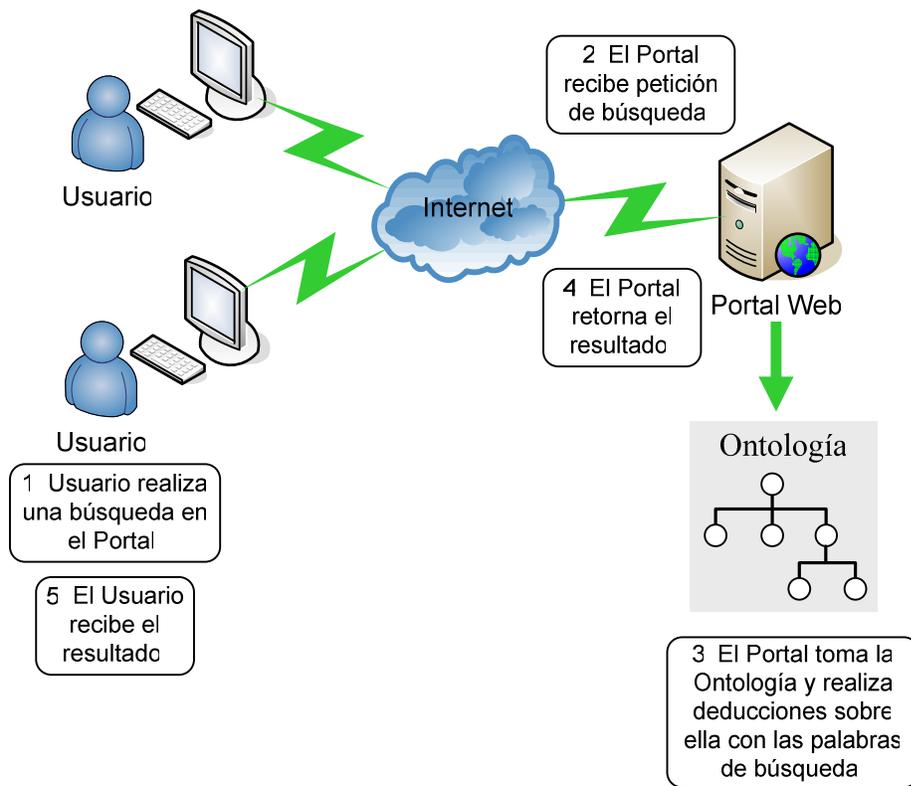
Como se mencionó en el capítulo 1, un portal Web es un sitio que provee información de contenido sobre un tópico común. Para que un portal sea exitoso, debe ofrecer contenidos interesantes. Regularmente este contenido es enviado por los miembros de la comunidad, quienes la indexan bajo subtemas. Sin embargo, un simple índice no provee a la comunidad con la habilidad suficiente para buscar el contenido que sus miembros requieren.

Para permitir búsquedas más inteligentes, los portales Web pueden definir una Ontología para su comunidad. Esta Ontología puede proveer toda la terminología usada para describir el contenido, así como axiomas que definen conceptos usando otros conceptos de la Ontología. Por ejemplo una Ontología podría definir los conceptos: *publicación*, *periódico*, *persona* y *autor*. También podría incluir definiciones como “todos los periódicos son publicaciones” o “*los autores de las publicaciones son personas*”. Cuando se combinan con hechos, estas definiciones permiten inferir hechos que son verdaderos. Estas inferencias pueden ofrecer a los usuarios del portal, resultados de búsqueda imposibles de obtener por medio de sistemas convencionales.

Un ejemplo de un portal basado en Ontologías es *OntoWeb* [20]. Este portal ayuda a la comunidad académica e industrial que está interesada en

investigación de Ontologías. En la figura 7, se puede apreciar el funcionamiento de un Portal Web mejorado, que utiliza Ontologías para mejorar las búsquedas.

Figura 7. Portal Web mejorado mediante el uso de Ontologías



2.4.5.2 Colecciones multimedia

Las Ontologías pueden ser utilizadas para proveer anotaciones semánticas a colecciones de imágenes, audio, video y otro tipo de objetos no basados en texto. Es aún más difícil para las computadoras extraer información semántica de recursos multimedia que hacerlo de texto de lenguaje natural. Es por esto que estos tipos de recursos son indexados con meta-etiquetas. Sin embargo, personas diferentes pueden describir estos objetos multimedia de formas diferentes, es importante que la búsqueda vaya más allá de simple comparación de palabras clave.

En este caso, las Ontologías pueden capturar conocimiento adicional acerca del dominio que puede ser utilizada para mejorar la recuperación de imágenes. Las Ontologías multimedia pueden ser de dos tipos:

- **Específicas del medio:** Estas Ontologías pueden tener taxonomías de diferentes tipos de medio y describir las propiedades de cada uno. Por ejemplo el audio puede incluir propiedades para identificar su longitud.
- **Específicas del contenido:** Estas Ontologías pueden describir el tema del recurso, por ejemplo, los participantes o interpretes.

Como ejemplo de una colección multimedia, consideremos un archivo de música clásica. Una Ontología de música clásica sería de mucha utilidad para hacer búsquedas en este archivo.

2.4.5.3 Administración de sitios Web corporativos

Las corporaciones grandes, normalmente poseen una numerosa cantidad de páginas Web que están relacionadas con cosas como: anuncios de prensa, productos, casos de estudio, procedimientos, comparaciones, documentación técnica y descripción de procesos. Las Ontologías pueden ser utilizadas para indexar estos documentos y proveer mejores medios de recuperación.

Aún cuando algunas corporaciones posean una taxonomía para organizar su información, esto no es suficiente. Una sola Ontología también es limitante porque las categorías son restringidas a un solo dominio. La habilidad de trabajar simultáneamente con varias Ontologías, incrementaría la riqueza de la descripción. Y es mejor aún, porque la habilidad de buscar con valores de diferentes parámetros es más útil que una búsqueda de palabras clave en una taxonomía.

Un sitio Web basado en Ontologías, podría resolver el problema que se presenta cuando un tipo de usuario realiza una búsqueda, pero el usuario no comparte la misma terminología que los autores del contenido buscado. Porque las personas usan términos distintos según su dominio, para nombrar los mismos conceptos. Sería realmente útil poseer una Ontología de términos para cada tipo de usuario, pero tener cada Ontología interrelacionada para que se pudieran hacer traducciones automáticas.

2.4.5.4 Documentación de diseño

Este sería el caso de un gran cuerpo de documentación de ingeniería, como el utilizado por la industria aeroespacial. Esta documentación podría ser de varios tipos diferentes, incluyendo documentación de diseño, documentación de elaboración y documentación de pruebas. Cada tipo podría tener su propia estructura jerárquica.

Las Ontologías pueden ser utilizadas para construir un modelo de información que permita la exploración del espacio de información en términos de los elementos representados, las asociaciones entre elementos, sus propiedades y vínculos a la documentación que la describe y define.

Un ejemplo concreto de esta aplicación, es la documentación de diseño del dominio del espacio aéreo, donde se incluyen los usuarios típicos:

- Ingeniero de mantenimiento, buscando toda la información relacionada con una parte en particular.
- Ingeniero de diseño, buscando las limitantes de reutilizar un ensamblaje particular.

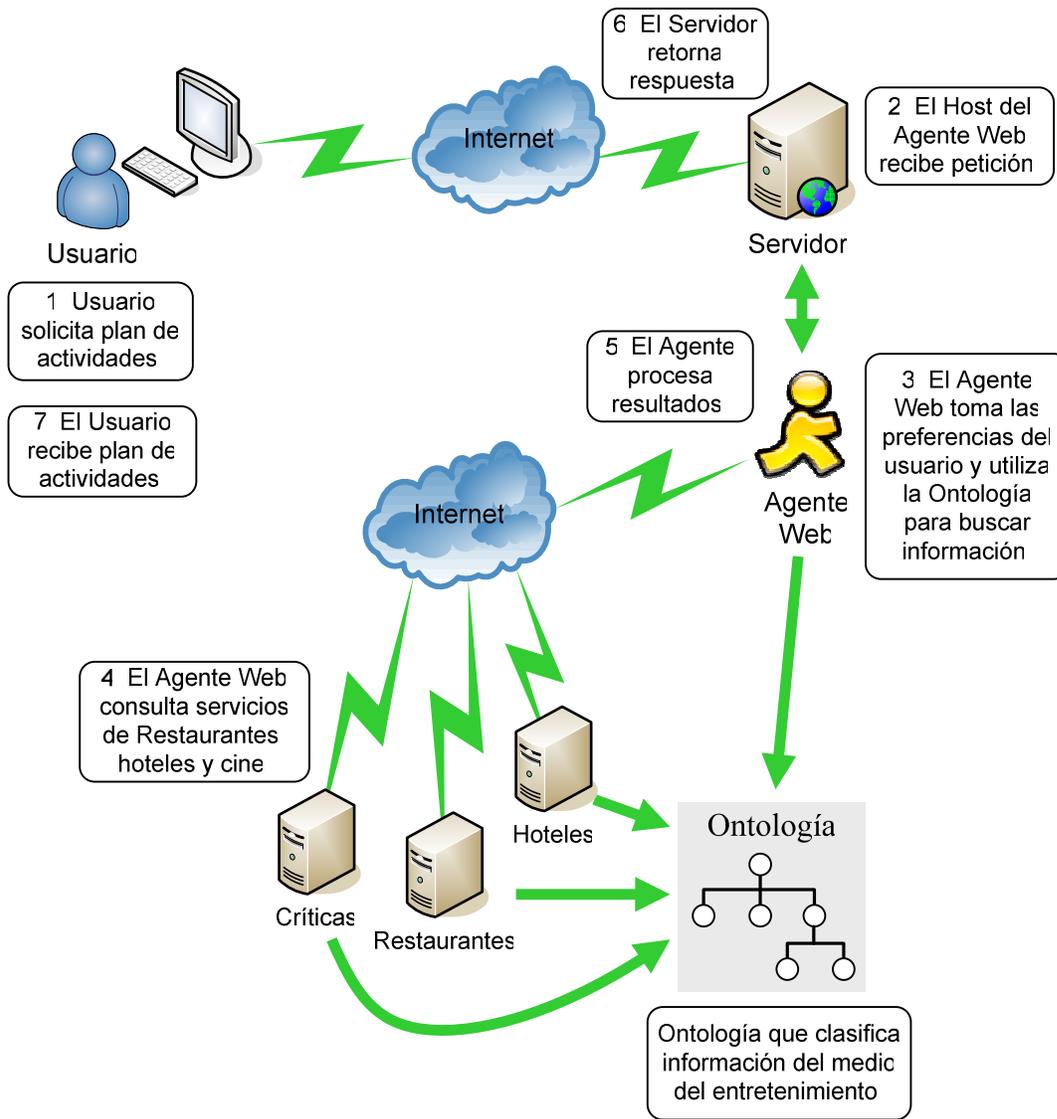
2.4.5.5 Agentes y servicios

La Web semántica puede proveer a los agentes la capacidad de entender e integrar diversos recursos de información. Un ejemplo específico es el de un planeador de actividades sociales, el cual puede tomar las preferencias de un usuario (por ejemplo el tipo de música que le gusta, comida favorita, etc.) y usar esta información para planear las actividades del usuario para un día. Esta tarea de planear actividades, dependerá de la riqueza del servicio ofrecido y de las necesidades del usuario. Durante este proceso, el agente puede consultar críticas y ratings para encontrar actividades que coincidan más con las preferencias del usuario (consultando críticas sobre películas y restaurantes para determinar el mejor).

Este tipo de agentes, requieren Ontologías de dominio que representen los términos de restaurantes, hoteles, cines, etc. Y Ontologías de servicio para representar los términos de los servicios. Estas Ontologías permitirán capturar información necesaria para que los agentes puedan discriminar y balancear entre las preferencias de los usuarios. Esta información puede ser proporcionada por varias fuentes, como los portales, sitios de servicios específicos, sitios de reservación y en general la Web.

Agentcities es un ejemplo de una iniciativa que explora el uso de agentes en un ambiente de servicios distribuidos a través de Internet [21]. Para hacer esto, será necesario construir una red de plataformas de agentes que representen ciudades reales o virtuales, como la ciudad capital o Antigua Guatemala, y poblarlas con los servicios de esas ciudades. Inicialmente estos servicios serán *B2C* (de negocio a consumidor), como hoteles, restaurantes, entretenimiento, etc. Pero eventualmente serán expandidos hasta incluir servicios *B2B* (de negocio a negocio).

Figura 8. Funcionamiento de un Agente Web de Servicios



En la Figura 8 puede apreciarse de forma visual el funcionamiento de un Agente Web que planea actividades sociales para sus usuarios. El Agente toma las preferencias del usuario y en base a esto consulta información Web debidamente clasificada mediante una Ontología. El objetivo del Agente es encontrar un resultado útil para el usuario.

2.4.5.6 Computación ubicua

Es un paradigma emergente de computación personal, caracterizada por el cambio de la computación dedicada, hacia capacidades permanentes de computación que residirán en nuestra vida diaria. La computación ubicua se compone de pequeños dispositivos inalámbricos portátiles. La presencia y la naturaleza inalámbrica de los dispositivos, requerirá arquitecturas de red que soporten configuraciones automáticas. La razón principal para desarrollar esta tecnología es que será dirigida a los consumidores ordinarios.

El objetivo de la computación ubicua es la interoperabilidad por accidente, es decir, que los dispositivos que no fueron necesariamente diseñados para operar juntos, sean capaces de descubrir la funcionalidad de cada uno y que puedan tomar ventaja de esto. Entender otros dispositivos y razonar acerca de su funcionalidad y servicios, es necesario para los diferentes escenarios de la computación ubicua, involucrando a docenas o cientos de dispositivos.

La interoperación de escenarios es dinámica (dispositivos apareciendo y desapareciendo en cualquier momento debido a que sus dueños se movilizan de un ambiente a otro) y no involucra a las personas en el proceso de configuración. Aquí el descubrimiento de servicios jugará un papel clave para la configuración automática, y se basará sobre estándares universales.

Para lograr la computación ubicua, se usará una Ontología que describirá las características de los dispositivos, el medio de acceso al dispositivo, la política de uso establecida por el usuario del dispositivo y otros requerimientos técnicos y restricciones que afectarán la incorporación del dispositivo a una red de computación ubicua.

2.5 Lenguajes para representar Ontologías

Existe una interacción directa entre las Ontologías y la Web Semántica. El rol de las Ontologías para la Web Semántica es tan importante que ha llevado a la creación de nuevas extensiones de los estándares Web con conceptos de lenguaje orientados a Ontologías.

La técnica de estructuración principal en la Web, es el etiquetado. Un lenguaje de etiquetado es una especificación de las etiquetas a usar, la forma de combinarlas y el significado de cada una. *HTML* es el lenguaje de etiquetado más popular y difundido en la Web, sin embargo solo indica como debe ser mostrada la información en el navegador y no define el significado del contenido.

2.5.1 Lenguajes tradicionales

2.6.1.1 *Ontolingua*

Es un lenguaje basado en *KIF* (*Knowledge Interchange Format*, formato de intercambio de conocimiento) y *FO* (*Frame Ontology*, marco de Ontología). Es el único lenguaje de definición de Ontologías usado por el servidor de *Ontolingua*. *KIF* fue desarrollado para resolver el problema de la heterogeneidad en los lenguajes de representación del conocimiento. Provee definición para objetos, funciones y relaciones. *KIF* tiene semánticas declarativas y está basado en cálculo de predicado de primer orden, con una notación prefijada. También ayuda a representar meta-conocimiento y reglas de razonamiento no monótonas [23].

Como *KIF* es un formato de intercambio, no es apto para usarse como especificación de Ontologías. *FO* es un medio de representación de conocimiento, que permite especificar una Ontología, siguiendo el paradigma de marcos, como clases, instancias, subclase-de, instancia-de, etc. Sin embargo *FO* no puede expresar axiomas, por lo que *Ontolingua* permite incluir expresiones *KIF* dentro de las definiciones basadas en *FO*. De esta forma con *Ontolingua* puede crear Ontologías de las siguientes formas:

- Usando solamente el vocabulario de *FO*.
- Usando expresiones *KIF*.
- Usando ambos lenguajes al mismo tiempo.

2.5.1.2 Protocolo *OKBC*

OKBC (*Open Knowledge Base Connectivity*, base de conectividad para el conocimiento abierto), es un protocolo, no un lenguaje. El protocolo es complementario para los lenguajes de especificación, desarrollados para compartir el conocimiento [23].

Existe un modelo llamado *GFP*, usado para representar formalismos que funciona debajo de *OKBC*, este permite una representación de conocimiento centrada en objetos, y ofrece medios de representación comúnmente encontrados en lenguajes de representación de Ontologías: constantes, marcos, slots, facetas y clases.

Además provee una interfaz completa que permite describir operaciones a ser realizadas en una base de conocimiento que es accedida en una red mediante el protocolo *OKBC*.

2.5.1.3 OCML

OCML (*Operational Conceptual Modeling Language*, lenguaje de modelado conceptual y operacional) es un lenguaje basado en marcos que provee el mecanismo de expresar relaciones, funciones, reglas (con encadenamiento hacia delante y atrás), clases e instancias [23].

Para lograr una ejecución eficiente del lenguaje, se agregaron mecanismos de lógica extra que logran un razonamiento eficiente. *OCML* es compatible con *Ontolingua*, puede considerarse un “*Ontolingua* operacional” y provee comprobación de teoremas y evaluación de funciones para sus declaraciones.

2.5.1.4 FLogic

FLogic es un acrónimo para *Frame Logic*. Su objetivo es integrar lenguajes basados en marcos con el cálculo de predicados de primer orden. Es un lenguaje limpio y declarativo que permite declarar la mayoría de los aspectos de los lenguajes orientados a objetos y basados en marcos, como: entidades de objetos, objetos complejos, herencia, polimorfismo, métodos y encapsulamiento [23].

Las aplicaciones de *FLogic* van desde bases de datos deductivas y orientadas a objetos hasta las Ontologías, y puede ser combinado con otras lógicas especializadas para mejorar el razonamiento de la información de las Ontologías.

2.5.1.5 LOOM

LOOM es un ambiente y lenguaje de programación de alto nivel dedicado a construir sistemas expertos y otras aplicaciones de inteligencia. Está basado en la lógica descriptiva, logrando una integración fuerte de los paradigmas basados en reglas y basados en marcos [23].

LOOM soporta un lenguaje de descripción para modelar objetos y sus relaciones. También soporta un lenguaje para especificar restricciones sobre los conceptos y relaciones, soportando afirmaciones sobre los conceptos. El acercamiento a la lógica descriptiva para modelar Ontologías, difiere de otros lenguajes. Las definiciones escritas con este lenguaje, tratan de explotar la existencia de clasificadores poderosos en el lenguaje, especificando conceptos mediante el uso de restricciones sobre ellos.

2.5.2 Lenguajes basados en la Web

2.5.2.1 XOL

XOL (*XML-Based Ontology Exchange Language*, lenguaje de intercambio de Ontologías basado en *XML*) fue diseñado para proveer un formato de intercambio de definiciones de Ontologías entre un grupo interesado de participantes. Por lo que no está orientado al desarrollo de Ontologías, sino como un lenguaje intermedio para transferir Ontologías entre sistemas de bases de datos diferentes, herramientas de desarrollo de Ontologías o programas de aplicación.

XOL permite definir un subconjunto de *OKBC* en sintaxis *XML*, llamado *OKBC-Lite*. Como *OKBC* define un protocolo para acceder sistemas de conocimiento basados en marcos, *XOL* puede ser útil para intercambiar información entre sistemas diferentes a través de la Web.

2.5.2.2 SHOE

SHOE (*Simple HTML Ontology Extensión*, extensión simple de Ontologías para *HTML*) fue desarrollado en la Universidad de *Maryland*, con la idea de incorporar semánticas entendibles por máquinas en el *HTML*. Recientemente, *SHOE* fue adaptado a la sintaxis de *XML*, permitiendo a los agentes recuperar información relevante de las páginas Web y documentos, mejorando los métodos de búsqueda y recuperación de la información.

El proceso de búsqueda con *SHOE*, consisten en tres pasos: definir la Ontología, anotar las páginas *HTML* con información Ontológica para describirse a sí mismas y otras páginas, y lograr que un agente recupera la información semántica, buscando todas las páginas existentes y manteniendo la información actualizada.

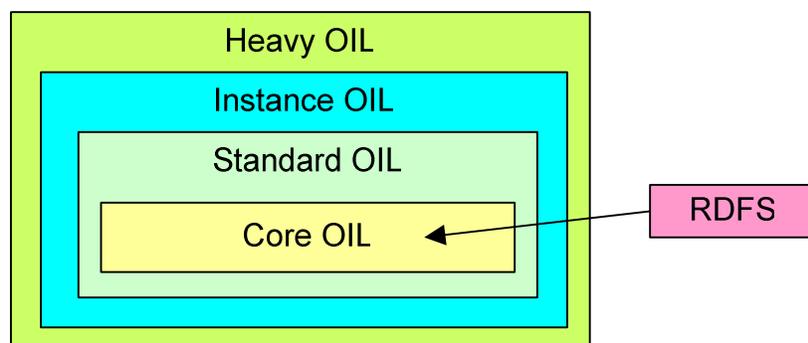
2.5.2.3 OIL

OIL (*Ontology Inference Layer*, capa de inferencia de Ontologías) es un lenguaje mejorado, basado en lógicas descriptivas que permite la interoperabilidad entre recursos Web. *OIL* es un estándar para especificar e intercambiar Ontologías cuya sintaxis y semántica se basa en estándares existentes: *OKBC* (*Open Knowledge Base Connectivity*) que es una *API* para

acceder sistemas de representación del conocimiento basados en marcos (*XOL*, *RDF*). El lenguaje es intuitivo para los usuarios humanos y provee poder de expresión adecuado para aplicaciones reales. *OIL* presenta una arquitectura de capas, cada capa agrega funcionalidad y complejidad a la capa anterior [24].

- **Core OIL:** coincide con *RDF Schema*. Esto significa que hasta el agente RDFS más simple puede procesar Ontologías *OIL*.
- **Standard OIL:** es un lenguaje orientado a capturar los primitivos suficientes para dar expresividad y permite que las semánticas sean especificadas precisamente.
- **Instance OIL:** incluye una integración exhaustiva que permite capacidades completas de bases de datos.
- **Heavy OIL:** puede incluir capacidades de representación y razonamiento adicionales.

Figura 9. Estructura en capas del lenguaje *OIL*



Fuente: *OIL. Ontology Inference Layer*. Disponible: <http://www.ontoknowledge.org/oil/oil.html>

2.5.2.4 DAML + OIL

Mientras que *RDFS* provee clasificación primitiva y reglas simples para el conocimiento, *DAML+OIL* (*DARPA Agent Markup Language + OIL*) va mucha más allá. *DAML+OIL* fue desarrollado sobre *RDFS* y posee una fuerte relación con *OIL*. Utilizando los *namespace* de *RDF* para organizar y asistir en la integración de varias Ontologías incompatibles, *DAML+OIL* es el punto de partida de estandarización del *W3C*. Sus principales adiciones al *RDFS* son semánticas formales, basados en lógica descriptiva y modelado de primitivos avanzados, como expresiones booleanas y algunos axiomas.

2.5.2.5 OWL

El *OWL* (*W3C Web Ontology*), es el lenguaje semántico estándar de etiquetado propuesto por el *W3C* (*World Wide Web Consortium*) para publicar y compartir Ontologías basadas en Internet [25].

También se basa en *RDF* y facilita grandemente a las máquinas comprender el contenido Web, mediante vocabularios adicionales para la descripción de términos.

El lenguaje *OWL* provee tres sub-lenguajes de expresión incremental diseñados para ser usados por comunidades específicas de implementadores y usuarios:

- **OWL Lite:** orientado a los usuarios que buscan una jerarquía de clasificación y características simples de limitación. Solo soporta cardinalidad de 0 y 1.

- **OWL DL:** orientado a los usuarios que desean máxima expresividad sin perder capacidad de cómputo. Esta construido sobre lógicas descriptivas, que estudian las lógicas de primer orden.
- **OWL Full:** ideal para los usuarios que desean máxima expresividad y la libertad de sintaxis de *RDF* sin garantías en el poder de cómputo. Permite aumentar el significado del vocabulario predefinido (*RDF*, *OWL*).

2.6 Herramientas

Desarrollar Ontologías a mano es una tarea muy difícil que consume mucho tiempo. Es imposible derivar Ontologías de datos complejos de esta forma. Por eso se necesitan herramientas avanzadas que permitan explotar esta tecnología.

Los editores de Ontologías facilitan la vida de los ingenieros del conocimiento, ayudándolos a modelar y construir Ontologías. Los editores permiten definir jerarquías de conceptos, definir atributos para los conceptos, definir axiomas y restricciones sobre éstos.

2.6.1 Protégé

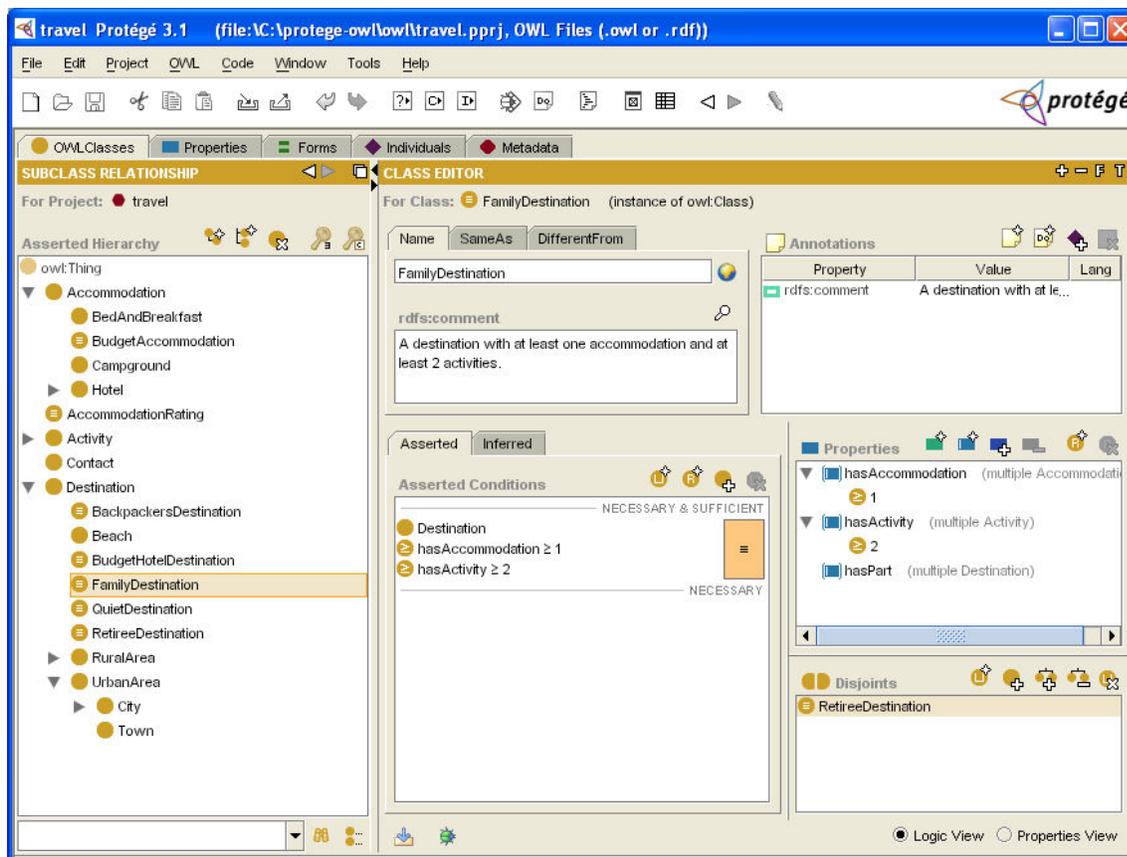
Protégé es una herramienta gratuita de código abierto, que permite crear y editar Ontologías y bases de conocimiento [26]. Fue desarrollada en la Universidad de *Stanford* y está basada sobre la plataforma Java. Implementa un conjunto rico de estructuras para modelar conocimiento y acciones que soportan la creación, visualización y manipulación de Ontologías en varios

formatos de representación. Además puede ser extendido mediante una arquitectura de plugins y un API basado en Java.

Actualmente existen dos formas de editar Ontologías: *Protégé-Frames* para editar Ontologías basadas en marcos de acuerdo al protocolo OKBC y *Protégé-OWL* para crear Ontologías destinadas a la Web Semántica (basadas en el lenguaje OWL promovido por el W3C).

En la Figura 10, se puede apreciar el visor de clases de *Protégé*.

Figura 10. Interfaz gráfica de la herramienta *Protégé*



Fuente: <http://protege.stanford.edu>

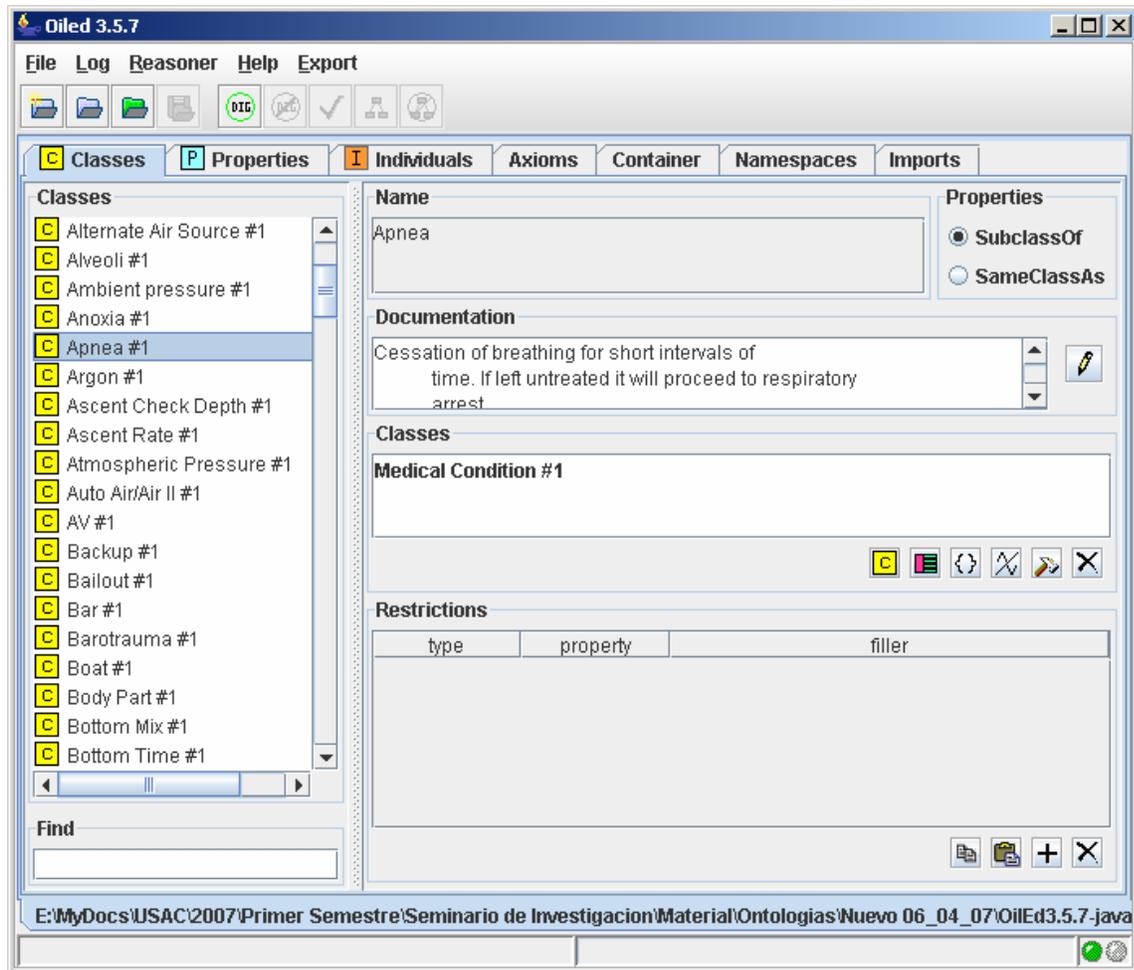
2.6.2 *OILEd*

Es otra herramienta basada en la plataforma Java, desarrollada por la Universidad de *Manchester* del Reino Unido. El modelo de conocimiento usado por *OILEd* es una extensión de los sistemas basados en marcos, y permite lenguajes altamente expresivos. Sin embargo, no soporta el desarrollo de Ontologías de gran escala, ni el manejo de versiones, migración e integración de Ontologías [27].

Los lenguajes soportados por este editor son: *DAML+OIL*, *RDF*, *RDFS* y *SHIQ* para clasificar. Una de las grandes características de *OILEd* es el *FACT Reasoner* (razonador de hechos), que es una interfaz de *CORBA* escrita en lenguaje *LISP*, que ayuda a verificar las consistencias de las Ontologías traduciendo de *DAML+OIL* a *SHIQ DL*.

En la Figura 11, puede apreciarse los componentes de la herramienta *Oiled* para crear Ontologías.

Figura 11. Pantalla principal de la herramienta *Oiled* v3.5.7



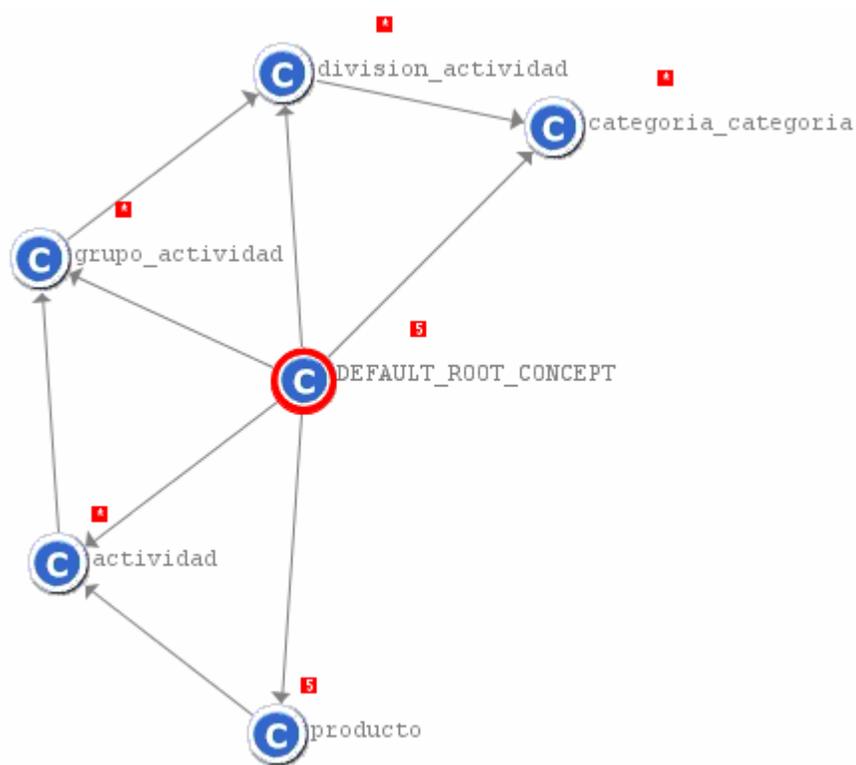
Fuente: <http://oiled.man.ac.uk/download/>

2.6.3 *OntoEdit*

Es una herramienta gráfica que soporta el desarrollo y mantenimiento de Ontologías. Está disponible en versión sencilla o en cliente/servidor. El modelo de Ontología interna en el que se basa, permite el desarrollo de dominios a través de clases, relaciones, axiomas, facetas y atributos. La herramienta es fácilmente extensible utilizando *plugins* y permite al usuario importar y exportar Ontologías a distintos formatos. Naturalmente soporta *DAML+OIL* y *RDFS*.

En la figura 12 se puede apreciar una Ontología que define productos y servicios. Esta es la misma Ontología que utilizaremos más adelante para desarrollar la aplicación *Peer-to-Peer*.

Figura 12. Ontología de productos y servicios visualizada en *OntoEdit*



2.7 Pasos para crear una Ontología

No existe una forma correcta de crear una Ontología, debido a que el diseño dependerá mucho de la persona que la desarrolle, de su educación, de sus experiencias y de la forma como él mire el mundo. Sin embargo hay ciertos pasos que nos pueden guiar en la estructuración de los conceptos del dominio que nos interese. Antes de empezar, existen tres reglas que nos ayudaran a tomar decisiones a la hora de diseñar una Ontología [17].

1. No existe una única forma correcta de modelar un dominio. Siempre hay alternativas. La mejor solución siempre depende de la aplicación que queremos construir y de las extensiones que se anticipan.
2. El desarrollo de Ontologías es un proceso iterativo, esto significa que se debe revisar una y otra vez para enriquecer la Ontología.
3. Los conceptos de la Ontología deberían acercarse a objetos (físicos o lógicos) y a las relaciones del dominio de interés. Deben haber nombres (objetos) o verbos (relaciones) en las sentencias que describan el dominio.

Esta reglas se refieren a decidir el objetivo de la Ontología y que tan detallada o general se desea.

2.7.1 Determinar el dominio y alcance de la Ontología

Se sugiere empezar el desarrollo de una Ontología, definiendo su dominio y alcance. Para determinar esto, hay que responder las siguientes preguntas.

- ¿Cuál es el dominio que la Ontología cubrirá?
- ¿Para qué se va a utilizar la Ontología?
- ¿Qué tipo de preguntas deberá responder la información de la Ontología?
- ¿Quién va a usar y mantener la Ontología?

Las respuestas a estas preguntas pueden cambiar durante el proceso de diseño, pero en cualquier momento pueden ayudar a limitar el alcance del modelo.

Considerando la Ontología mostrada en la figura 12. El dominio de la Ontología es representar productos y servicios. En la aplicación P2P, esta Ontología es utilizada para realizar búsquedas precisas de los productos y servicios que ofrecen las Pequeñas Y Medianas Empresas (PYMES).

2.7.2 Reutilizar Ontologías existentes

Siempre vale la pena considerar lo que alguien más ha realizado y verificar si se puede refinar y extender las fuentes para nuestro dominio y tarea particular. Existen muchas Ontologías disponibles en formato electrónico que pueden ser importadas a una herramienta de desarrollo de Ontologías como las mencionadas en el inciso 2.5.

Tabla VI. Listado de Ontologías reutilizables disponibles en la Web

Nombre	URL	Tipo de Sitio
Ontolingua	http://www.ksl.stanford.edu/software/ontolingua	Librería
DAML	http://www.daml.org/ontologies	Librería
UNXPSC	www.unspsc.org	Comercial
RosettaNet	www.rosettanet.org	Comercial
DMOZ	www.dmoz.org	Comercial

Fuente: Natalya F. Noy, Deborah L. McGuinness. *Ontology Development 101: A Guide to Creating Your First Ontology*. Stanford University.

2.7.3 Enumerar los términos de la Ontología

Es útil escribir una lista de todos los términos de los que nos gustaría hablar, que nos gustaría decir de ellos y determinar los atributos de esos términos. Es importante obtener una lista exhaustiva de términos sin importar el traslape de los conceptos que representan, relaciones entre los conceptos, o cualquier propiedad que los conceptos tengan, las relaciones entre ellos o las propiedades que los conceptos puedan tener.

2.7.4 Definir las clases y su jerarquía

Existen varios métodos para desarrollar la jerarquía de clases.

- *Arriba-abajo*: Se inicia con la definición de los conceptos más generales del dominio hasta llegar a conceptos más específicos
- *Abajo-arriba*: Se inicia con la definición de los conceptos más específicos de la jerarquía hasta agruparlos en grupos más generales de conceptos.
- *Combinada*: Es una combinación de ambos métodos. Se puede iniciar con unos conceptos generales y luego unos específicos. Luego se pueden relacionar en un nivel intermedio.

Ninguno de los tres métodos es mejor que otro. El método a seguir depende del punto de vista del diseñador sobre el dominio a desarrollar.

2.7.5 Definir las propiedades de las clases (slots)

Solo las clases no proveen suficiente información para responder las preguntas que debe resolver la Ontología. Una vez definidas algunas clases, debemos describir la estructura interna de los conceptos. Las propiedades que se obtuvieron de cada concepto se convertirán en *slots* que serán adjuntados a la clase. Como los slots pueden ser heredados por las subclases, es necesario colocarlos en la clase más general.

2.7.6 Definir las facetas de los slots

Los slots pueden tener diferentes facetas que describen el tipo de valor, valores permitidos, número de valores (cardinalidad) y otras características únicas de los slots.

- *Cardinalidad*: Define cuantos valores puede tener un slot.
- *Tipo-valor*: Define el tipo de valor que puede colocarse en un slot. Los tipos más comunes son: Cadena, Número, Booleano, Enumerado e Instancia.
- *Rango y Dominio*: Son las clases que son permitidas para un slot de tipo Instancia.

2.7.7 Crear instancias

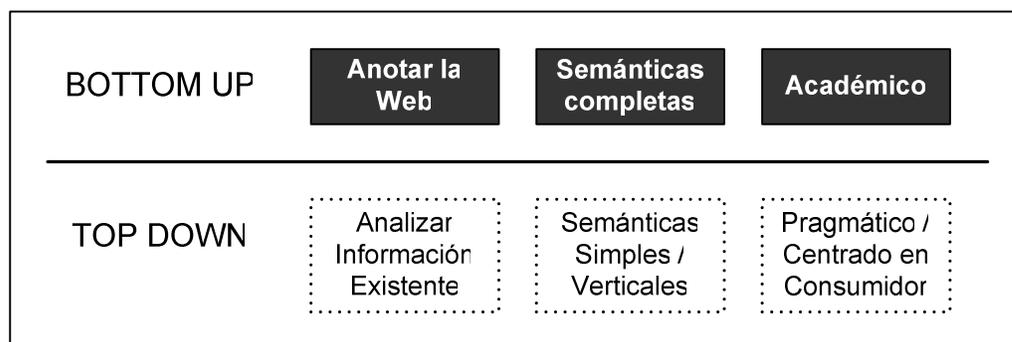
El último paso es crear instancias individuales de clases en la jerarquía. Definir una instancia de una clase requiere: escoger una clase, crear una instancia individual de la clase y llenar los valores de los slots.

2.8 Aplicaciones semánticas actuales

Antes que nada, definamos el término “Aplicación semántica”. Las aplicaciones semánticas tratan de determinar el significado del texto, de otros datos y luego crear conexiones sobre éstos para los usuarios [28]. Actualmente existen dos formas de implementar la Web Semántica [29].

- **Bottom up:** involucra encapsular anotaciones semánticas (meta-datos) en la información.
- **Top down:** se basa en analizar la información existente y elaborar la infraestructura necesaria para soportarla.

Figura 13. Formas de implementar la Web Semántica



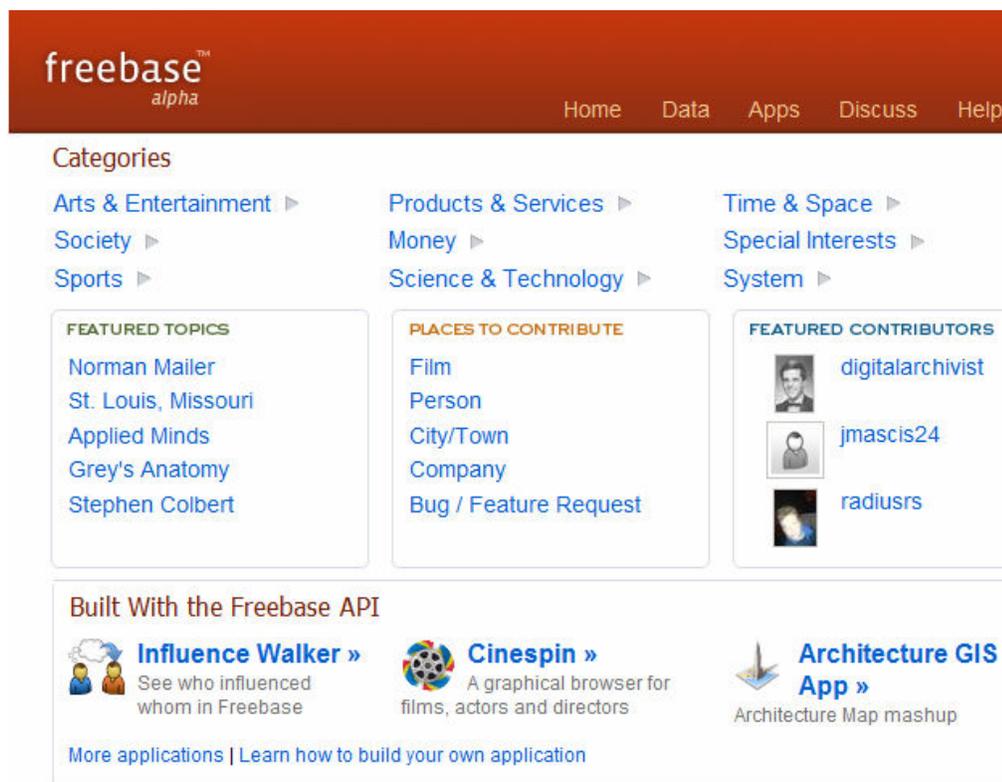
Fuente: Alex Iskold. *Top-Down: A New Approach to the Semantic Web*. Disponible: http://www.readwriteweb.com/archives/the_top-down_semantic_web.php

A continuación vamos a ver algunas de las aplicaciones más recientes que utilizan elementos de la Web Semántica para proveer soluciones que no serían posibles utilizando la Web actual.

2.8.1 Freebase

Freebase es una base de datos que contiene todo tipo de datos y provee una API para manipularlos. Debido a que es una base de datos abierta, cualquiera puede ingresar nueva información dentro de *Freebase*. Una página dentro de *Freebase* es similar a una página en Wikipedia. Al ingresar nuevos datos, la aplicación puede hacer sugerencias acerca del contenido. Los temas en *Freebase* son organizados por tipo, y es posible conectar las páginas con vínculos y etiquetas semánticas. En general *Freebase* trata sobre datos compartidos y lo que se puede hacer con ellos. En la figura 14 puede apreciarse la página principal de *Freebase* [30].

Figura 14. Página principal de la base de datos *Freebase*



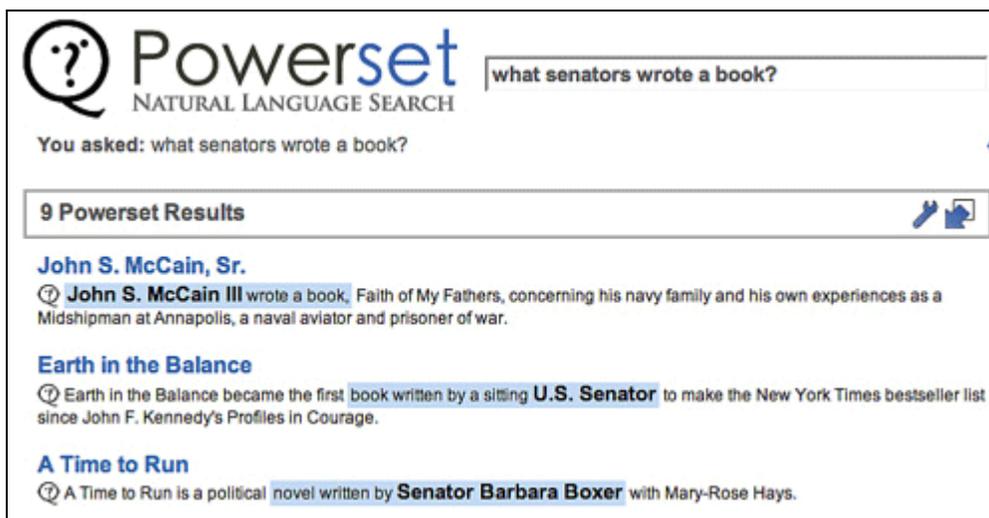
The screenshot shows the Freebase website homepage. At the top, there is a dark red header with the 'freebase alpha' logo on the left and navigation links for 'Home', 'Data', 'Apps', 'Discuss', and 'Help' on the right. Below the header, the 'Categories' section lists various topics such as 'Arts & Entertainment', 'Society', 'Sports', 'Products & Services', 'Money', 'Science & Technology', 'Time & Space', 'Special Interests', and 'System'. The main content area is divided into three columns: 'FEATURED TOPICS' (listing Norman Mailer, St. Louis, Missouri, Applied Minds, Grey's Anatomy, and Stephen Colbert), 'PLACES TO CONTRIBUTE' (listing Film, Person, City/Town, Company, and Bug / Feature Request), and 'FEATURED CONTRIBUTORS' (listing digitalarchivist, jmascis24, and radiusrs). At the bottom, there is a section titled 'Built With the Freebase API' featuring three applications: 'Influence Walker' (See who influenced whom in Freebase), 'Cinespin' (A graphical browser for films, actors and directors), and 'Architecture GIS App' (Architecture Map mashup). A link for 'More applications | Learn how to build your own application' is provided at the bottom of this section.

Fuente: www.freebase.com

2.8.2 Powerset

Es un motor de búsqueda que acepta lenguaje natural para realizar las consultas. El sistema se basa en tecnologías semánticas que solo han estado disponibles en los años recientes. Puede realizar conexiones semánticas, lo que ayuda a crear la base de datos semántica. La idea es que *Powerset* extrae el significado y el conocimiento automáticamente del texto ingresado. Este producto todavía no es público, pero está en proceso de desarrollo. En la figura 15 se puede apreciar el resultado de una búsqueda realizada con *Powerset* [31].

Figura 15. Ejemplo de una búsqueda realizada en *Powerset*



The screenshot shows the Powerset search engine interface. At the top left is the Powerset logo, which consists of a question mark inside a speech bubble, followed by the text "Powerset" and "NATURAL LANGUAGE SEARCH" below it. To the right of the logo is a search input field containing the text "what senators wrote a book?". Below the search field, it says "You asked: what senators wrote a book?". Underneath that is a section titled "9 Powerset Results" with a small icon of a hand pointing to a document. The results are listed as follows:

- John S. McCain, Sr.**
① **John S. McCain III** wrote a book, *Faith of My Fathers*, concerning his navy family and his own experiences as a Midshipman at Annapolis, a naval aviator and prisoner of war.
- Earth in the Balance**
① *Earth in the Balance* became the first book written by a sitting **U.S. Senator** to make the New York Times bestseller list since John F. Kennedy's *Profiles in Courage*.
- A Time to Run**
① *A Time to Run* is a political novel written by **Senator Barbara Boxer** with Mary-Rose Hays.

Fuente: <http://www.readwriteweb.com/>

2.8.3 Twine

“*Twine* es una nueva forma revolucionaria de compartir, organizar y encontrar información” [32]. *Twine* dice ser la primera aplicación de la Web Semántica para las masas. *Twine* automáticamente aprende acerca de las personas y sus intereses conforme se va poblando contenido en la aplicación. Al ingresar nuevos datos, *Twine* escoge cierto contenido y lo marca con etiquetas semánticas, por ejemplo el nombre de una persona, y crea nueva información semántica. Además han aplicado un poco de *machine learning* con Wikipedia, para aprender acerca de nuevos conceptos, su fundador Nova Spivack, ha comparado su invención *Twine* con Google, diciendo: “es una generación *bottom-up* generada por el usuario de la Web” [28].

Figura 16. Página de introducción a la aplicación *Twine*



Fuente: www.twine.com

2.8.4 Hakia

Hakia es un motor de búsqueda con bastante futuro. Utiliza métodos para procesar consultas elaboradas en lenguaje natural y tratar de entregar resultados de búsqueda significativos. *Hakia* intenta analizar los conceptos de una consulta, al realizar análisis de sentencias. Actualmente la mayoría de motores de búsqueda, incluyendo Google, analizan palabras clave. Según la compañía, el futuro de los motores de búsqueda irá más allá del análisis de palabras clave, los motores de búsqueda hablarán de regreso con la persona y en efecto se convertirán en asistentes de búsqueda. Actualmente tienen una versión beta de su buscador disponible en la Web [33]. *Hakia* utiliza dos tecnologías principalmente: *QDEX* (*Query Detection and Extraction*, o detección y Extracción de Consultas); este componente realiza el trabajo fuerte de levantar la consulta y analizarla al nivel de sentencia. Y el Algoritmo *SemanticRank*; que utiliza semánticas que relacionan unos conceptos con otros.

Figura 17. Pantalla principal del motor de búsqueda *hakia*

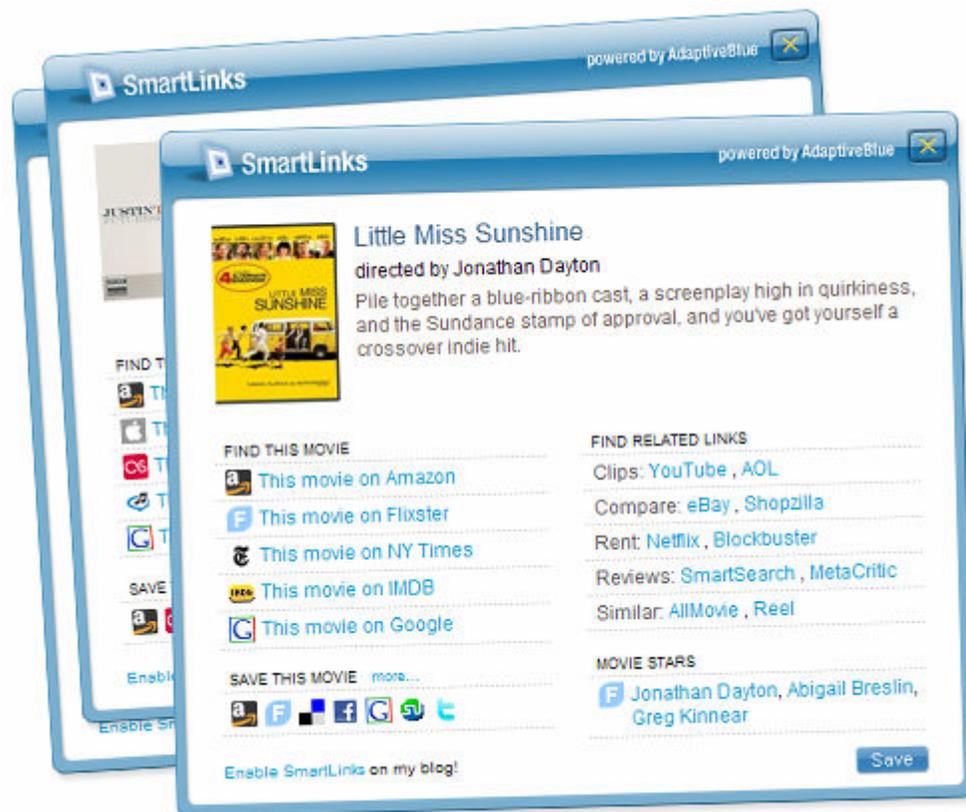


Fuente: www.hakia.com

2.8.5 AdaptiveBlue

AdaptiveBlue ha lanzado una nueva versión de su producto *SmartLinks*, que permite a publicadores de sitios Web agregar hipervínculos cargados semánticamente a su sitio web [34]. *SmartLinks* agregan una capa de información contextual a ciertos tipos de enlaces, incluyendo hipervínculos a libros, películas, música, acciones y vino. *SmartLinks* entiende los hipervínculos y los encapsula con información adicional, toma información sin estructura y la convierte en información estructurada al entender un elemento básico de la Web y agregarle semántica. En la figura 18 se puede apreciar un ejemplo de *SmartLinks*.

Figura 18. Ejemplo de un hipervínculo mejorado con *SmartLinks*

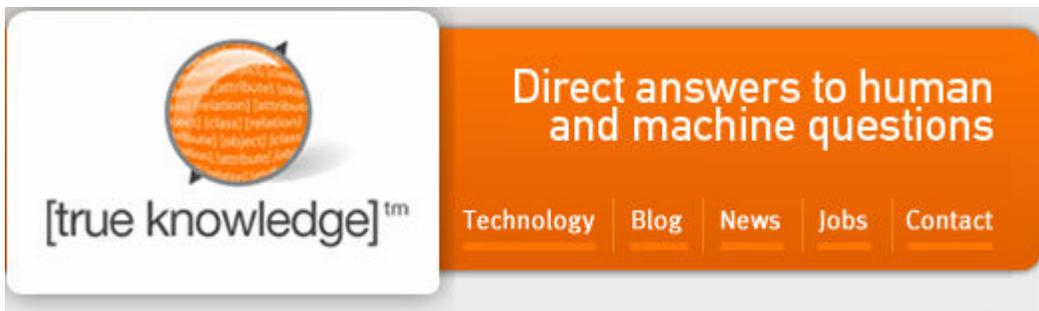


Fuente: www.adaptiveblue.com

2.8.6 *TrueKnowledge*

TrueKnowledge es un motor de búsqueda semántico, que funciona desde noviembre 2007. *TrueKnowledge* combina análisis de lenguaje natural, una base de conocimiento interna y bases de datos externas para ofrecer respuestas inmediatas a las consultas realizadas por los usuarios [35]. En lugar de indicar simplemente páginas Web donde el motor de búsqueda cree que uno podrá encontrar la respuesta, ofrece una respuesta explícita y muestra la ruta de razonamiento utilizada para llegar a esa respuesta. Su fundador William-Pedoe, mencionó que ha estado trabajando en el software durante los últimos 10 años [28]. En la figura 19, se puede apreciar el encabezado del sitio Web oficial de *TrueKnowledge*.

Figura 19. Encabezado del sitio Web oficial de *TrueKnowledge*



Fuente: www.trueknowledge.com

2.8.7 *Triplt*

Triplt es una aplicación que administra tus planes de viaje. Al planificar un viaje, uno envía sus reservaciones al correo plans@tripit.com, y el sistema maneja el resto [36]. Su tecnología pendiente de patente llamado *iterator* es

un paso más rumbo a la Web Semántica, ya que extrae información útil de estos correos y genera un plan de viaje bien organizado y estructurado. *TripIt* obtiene información de Wikipedia para los lugares que uno visita. Utiliza microformatos, por ejemplo el formato *iCal*, el cual está bien integrado en software de manejo de calendarios.

La compañía recalca que “en lugar de imprimir 20 páginas de planeación, solo debes imprimir 3 páginas, y lo demás es realizado automáticamente para uno” [28]. Entre sus planes futuros desean crear un motor de recomendaciones que pueda indicar a las personas a donde ir y a quien conocer. En la figura 20, se pueden apreciar los pasos necesarios para utilizar el agente *TripIt*.

Figura 20. Pasos básicos para utilizar el agente *TripIt*

The image shows the TripIt website interface. At the top left is the TripIt logo with 'BETA' and the tagline 'Organize your travel'. To the right are links for 'Login | Sign-up | Help/FAQ | Feedback'. Below the logo is a navigation bar with 'Home', 'MyTrips', 'MyFriends', and 'TripSearch'. A 'NEW Mobile Access with TripIt To Me!' banner is on the right. The main content area is titled 'How TripIt Works' and features three numbered steps:

- 1** Just email TripIt your travel plans—no matter where you booked. (Illustrated with icons for airplane, car, and hotel pointing to an envelope icon).
- 2** TripIt builds you a master itinerary with all your plans and more. (Illustrated with a document icon containing icons for airplane, globe, hotel, car, and fork/knife).
- 3** With TripIt, it's easy to share, print and access your itinerary from anywhere. (Illustrated with icons for a smartphone, printer, and a group of people, all connected to a document icon).

Below the steps, it says 'Just forward your travel confirmation emails to:' followed by the email address plans@tripit.com in a white box. To the right is an orange 'Learn More' button.

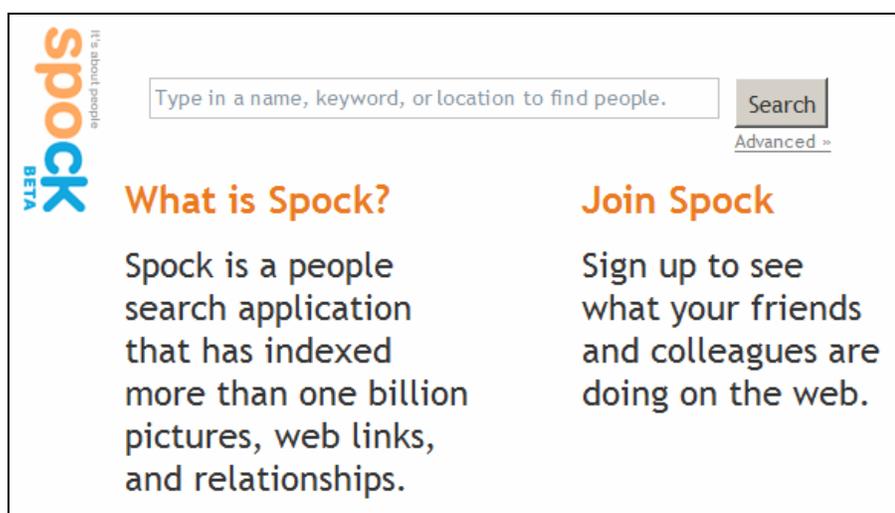
Fuente: www.tripit.com

2.8.8 Spock

Spock es un motor de búsqueda de personas, este motor contiene cuatro elementos que hace especial a su solución: (1) La perspectiva centrada en personas de las consultas. (2) Conjunto rico de atributos que caracterizan a las personas, por ejemplo: geografía, fecha de nacimiento, ocupación, etc. (3) Uso de etiquetas como vínculos o relaciones entre personas. (4) Mecanismo de auto-corrección alimentado por las personas [37].

Spock es un motor de búsqueda vertical, que conoce atributos importantes que las personas poseen, por ejemplo: nombre, edad, género, ocupación y ubicación. La parte más interesante de *Spock*, es su utilización de etiquetas, su tecnología permite rastrear todas las frases que son utilizadas frecuentemente para convertirlas en etiquetas, junto con el poder de creación de etiquetas de los usuarios, hacen a esta aplicación muy buena para encontrar personas. En la figura 21, se puede apreciar la versión beta de este motor de búsqueda.

Figura 21. Versión beta del motor de búsqueda *Spock*



Fuente: www.spock.com

3. LAS REDES *PEER TO PEER*

3.1 Introducción

La tecnología *Peer-to-Peer* (*P2P*, punto a punto) permite a cualquier dispositivo conectado a la red, dar servicios a otro dispositivo conectado a la red. Un dispositivo en una red *P2P* puede proveer acceso a cualquier tipo de recurso que tenga a disposición, por ejemplo documentos, capacidad de almacenamiento, poder de procesamiento o hasta su operador humano. La tecnología *P2P* es una extensión natural de la filosofía de Internet a través de la descentralización y tiene la capacidad de dar poder a una nueva clase de aplicaciones robustas al esparcir recursos a través de Internet [38].

La mayoría de los servicios en Internet son distribuidos usando la arquitectura tradicional cliente/servidor. En esta arquitectura, los clientes se conectan al servidor usando un protocolo común de comunicación, como el *http*, para tener acceso a un recurso específico. La mayor parte del procesamiento necesario para entregar el servicio al cliente ocurre del lado del servidor, dejando al cliente sin carga. Las aplicaciones más populares de Internet utilizan este modelo de entrega de servicios, incluyendo a *WWW*, *FTP*, telnet y correo.

Lamentablemente, esta arquitectura tiene una gran desventaja. Al incrementar el número de clientes, también aumenta la carga y la demanda de ancho de banda en el servidor, lo que no permite atender clientes adicionales. La ventaja de esta arquitectura es que requiere menos poder de procesamiento del lado del cliente. Irónicamente, la mayoría de usuarios han sido persuadidos

a actualizar sus computadoras a niveles superiores necesarios para las aplicaciones de Internet más usadas: navegar la Web y leer correo electrónico.

El cliente en la arquitectura cliente/servidor, tiene un rol pasivo, capaz de demandar servicios a un servidor pero incapaz de proveer servicios a otros clientes. Si todas las máquinas de la red corrieran un cliente y servidor a la vez, tendríamos la base de una red *P2P*.

Al crecer Internet, la cantidad finita de direcciones *IP*, motivo a los proveedores de Internet a dar direcciones *IP* dinámicamente a las máquinas que se conectaban a la red a través de conexiones de discado, esto les impide correr servidores útiles, ya que los clientes deben conocer la dirección *IP* antes. Es por esto, que los servicios son centralizados en servidores con direcciones *IP*. Otra razón que impide a los clientes ser servidores, es que si forman parte de una red privada, que por lo regular es aislada de Internet con un firewall, que es un dispositivo diseñado para prevenir conexiones arbitrarias hacia adentro y afuera de la red privada.

Consideremos la cantidad de almacenamiento y poder de procesamiento que estas máquinas cliente representan. Asumamos que solamente 10 millones de computadoras están conectadas a Internet en cualquier momento, cada una con un procesador de 100 *Mhz*, 100 *MB* de almacenamiento disponible, 1000 *bps* de ancho de banda y 10% de poder de procesamiento sin usar. En cualquier momento, estos clientes representan 1000 *terabytes* de almacenamiento, 1.25 *GBps* de ancho de banda y 100,000 *GHz* de poder de procesamiento desperdiciado. Estos son valores conservadores que dan idea del potencial sin aprovechar esperando a ser explotado [38].

El *P2P* es la clave para aprovechar este potencial, dando a las máquinas individuales el mecanismo para proveer servicios unos a otros con alta disponibilidad y a un costo mucho más bajo que maximiza la utilización de los recursos de todos los nodos de la red.

La computación *P2P* nació en el año 1979, cuando Tom Truscott y Jim Ellis inventaron *usenet* como medio de intercambiar información entre dos computadoras en los primeros días antes de Internet [38]. La computación *P2P*, ganó popularidad gracias al surgimiento del programa *Napster* en 1999, este permitía a los usuarios compartir canciones digitales en formato mp3, sin embargo sufrió una demanda por parte de la industria musical debido a que se violaban los derechos de autor de las canciones que se compartían.

Aún así, el *P2P* se está convirtiendo en una importante técnica en diferentes áreas, como en la computación distribuida y colaborativa, tanto en la Web como en redes *ad-hoc*.

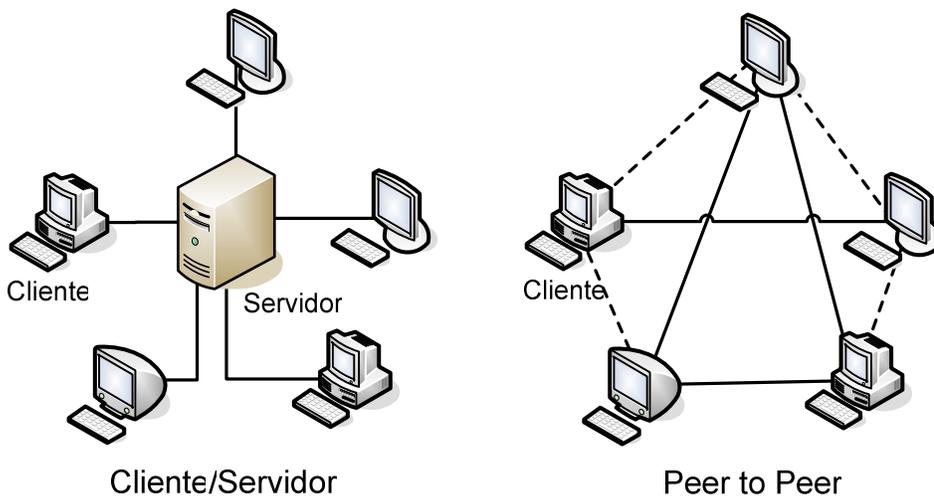
En el *P2P*, todo gira en torno a *compartir*, en dar a la comunidad y obtener algo también. Un usuario da algunos recursos y obtiene otros recursos a cambio. En el caso de *Napster*, se trataba de ofrecer música al resto de la comunidad y obtener música a cambio. Podría tratarse de donar recursos para una buena causa, como combatir el cáncer o la búsqueda de vida extraterrestre en el espacio, donde el beneficio es tener la satisfacción de ayudar a otros.

En un sistema de este tipo, los nodos son autónomos cuando no son completamente controlados por otros nodos o por el mismo usuario. Los nodos dependen unos de otros para obtener la información, los recursos, el re-envío de solicitudes, etc. Todo esto es esencial para el funcionamiento del sistema como un todo y para el beneficio de todos los participantes. Sin embargo, los

nodos no pueden confiar completamente en otros nodos, por lo que la escalabilidad y redundancia son temas más importantes que en la computación cliente/servidor.

En la figura 22, se puede apreciar la diferencia entre la computación cliente/servidor, en la cual los clientes se conectan a un servidor y el *P2P*, donde cada nodo es cliente y servidor de otros nodos.

Figura 22. Comparación de la computación cliente/servidor y el *P2P*



Fuente: Milojic, Dejan S. y otros. *Peer-to-Peer Computing*. HP Laboratories. <http://www.hpl.hp.com/techreports/2002/HPL-2002-57R1.pdf>

3.2 Elementos

El *P2P* es la solución a una pregunta: ¿Cómo conectar un conjunto de dispositivos de tal forma que puedan compartir información, recursos y servicios? Para responderla apropiadamente, se requiere responder varias preguntas implicadas [38]:

- ¿Cómo aprende un dispositivo la presencia de otro dispositivo?
- ¿Cómo se organizan los dispositivos para resolver intereses comunes?
- ¿Cómo logra un dispositivo que sus capacidades sean conocidas?
- ¿Qué se requiere para identificar a un dispositivo unívocamente?
- ¿Cómo intercambian datos los dispositivos?

Todas las redes *P2P*, se construyen sobre elementos fundamentales que proveen respuestas a estas preguntas y otras. Desafortunadamente, varios de estos elementos son asumidos o son implícitos en redes *P2P* propietarias y son codificadas en varias implementaciones, lo que produce inflexibilidad

3.2.1 Nodos

Un *peer* es un nodo en una red *P2P* que forma la unidad fundamental de procesamiento en cualquier solución *P2P*. Un nodo es una entidad capaz de realizar trabajo útil y comunicar su resultado a otra entidad sobre una red de forma directa o indirecta [38]. Existen tres tipos de nodos en cualquier red *P2P*:

- Nodos simples
- Nodos *Rendezvous*
- Nodos de ruteo

3.2.1.1 Nodos simples

Están diseñados para servir a un único usuario, permitiendo a ese usuario proveer servicios desde su dispositivo y consumir servicios de otros nodos en la red. Puede estar aislado de la red detrás de un *firewall*. Los nodos fuera del

firewall probablemente no podrán comunicarse directamente con el nodo ubicado dentro de él. Debido a su limitado acceso de la red, los nodos simples tienen la menor cantidad de responsabilidad en cualquier red *P2P*.

3.2.1.2 Nodos *Rendezvous*

Rendezvous significa un lugar de recolección o reunión. En el *P2P*, un nodo *rendezvous* provee a otros nodos una ubicación de red a usar para descubrir otros nodos y recursos. Los nodos envían peticiones de búsqueda a un nodo *rendezvous*, y éste provee información de los nodos que conoce en la red. Por lo regular este nodo existe fuera del firewall de una red privada. Podría existir dentro de un *firewall*, pero debe ser capaz de atravesar el *firewall* utilizando un protocolo autorizado por el *firewall* o usando un nodo de ruteo fuera de él.

3.2.1.3 Nodos de ruteo

Provee un mecanismo para comunicar nodos que se encuentran separados de la red por medio de un *firewall* o un equipo *NAT (Network Address Translation)*. Un nodo de ruteo sirve como intermediario de comunicación para que los nodos afuera del firewall puedan comunicarse con un nodo adentro de él y viceversa.

3.2.2 Grupos de nodos

Las soluciones propietarias y especializadas del *P2P* junto con sus protocolos, dividen el espacio de red de acuerdo a la aplicación. Es decir si alguien quiere compartir archivos usa el protocolo *Gnutella* [39], si quiere mensajería instantánea usa *ICQ* [43]. La incompatibilidad de los protocolos divide la red basado en la aplicación usada por los nodos involucrados.

Cuando todos los nodos hablan el mismo protocolo, como en el caso de *JXTA*, se crearon los grupos de nodos para dividir la red. Un grupo es un conjunto de nodos formados para servir un interés común u objetivo dictado por los nodos involucrados. Los grupos de nodos pueden ofrecer servicios a sus miembros que no son accesibles por otros nodos de la red *P2P*. La red *P2P* se puede dividir en grupos de nodos basándose en los siguientes criterios:

- a) **La aplicación en la que colaboran como un grupo.** Un grupo de nodos se forma para intercambiar servicios que los miembros no desean hacer disponibles para toda la población *P2P*. Una razón puede ser la naturaleza privada de los datos usados en la aplicación.
- b) **Los requerimientos de seguridad de los nodos involucrados.** Un grupo de nodos puede emplear autenticación para restringir quien puede unirse al grupo y acceder los servicios ofrecidos.
- c) **La necesidad de conocer el estado de los miembros.** Los miembros del grupo pueden monitorear otros miembros. Conocer el estado de los nodos puede ser útil para mantener un nivel de servicio mínimo para la aplicación del grupo de nodos.

3.2.3 Transporte de red

Para intercambiar datos, los nodos deben utilizar algún tipo de mecanismo para manejar la transmisión de datos sobre la red. La capa *transporte de red* es la responsable de todos los aspectos de transmisión, incluyendo dividir los datos en paquetes y agregar encabezados a los paquetes para controlar su destino. Un transporte de red puede ser un protocolo de bajo nivel como *TCP* y *UDP*, o puede ser uno de alto nivel como *http* y *SMTP*. El concepto de un transporte de red en el *P2P*, se puede dividir en tres partes, como se aprecia en la tabla VI.

Tabla VII. Partes de un transporte de red

Parte	Descripción
Punto	Corresponde a las interfaces de red utilizadas para enviar y recibir datos.
Enlace	Es un camino unidireccional, asíncrono y virtual de comunicación entre dos o más puntos.
Mensaje	Contenedor para los datos que son transmitidos a través de un enlace de un punto a otro.

Fuente: Brendon J. Wilson. *JXTA*. New Riders Publishing. Primera edición. 2002.

Para comunicarse usando un enlace, un nodo primero debe conocer los puntos, uno para el origen y uno para cada destino del mensaje, y conectarlos al vincular un enlace a cada punto. El enlace no es responsable de llevar los datos, es más bien una abstracción que representa el hecho que los puntos están conectados. Para enviar información, un nodo empaca los datos a enviar en un mensaje y luego lo envía a través del enlace. El nodo que recibe el mensaje extrae los datos.

3.2.4 Servicios

Los *servicios* proveen la funcionalidad que los nodos pueden usar para realizar trabajo útil en un nodo remoto. El trabajo incluye transferir un archivo, proveer información de estado, cálculos o cualquier cosa que se desee un nodo haga por nosotros. Los servicios son lo que motiva a reunir dispositivos en una red *P2P*, sin servicios no hay *P2P*. Los servicios se pueden dividir en:

- **Servicios de nodo:** funcionalidad ofrecida por un nodo particular en la red para otros nodos. La capacidad de este servicio será única para ese nodo y estará disponible solo cuando ese nodo esté conectado.
- **Servicios de grupo:** funcionalidad ofrecida por un grupo para los miembros del grupo. Esta funcionalidad puede ser provista por varios miembros del grupo, dando acceso redundante al servicio. El servicio está disponible siempre que un miembro del grupo esté conectado y dando el servicio.

3.2.5 Anuncios

Hasta ahora, las aplicaciones *P2P* han usado una forma informal de anunciarse, por ejemplo los resultados de una búsqueda pueden considerarse un anuncio que especifica la ubicación de un archivo en la red. Un anuncio se define como una representación estructurada de una entidad, servicio o recurso puesto a disposición por un nodo o grupo como parte de una red *P2P*.

JXTA da poder, al describir los recursos como anuncios y aprender como los anuncios simplifican la tarea de organizar una red *P2P*.

3.2.6 Protocolos

Cualquier intercambio de información se basa en un protocolo que diga que datos enviar y en qué orden. Un protocolo es una forma de estructurar el intercambio de información entre dos o más entidades que usan reglas que han sido previamente acordadas por todos los involucrados.

En lugar de enviar y recibir datos puros, los protocolos simplemente organizan el intercambio de anuncios entre nodos y contienen la información requerida para realizar una funcionalidad particular.

3.2.7 Identificación

La mayoría de elementos en una red *P2P*, necesitan algo que los identifique de forma unívoca en la red:

- **Nodos:** cada nodo necesita un identificador que los demás nodos puedan usar para localizarlo en la red.
- **Grupos:** un nodo necesita una forma de identificar a qué grupo desea unirse para realizar una acción.
- **Enlace:** para comunicarse, un nodo debe tener una forma de identificar un enlace que conecta dos puntos en la red.
- **Contenido:** la información debe ser identificada unívocamente para permitir a los nodos replicarla en toda la red. Los nodos pueden usar este identificador para encontrar el contenido en cualquier nodo.

3.3 Objetivos del *P2P*

Como en cualquier otro sistema de cómputo, la meta de los sistemas *P2P* es dar soporte a las aplicaciones que satisfacen las necesidades de los usuarios. La mayoría de las decisiones de usar una solución *P2P* se basa en uno de los siguientes objetivos.

3.3.1 Reducir y compartir costos

Los sistemas centralizados que sirven a muchos clientes, cargan con la mayoría de los costos del sistema. Cuando el costo es muy grande, una arquitectura *P2P* puede ayudar a distribuir el costo entre todos los nodos. Debido a la característica autónoma de los nodos, es importante que los costos sean compartidos de forma razonable y equitativa. Por ejemplo en el caso de *Napster*, el costo de almacenar las canciones era compartido entre todos los nodos del sistema.

3.3.2 Agregar recursos (mejorar rendimiento)

Un sistema descentralizado se presta naturalmente a la agregación de recursos, ya que cada nodo del sistema *P2P* trae consigo ciertos recursos como poder de cómputo y espacio de almacenamiento. Las aplicaciones que se benefician de grandes cantidades de recursos, se inclinan a una arquitectura *P2P* para obtener más recursos y resolver un problema mayor.

3.3.3 Mejorar escalabilidad y confiabilidad

Con la ausencia de una autoridad centralizada que gobierne los nodos autónomos, mejorar la escalabilidad y confiabilidad del sistema es un objetivo importante. La *escalabilidad* se relaciona con aspectos como el uso del ancho de banda, cuantos sistemas se pueden soportar, cuantos usuarios, etc. La *confiabilidad* se relaciona con la disponibilidad del sistema, el fallo de la red, desconexión, disponibilidad de recursos, etc. Como resultado, se han innovado los algoritmos para descubrir recursos, se ha hecho mucha investigación para mejorar los sistemas actuales y se han desarrollado nuevas plataformas *P2P*.

3.3.4 Incrementar la autonomía

Muchos de los usuarios de los sistemas distribuidos, no están dispuestos a confiar en un proveedor de servicios centralizados. En su lugar, prefieren que todos los datos y el trabajo sobre ellos, se realice de forma local. Los sistemas *P2P* soportan este nivel de autonomía simplemente porque requieren que el nodo local realice trabajo para el usuario.

3.3.5 Mejorar la privacidad y anonimato

En un servidor central, es difícil asegurar el anonimato porque el servidor normalmente puede identificar al cliente, al menos a través de su dirección *IP*. Al utilizar una arquitectura *P2P*, los usuarios pueden evitar dar información acerca de ellos a alguien más. Un ejemplo claro de esto es *Freenet* [41], que es una aplicación que utiliza un esquema de reenvío de mensaje que no permite localizar al usuario que originalmente solicitó un servicio.

3.3.6 Facilitar el dinamismo

Los sistemas *P2P*, asumen que el ambiente donde se desempeñan es altamente dinámico. Es decir, los recursos como los nodos, estarán entrando y saliendo del sistema continuamente. Cuando una aplicación debe soportar un ambiente bastante dinámico, la arquitectura *P2P* es una elección natural. Un ejemplo de este dinamismo, son las aplicaciones de mensajería instantánea, en la cual se mantienen listas de contactos, que son usadas para informar al usuario cuando alguien de esa lista se conecta.

3.3.7 Habilitar comunicaciones *ad-hoc*

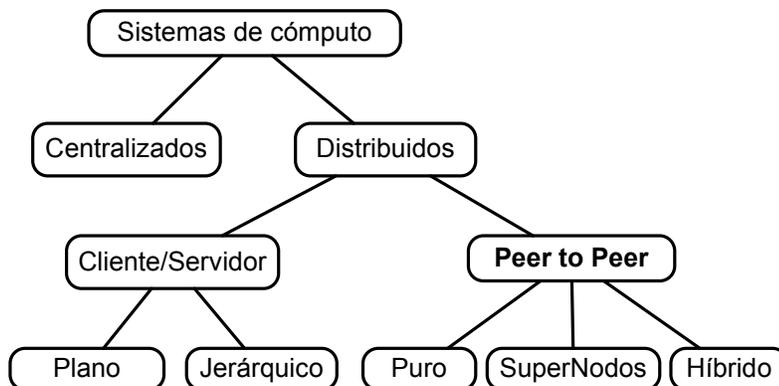
Ad-hoc significa ambientes donde los miembros van y vienen dependiendo de su ubicación física o sus intereses actuales. El *P2P* encaja en esta descripción, porque naturalmente toma en cuenta los cambios en el grupo de participantes. Los sistemas *P2P* no dependen de una infraestructura establecida porque crean la suya propia.

3.4 Clasificación del P2P

Todos los sistemas de cómputo pueden clasificarse en centralizados y distribuidos. Los sistemas distribuidos pueden clasificarse en el modelo cliente/servidor y el modelo P2P. El modelo cliente/servidor puede ser plano donde todos los clientes se comunican con un solo servidor, o puede ser jerárquico para mejorar la escalabilidad.

Según el grado de descentralización, la computación P2P puede ser clasificado en tres modelos: *puro* cuando no existe un servidor central, por ejemplo *Gnutella* [39] y *Freenet* [41]. También puede ser *híbrido* cuando se utiliza un servidor para obtener meta-información, como el listado de nodos conectados, por ejemplo *Napster* y *Aimster* [42]. Finalmente tenemos los *super nodos* que se encuentran en un punto intermedio, ya que utilizan nodos especiales que contienen información centralizada que otros nodos pueden consular, como ejemplo tenemos a *KaZaa* [43]. En la figura 23 se puede apreciar esta clasificación.

Figura 23. Clasificación de los sistemas de cómputo



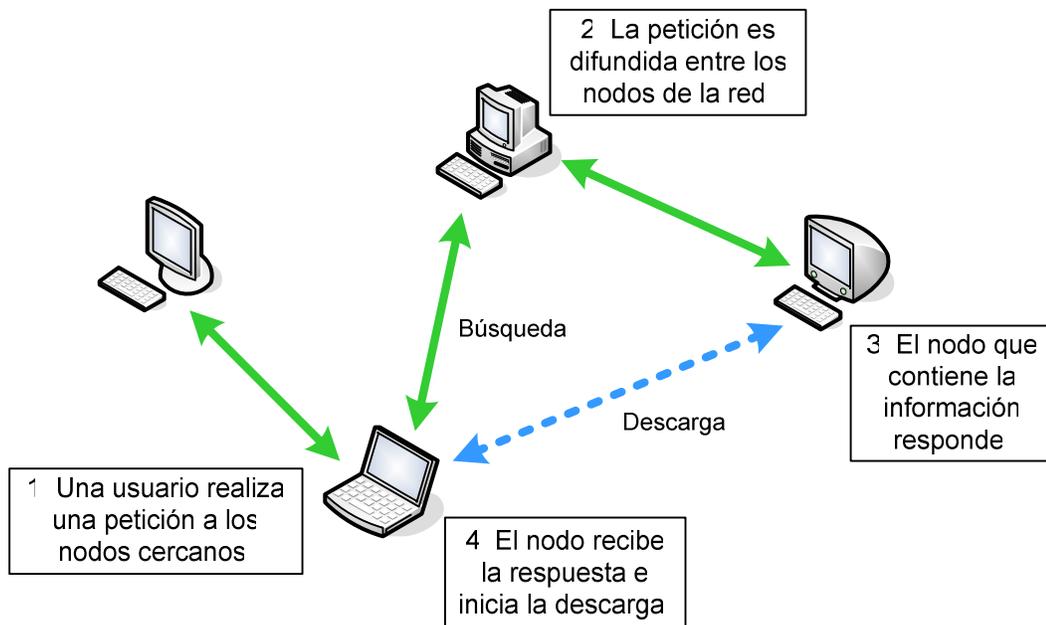
Fuente: Milojevic, Dejan S. y otros. *Peer-to-Peer Computing*. HP Laboratories. <http://www.hpl.hp.com/techreports/2002/HPL-2002-57R1.pdf>

3.4.1 Puro

En este modelo no existe un servidor central donde los nodos anuncien los recursos compartidos. En su lugar, cada petición de un nodo es difundida directamente a los nodos conectados, quienes a su vez difunden la petición entre sus nodos, así hasta que la petición es atendida o la petición alcanza un nivel máximo de difusión, que por lo regular oscila entre 5 y 9 niveles.

Este modelo es usado por *Gnutella* y requiere bastante ancho de banda. Debido a esto, las propiedades de escalabilidad del modelo son confundidas con sus propiedades de alcance. Si el objetivo es alcanzar a todos los nodos de la red, entonces este modelo no es muy escalable, sin embargo es bastante eficiente en comunidades limitadas. Manipulando el número de conexiones de cada nodo y configurando la propiedad *TTL* de los mensajes, este modelo puede llegar a alcanzar cientos de miles de nodos.

Figura 24. Funcionamiento del modelo P2P Puro

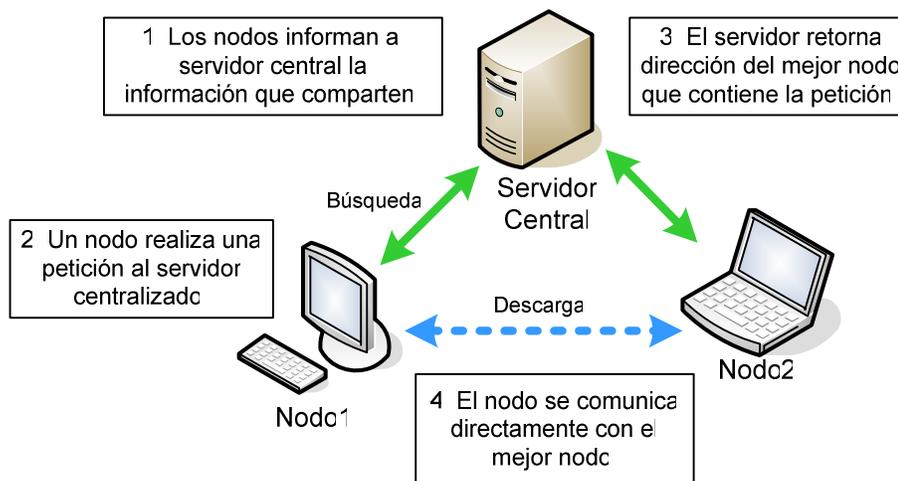


3.4.2 Híbrido

Este modelo fue popularizado por *Napster*. Los nodos de la red se conectan a un directorio centralizado donde publican información sobre el contenido que ofrecen para compartir. Cuando un nodo solicita información, el índice central busca el mejor nodo que coincida con la petición. El mejor nodo puede ser el más barato, el más rápido, o el más disponible, dependiendo de las necesidades del usuario. Luego el nodo que solicita se conectará directamente con el mejor nodo, donde ocurrirá un intercambio de información. Este modelo se puede apreciar mejor en la figura 25. Otra aplicación popular que utiliza este modelo es *ICQ* (mensajería instantánea) [40].

La desventaja de este modelo es la escalabilidad, ya que al tener una red de nodos bastante extensa, el servidor debe ser lo suficientemente poderoso para poder atender las peticiones de los usuarios.

Figura 25. Modelo de directorio centralizado

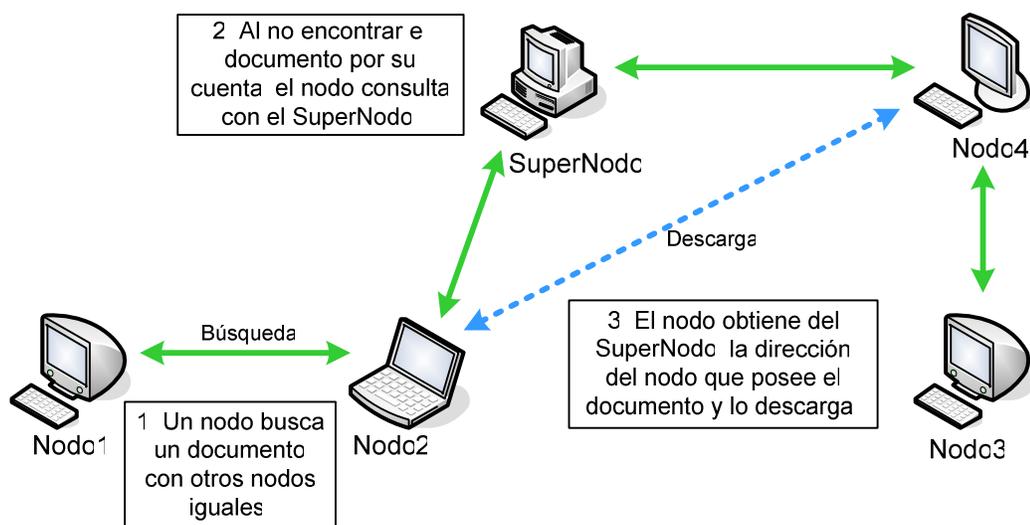


3.4.3 Super nodos

Este modelo es similar al modelo puro, con la diferencia que algunos de los nodos, asumen un rol más importante que los demás y reciben el nombre de *super nodos*, cuyo objetivo es servir de índice central para el contenido compartido de los nodos locales. En la figura 26 se puede apreciar el funcionamiento de este modelo.

La forma en que estos *super nodos* son asignados varía de sistema en sistema, algunos sistemas utilizan por ejemplo el poder de procesamiento de la máquina. Sin embargo, lo importante es saber que estos *super nodos* no constituyen puntos únicos de fallo para la red *P2P*, ya que son dinámicamente asignados, y si llega a fallar uno de ellos, el sistema automáticamente toma acción para reemplazarlos con otros.

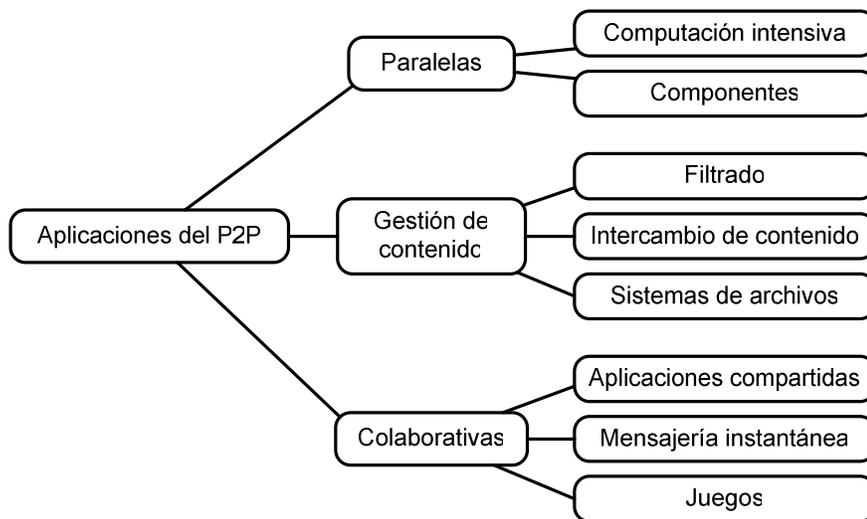
Figura 26. Funcionamiento del modelo *P2P* super nodos



3.5 Aplicaciones del P2P

Existen tres categorías principales de las aplicaciones del P2P, cada una de las cuales tiene aplicaciones más especializadas, como se puede apreciar en la figura 27.

Figura 27. Clasificación de aplicaciones P2P



Fuente: Milojevic, Dejan S. y otros. *Peer-to-Peer Computing. HP Laboratories.*
<http://www.hpl.hp.com/techreports/2002/HPL-2002-57R1.pdf>

3.5.1 Paralelas

Las aplicaciones paralelas, dividen una gran tarea de cómputo en piezas más pequeñas que pueden ser ejecutadas en paralelo sobre un conjunto de nodos independientes. Se dividen en aplicaciones de computación intensiva y por componentes. En las aplicaciones de computación intensiva, cada nodo ejecuta el mismo componente de código con parámetros distintos, mientras que en las aplicaciones por componentes, cada nodo ejecuta un componente distinto de código. Como ejemplos de la primera tenemos la búsqueda de vida

extraterrestre, descifrar claves, cálculo de riesgos, evaluación de crédito y análisis demográfico. Ejemplos de la segunda tenemos a JavaBeans y a los servicios Web en general.

3.5.2 Gestión de contenido

Las aplicaciones de gestión de contenido y archivos se centran en la forma de almacenar información y recuperarla de varios nodos de la red. Como ejemplo de aplicaciones de intercambio de contenido tenemos a *Napster* y *Gnutella* [39], ya que permiten a los nodos buscar y descargar archivos, principalmente música digital que otros nodos han compartido. Las aplicaciones de filtrado se centran en técnicas de filtrado que permiten construir índices sobre una red de nodos. Ejemplo de estas tecnologías tenemos a *JXTA* [38] y *OpenCOLA*.

3.5.3 Colaborativas

Las aplicaciones *P2P* colaborativas permiten a los usuarios colaborar en tiempo real, sin depender un servidor central para coleccionar y distribuir la información. La mensajería instantánea como *MSN Messenger*, *AIM* y *Yahoo Messenger*, son ejemplos de aplicaciones de este tipo que se han vuelto populares. También están emergiendo aplicaciones que permiten a los usuarios que posiblemente se encuentran a miles de kilómetros de distancia, interactuar al ver y editar la misma información al mismo tiempo. Los juegos son otro ejemplo de aplicaciones colaborativas, como *Scour Exchange*.

3.6 El *P2P* y los negocios

Desde el inicio el *P2P* estuvo bastante alejado de los negocios, sin embargo, esto está empezando a cambiar. Toda la tecnología usada con los modelos *P2P* presenta grandes ventajas para las organizaciones. Hoy en día es difícil encontrar a un usuario que no haya oído hablar de *Napster* o de otras aplicaciones de intercambio de recursos.

La popularidad de tal servicio fue tanta, que las compañías discográficas demandaron a *Napster* legalmente. Esto nos lleva a la conclusión que el modelo de transacción *P2P* puede convertirse en un modelo de negocio. Existen dos formas que pueden ser utilizadas para generar ganancias utilizando el *P2P* [45].

- **Publicidad**

Las aplicaciones *P2P* pueden convertirse en plataformas de publicidad para otras compañías, donde sus usuarios aceptarían recibir anuncios publicitarios por medio de la aplicación.

- **Venta de información**

Se puede vender la información de los usuarios de la comunidad *P2P*, incluyendo sus datos personales, sus gustos y preferencias, a otras empresas que estén interesadas en hacer estudios de mercado. Esto debe hacerse siempre con el consentimiento de los usuarios.

3.7 Características

3.7.1 Descentralización

Uno de los puntos fuertes de la descentralización se basa en el control de los datos y recursos en las manos de los usuarios. En un sistema completamente descentralizado, cada nodo es un participante igual. Sin embargo esto hace la implementación mucho más difícil, y es por eso que existe el modelo híbrido.

En sistemas completamente descentralizados, como *Freenet* [41] y *Gnutella* [39], el simple hecho de encontrar la red es difícil. En el caso de *Gnutella* [39], los nuevos nodos deben descubrir la dirección IP de otro nodo, el nodo se une a la red al establecer una conexión con por lo menos un nodo actual de la red. Luego puede descubrir otros nodos y almacenar su dirección IP localmente.

3.7.2 Escalabilidad

Un efecto inmediato de la descentralización es la escalabilidad. La escalabilidad es limitada por factores como la cantidad de operaciones centralizadas (sincronización y coordinación) que deben ser realizadas, la cantidad de estado que debe ser mantenido, el paralelismo heredado que una aplicación exhibe, y el modelo de programación usado en la implementación.

La escalabilidad también depende del factor de comunicación a computación entre los nodos de un sistema *P2P*. Por ejemplo *Napster* era bien

escalable debido a que los nodos descargaban directamente los archivos de los otros nodos. *SETI@home* [46] que se dedica a la búsqueda de vida extraterrestre, presenta un problema de escalabilidad, por la cantidad de datos que se deben transmitir luego del procesamiento.

3.7.3 Anonimato

Una de las metas del *P2P* es permitir a las personas usar los sistemas sin preocuparse por implicaciones legales. Otra de las metas es garantizar que la censura del contenido digital no sea posible. Existen algunos aspectos relacionados con el anonimato:

- **Autor:** el autor o creador de un documento no puede ser identificado.
- **Publicador:** la persona que publicó el documento en el sistema no puede ser identificada.
- **Lector:** las personas que leen o consumen datos no pueden ser identificadas.
- **Servidor:** los servidores que comparten un documento, no pueden ser identificados basados en el documento.
- **Documento:** los servidores no conocen los documentos que almacenan.
- **Consulta:** un servidor no puede decir que documento utiliza para responder la consulta de un usuario.

Sin importar las entidades anteriores, existen tres tipos diferentes de anonimato entre cada par que entabla una comunicación:

1. *Anonimato del emisor*: esconde la identidad del emisor.
2. *Anonimato del receptor*: esconde la identidad del receptor.
3. *Anonimato mutuo*: las identidades del emisor y receptor son escondidas entre ambos y entre cualquier otro nodo.

Además de estos tipos de anonimato, también es importante conocer el grado de anonimato que cierta técnica puede lograr. Existe un rango de grados de anonimato que van desde *privacidad absoluta*, *más allá de la sospecha*, *inocencia probable* y *probablemente expuesto*. Por ejemplo, más allá de la sospecha significa que aunque un atacante puede ver la evidencia de un mensaje enviado, el emisor no parece más el generador del mensaje que cualquier otro emisor potencial en el sistema.

Estas técnicas son usadas para reforzar diferentes tipos de anonimato y con diferentes restricciones.

3.7.3.1 Multicasting

Esta técnica puede ser utilizada para reforzar el anonimato del receptor. Un grupo de *multicasting* es formado por participantes que desean permanecer anónimos. Una entidad que se interesa en recibir un documento, se suscribe a un grupo de *multicasting*. El participante que posee el documento, lo envía a todo el grupo. La identidad del solicitante es efectivamente escondida del emisor y de otros miembros del grupo, y el anonimato del solicitante queda *más allá de la sospecha*.

3.7.3.2 Aplicar *spoofing* al emisor

Para protocolos no orientados a la conexión como *UDP*, el anonimato del emisor puede reforzarse al cambiar la dirección *IP* del emisor. Esto implica modificar el protocolo. Además esto no siempre es posible debido a que los proveedores de servicio de Internet filtran las direcciones inválidas.

3.7.3.3 Identificar el *spoofing*

Además de cambiar la dirección de origen, el anonimato también puede asegurarse al cambiar la identidad del participante que se comunica. Por ejemplo en *Freenet* [41], un nodo puede pasar un archivo a un solicitante (ya sea de su propio caché o proveniente de otro nodo) y reclamar ser el dueño del contenido. El que responde es *posiblemente inocente* desde el punto de vista del atacante, debido a que existe una probabilidad no trivial de que el verdadero nodo que respondió sea otro.

3.7.3.4 Rutas cubiertas

En lugar de comunicarse directamente, dos participantes pueden comunicarse a través de nodos intermedios. La mayoría de las técnicas existentes aseguran solo el anonimato del emisor. Un participante que desea esconder su identidad, prepara una ruta cubierta con otro participante al final de la ruta. Cambiando la longitud de la ruta cubierta y cambiando la ruta seleccionada con diferente frecuencia, se pueden lograr diferentes grados de anonimato.

3.7.3.5 Sinónimos no rastreables

LPWA es un servidor *Proxy* que genera sinónimos consistentes para clientes no rastreables por los servidores. Un cliente abre una cuenta y puede ser reconocido al regresar a la cuenta abierta, mientras la verdadera identidad del cliente se esconde del servidor. Esta técnica asegura anonimato del emisor y depende de un servidor *Proxy*. El grado de anonimato que puede ser logrado se encuentra entre *privacidad absoluta* y *más allá de la sospecha*.

3.7.3.6 Colocación no voluntaria

Esta nueva técnica logra anonimato a través de colocar de forma no voluntaria un documento en un nodo. El que publica, fuerza un documento sobre un nodo. Debido a que la colocación es no voluntaria, el nodo que reside el documento no puede ser culpado de ser su dueño.

3.7.4 Organización autónoma

Se puede definir la organización autónoma como un proceso donde la organización (restricciones y redundancia) de un sistema aumenta espontáneamente, sin que este aumento esté controlado por el ambiente u otro sistema externo. En el *P2P*, la organización autónoma es necesaria para soportar la escalabilidad, la resistencia a fallos y la conexión de recursos. Es difícil para cualquier configuración predefinida, mantenerse intacta por cualquier período de tiempo, por lo que el sistema *P2P* debe adaptarse a los cambios de los nodos conectándose y desconectándose del sistema. Y debido a que tener

un equipo dedicado de personas a administrar tal sistema sería demasiado costoso, la administración es distribuida entre los nodos.

3.7.5 Costo compartido

El costo compartido, reduce el costo de ser el dueño del sistema y su contenido, así como el costo de mantenerlos. Esta premisa se puede aplicar a todos los sistemas *P2P*. Como ejemplo, *SETI@home* es más rápido que la supercomputadora más rápida del mundo y a solo una fracción de su costo, equivalente a 1%.

3.7.6 Conectividad ad-hoc

La naturaleza de conectividad ad-hoc, tiene un fuerte efecto en todas las clases de sistemas *P2P*. Mientras que en la computación distribuida, los nodos entrando y saliendo de la red se consideran como un evento excepcional, en el *P2P* es considerado un evento usual.

En los sistemas *P2P* de contenido, los usuarios esperan acceder el contenido inmediatamente, dependiendo de la conectividad de los proveedores de contenido. En los sistemas que ofrecen más garantías, como acuerdos de nivel de servicio, la naturaleza ad-hoc es reducida usando proveedores de servicio redundantes, sin embargo, las partes de los proveedores aún pueden estar no disponibles.

3.7.7 Rendimiento

El rendimiento es un punto importante en el *P2P*. Los sistemas incrementan el rendimiento, al agregar capacidad de almacenamiento distribuida, por ejemplo *Napster* y *Gnutella* [39]. Y ciclos de procesamiento en el caso de *SETI@home* [46] con dispositivos dispersos a través de la red.

El rendimiento es influenciado por tres tipos de recursos: *procesamiento*, *almacenamiento* y *red*. En particular, los tiempos de espera en la red pueden ser mayores en redes *WAN*. El ancho de banda es un gran factor cuando una gran cantidad de mensajes son propagados en la red y gran cantidad de archivos son transferidos entre los nodos. Esto limita la escalabilidad del sistema. A este nivel el rendimiento no se enfoca en los milisegundos, sino más bien trata de responder preguntas como ¿cuánto tiempo toma transferir un archivo? ¿Cuánto ancho de banda consume una consulta?

En el modelo puro del *P2P*, no existe coordinador central, la comunicación es manejada por los nodos individuales, cada uno de los cuales envía mensajes para buscar información y datos. Pero esto presenta el problema de inundar la red con mensajes y consumir el ancho de banda. Existen tres soluciones para optimizar el rendimiento: replicación, caché y ruteo inteligente.

3.7.7.1 Replicación

Se basa en colocar copias de los archivos, cerca de los nodos solicitantes, de esta forma se minimiza la distancia de conexión entre el nodo que solicita y el nodo que provee la información. Los cambios a los archivos, deben ser propagados a todas las replicas existentes. Al combinar la replicación con el

ruteo inteligente, se minimiza la distancia y el tiempo de espera al enviar las solicitudes a los nodos locales más cercanos.

3.7.7.2 Caché

Al usar caché, se reduce la longitud requerida para recuperar un archivo, y por ende la cantidad de mensajes intercambiados entre los nodos. Reducir estas transmisiones es importante para los sistemas *P2P*, porque la latencia de comunicación entre los nodos forma un gran cuello de botella. Por ejemplo en *Freenet* [41], al encontrar y propagar un archivo al nodo solicitante, el archivo es cacheado en todos los nodos de la ruta de retorno.

3.7.7.3 Ruteo inteligente

La idea principal detrás del ruteo inteligente, es tratar de conectar dos nodos que sean compatibles desde el punto de vista de sus preferencias de búsqueda. Establecer un buen conjunto de nodos, reduce la cantidad de mensajes propagados en la red y la cantidad de nodos que procesan una solicitud antes que el resultado sea encontrado. Varios sistemas académicos como por ejemplo *Oceanstore*, mejoran el rendimiento al mover de forma proactiva la información dentro de la red. La gran ventaja de esto, es que los nodos deciden con quien conectarse/desconectarse en base a información local.

3.7.8 Seguridad

Los sistemas *P2P* comparten sus necesidades de seguridad con los sistemas distribuidos tradicionales. Sin embargo con el desarrollo de nuevos sistemas *P2P*, han aparecido más requerimientos de seguridad.

Se han creado sistemas *P2P* que intentan proteger el contenido compartido, así como el anonimato del autor, del nodo transmisor y receptor. Para lo que se han desarrollado métodos de encriptación de llave múltiple.

Como en el *P2P* los archivos se copian de un nodo a otro de forma fácil, es necesario también proteger a los autores para que su propiedad intelectual no sea robada. Una forma de hacer esto, es agregar una firma digital al archivo compartido, que lo permita reconocer sin afectar su contenido.

También es importante la reputación de los nodos, es decir, una forma de medir que tan bueno o útil es un nodo. Una forma de medir esto, sería por la cantidad de archivos que comparte para los demás nodos, entonces esto le daría al nodo una reputación alta. Mientras que alguien que no comparte y solo descarga, tendría una reputación baja.

Finalmente, las aplicaciones *P2P* requieren conexión directa entre los nodos, y muchas veces esto no es posible debido a que los nodos se encuentran en un ambiente corporativo, en donde los firewalls bloquean todas las conexiones *TCP* entrantes que no hayan sido iniciadas desde el nodo interno. Además de esto existen usuarios que usan *NAT* (*Network Address Translation*, traducción de direcciones de red) para compartir un mismo enlace a Internet. Esto ha llevado a desarrollar técnicas para poder atravesar dichas tecnologías.

3.7.9 Transparencia

La transparencia se refiere a que los sistemas *P2P* no requieren ningún tipo considerable de configuración, ya sea de las redes o de los dispositivos para poder funcionar. Deberían funcionar en Internet, intranets y redes privadas, utilizando conexiones de alta velocidad o de líneas telefónicas. También deberían de poder correr en una gran variedad de dispositivos, por ejemplo en *PDA*s (*Personal Digital Assistant*, asistente personal digital), computadoras de escritorio, teléfonos móviles, etc.

En la práctica, lograr la transparencia es un reto que asumen los desarrolladores de herramientas y plataformas para redes *Peer-to-Peer*.

3.7.10 Tolerancia a fallos

Uno de los objetivos que se quieren alcanzar al diseñar un sistema *P2P*, es evitar un punto central de fallo, y aunque el modelo *P2P* Puro ya logra esto, siempre se enfrentan con fallas como desconexiones, nodos inalcanzables, particiones y fallas de nodo. Estas fallas se presentan más en unas redes, como las inalámbricas, que en otras, como las de cableado estructurado. Sería deseable que el sistema continuara funcionando con los nodos activos aún cuando se presentara una falla.

La no disponibilidad de recursos puede ocurrir porque el recurso es inalcanzable debido a una falla en la red, o porque el nodo que contiene el recurso ha fallado o se ha desconectado. El primer caso puede resolverse usando una ruta diferente. Replicar los recursos ayuda a resolver el problema.

En conclusión, la responsabilidad del mantenimiento del sistema es completamente distribuida y cada nodo debe hacer su parte para asegurar la disponibilidad de recursos.

3.7.11 Interoperabilidad

Para desarrollar sistemas *P2P* que sean capaces de interoperar juntos, hay que tomar en cuenta ciertos requerimientos.

- Determinar cómo los sistemas pueden interoperar.
- ¿Cómo se comunican los sistemas? ¿qué protocolo deberían utilizar?, a través de sockets, mensajes o http.
- ¿Como los sistemas intercambian peticiones y datos, y como ejecutan tareas a un nivel superior?
- Determinar si los sistemas son compatibles en los protocolos superiores, ¿puede un sistema confiar en otro para buscar apropiadamente una porción de información?
- ¿Como los sistemas anuncian y mantienen el mismo nivel de seguridad, calidad de servicio y disponibilidad?

En el *P2P*, se han hecho grandes esfuerzos para mejorar la interoperabilidad, aún cuando todavía no es utilizada. El *P2P Working Group* (grupo de trabajo del P2P) es un intento para reunir a la comunidad de desarrolladores *P2P* y establecer un espacio común al escribir reportes y especificaciones que permitan un entendimiento común entre los desarrolladores.

JXTA se aproxima a la interoperabilidad, como un esfuerzo de código abierto, al intentar imponer un estándar por defecto. Hay un número de desarrolladores que son invitados a contribuir al código común con diferentes partes de funcionalidad. Solo una mínima arquitectura es soportada como base, permitiendo a otros sistemas contribuir con piezas compatibles con su propia implementación.

4. LA PLATAFORMA JXTA

4.1 Introducción

Desafortunadamente, las aplicaciones *P2P* actuales, tienden a usar protocolos que son propietarios e incompatibles por naturaleza, reduciendo la ventaja que se obtiene al reunir dispositivos en redes *P2P*. Cada red forma una comunidad cerrada, completamente independiente de las demás redes e incapaz de compartir sus recursos.

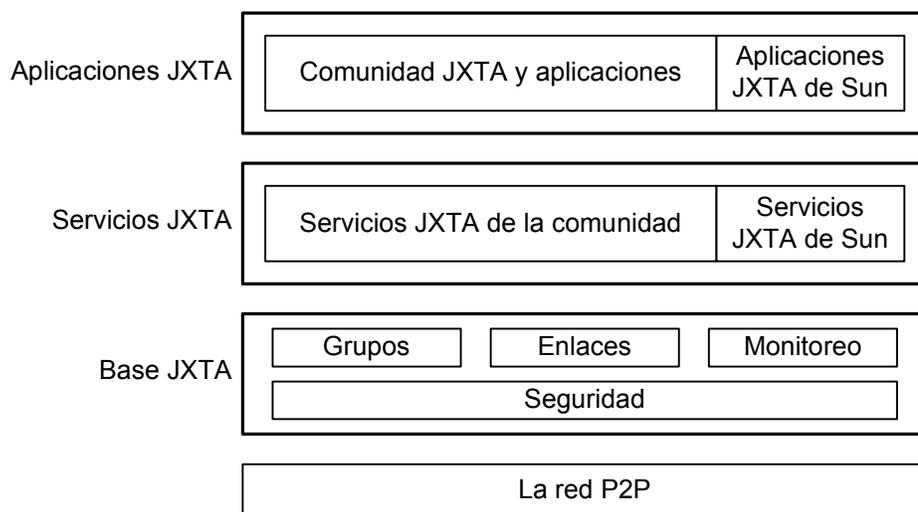
Para evolucionar el *P2P* a una plataforma madura, los desarrolladores necesitan reenfocar sus esfuerzos de programar fundamentos de redes *P2P* hacia la creación de aplicaciones *P2P* en una base sólida y bien definida. Para hacer esto, los desarrolladores del *P2P* necesitan un lenguaje común que permita a los nodos comunicarse y realizar las cosas fundamentales de las redes *P2P*.

Para atender esta necesidad, *Sun Microsystems* creó el proyecto *JXTA* (pronunciado *juxtapose* o *juxta*), formado por un equipo de desarrollo bajo la guía de Bill Joy y Mike Clary, para diseñar una solución que sirva a todas las aplicaciones *P2P*. *JXTA* es simplemente un conjunto de especificaciones de protocolo. Esto es lo que lo hace tan poderoso. Cualquiera que desee crear una nueva aplicación *P2P*, es aliviado de la dificultad de diseñar protocolos adecuados que manejen las funciones fundamentales de la comunicación *P2P* [38].

4.2 Capas lógicas

La plataforma *JXTA* se compone de tres capas lógicas, como se muestran en la figura 28. Cada capa depende de la capa inferior.

Figura 28. Capas lógicas de la plataforma *JXTA*



Fuente: Brendon J. Wilson. *JXTA*. New Riders Publishing. Primera edición. 2002.

4.2.1 Capa base

La capa base provee los elementos esenciales a cada solución *P2P*. Idealmente, los elementos de esta capa son compartidos por todas las aplicaciones *P2P*. Estos elementos fueron descritos en el capítulo 3. Los elementos son los siguientes: nodos, grupos, transporte de red (puntos, enlaces y mensajes), anuncios, identificación de entidades, protocolos, seguridad y autenticación.

La capa base incluye los seis protocolos de *JXTA*. Y aunque los protocolos son implementados como servicios base, se pueden distinguir de los servicios de la capa de servicios.

4.2.2 Capa de servicios

Provee servicios de red que son deseables pero no necesariamente forman parte de todas las soluciones *P2P*. Estos servicios implementan funcionalidad que puede ser incorporada en diferentes aplicaciones *P2P*, como por ejemplo: buscar recursos en un nodo, compartir documentos desde un nodo y autenticar los nodos.

La capa de servicios encapsula funcionalidad adicional desarrollada por la comunidad *JXTA*, además de los servicios creados por el equipo original.

4.2.3 Capa de aplicaciones

La capa de aplicación usa la capa de servicios para proveer las aplicaciones comunes del *P2P* que conocemos hasta ahora. Debido a que una aplicación puede abarcar un solo servicios o agregar varios, a veces es difícil determinar lo que constituye un servicio y una aplicación. Por lo regular el uso de una interfaz de usuario implica una aplicación en lugar de un servicio.

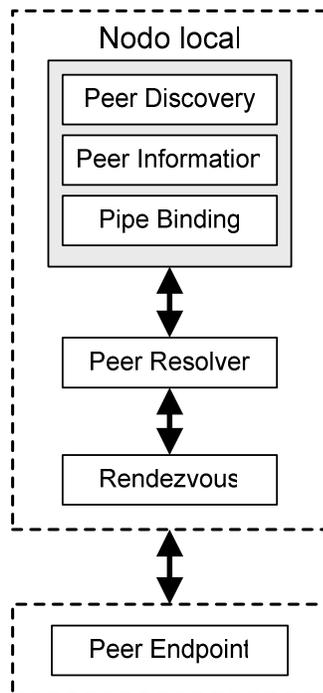
Entre las aplicaciones, tenemos a todas las soluciones *P2P* creadas por la comunidad *JXTA*, así como aplicaciones de demostración creados por *Sun*, como la *JXTA Shell*.

4.3 Protocolos

Al diseñar el conjunto de protocolos de *JXTA*, se tomó la decisión de resolver las necesidades de la mayoría de aplicaciones *P2P*. Con esto en mente, no se distingue entre el tipo de sistema operativo usado por un nodo. Un nodo puede ser cualquier dispositivo, desde el más pequeño dispositivo embebido hasta una supercomputadora. Además de esto, asume que los nodos y sus recursos pueden aparecer y desaparecer espontáneamente de la red, y que la ubicación de un nodo puede cambiar espontáneamente o ser ocultada por un equipo *NAT* o un *firewall*.

Para lograr estos objetivos, se diseñaron seis protocolos basados en mensajes *XML*. Estos protocolos se pueden apreciar en la figura 29.

Figura 29. Protocolos de la plataforma *JXTA*

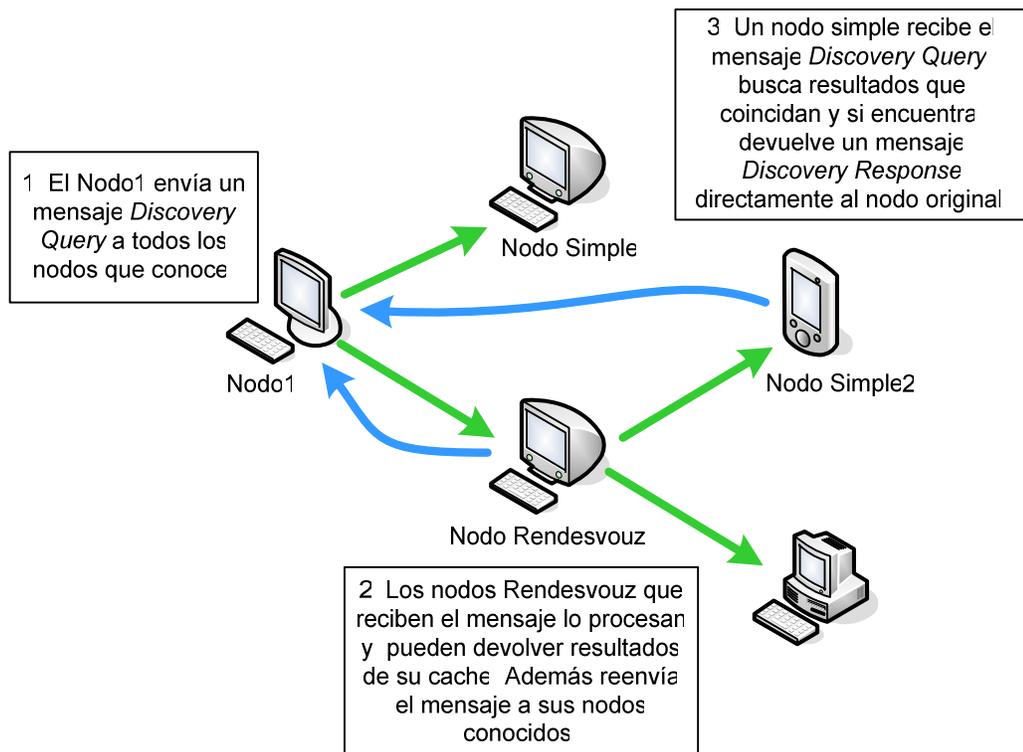


Fuente: Brendon J. Wilson. *JXTA*. New Riders Publishing. Primera edición. 2002.

4.3.1 Peer Discovery

En el capítulo 3, vimos que los anuncios son la unidad básica de intercambio entre nodos para proveer la información de los recursos disponibles. Este protocolo ayuda a descubrir los recursos existentes entre los nodos de la red *P2P*. Consiste de dos mensajes. El mensaje *Discovery Query* que define el formato de solicitud para descubrir anuncios. El mensaje *Discovery Response* que define el formato de la respuesta. En la figura 30 se puede apreciar el funcionamiento de este protocolo.

Figura 30. Funcionamiento del protocolo Peer Discovery



Fuente: Brendon J. Wilson. *JXTA*. New Riders Publishing. Primera edición. 2002.

4.3.1.1 Mensaje *Discovery Query*

Este mensaje es enviado a otros nodos para descubrir anuncios. Tiene un formato simple, como se muestra a continuación.

```
<?xml version="1.0" encoding="UTF-8"?>
<jxta:DiscoveryQuery>
<Type> . . . </Type>
<Threshold> . . . </Threshold>
<PeerAdv> . . . </PeerAdv>
<Attr> . . . </Attr>
<Value> . . . </Value>
</jxta:DiscoveryQuery>
```

Los nodos retornan anuncios que coincidan con todos los requerimientos de los parámetros. El significado de cada parámetro se lista en la tabla VII.

Tabla VIII. Parámetros del mensaje *Discovery Query*

Parámetro	Descripción
<i>Type</i>	Parámetro requerido. Es un valor entero que indica el tipo de anuncio solicitado. 0=nodos 1=grupos 2=cualquier otro tipo.
<i>Threshold</i>	Elemento opcional. Es un valor que indica la cantidad máxima de anuncios que el nodo que responde debería dar como respuesta.
<i>PeerAdv</i>	Elemento opcional. Contiene el anuncio del nodo que hace la solicitud. Identificador único del nodo originario para permitir a otros nodos responder directamente la solicitud.
<i>Attr, Value</i>	Par de elementos opcionales, que juntos indican el criterio que un anuncio debe cumplir para ser retornado como respuesta de la consulta.

Fuente: Brendon J. Wilson. *JXTA. New Riders Publishing*.
Primera edición. 2002.

4.3.1.2 Mensaje *Discovery Response*

Para responder un *mensaje Discovery Query*, un nodo crea un *mensaje Discovery Response* conteniendo los anuncios que coincidan con el criterio de búsqueda de la solicitud. Su formato se muestra a continuación:

```
<?xml version="1.0" encoding="UTF-8"?>
<jxta:DiscoveryResponse>
<Type> . . . </Type>
<Count> . . . </Count>
<PeerAdv> . . . </PeerAdv>
<Attr> . . . </Attr>
<Value> . . . </Value>
<Response Expiration="expiration time">
. . .
</Response>
</jxta:DiscoveryResponse>
```

Los parámetros del mensaje de respuesta coinciden bastante con los de solicitud y se describen en la Tabla VIII.

Tabla IX. Parámetros del mensaje *Discovery Response*

Parámetro	Descripción
<i>Type</i>	Es similar al parámetro de solicitud.
<i>Count</i>	Elemento opcional. Es un valor entero que representa la cantidad total de elementos de respuesta.
<i>PeerAdv</i>	Elemento opcional. Contiene el anuncio del nodo que envió la respuesta a la solicitud original.
<i>Attr, Value</i>	Par de elementos opcionales, que juntos indican el criterio de búsqueda que generó esta respuesta.
<i>Response</i>	Elemento opcional que contiene el anuncio que coincidió con el criterio de búsqueda.

Fuente: Brendon J. Wilson. *JXTA. New Riders Publishing*.
Primera edición. 2002.

4.3.2 *Peer Resolver*

El protocolo *Peer Discovery* funciona encima de otro protocolo llamado *Peer Resolver*, el cual se encarga de envolver los mensajes de solicitud en un formato más genérico para enviarlo a un manejador específico en un nodo remoto. En el nodo remoto, una instancia de un servicio de resolución, es responsable de trasladar cualquier mensaje entrante al manejador apropiado, y de devolver cualquier respuesta generada por el manejador.

En general se necesitan dos mensajes, el mensaje *Resolver Query* para enviar consultas y el mensaje *Resolver Response* para enviar respuestas.

4.3.2.1 Mensaje *Resolver Query*

Todas las consultas enviadas a otros nodos, son envueltas dentro de un mensaje *Resolver Query*. Su formato se muestra a continuación.

```
<?xml version="1.0" encoding="UTF-8"?>
<jxta:ResolverQuery xmlns:jxta="http://jxta.org">
  <HandlerName> . . . </HandlerName>
  <Credential> . . . </Credential>
  <QueryID> . . . </QueryID>
  <SrcPeerID> . . . </SrcPeerID>
  <Query> . . . </Query>
</jxta:ResolverQuery>
```

Los elementos dentro de éste mensaje, proveen todos los detalles que un nodo necesita para coincidir una consulta a un manejador registrado.

- **HandlerName:** elemento obligatorio, contiene el nombre único especificando el nombre del manejador que debería ser invocado en el nodo remoto para procesar la consulta.
- **Credential:** elemento opcional. Indica el *token* de autenticación que identifica el nodo fuente y su autorización para enviar el mensaje al grupo de nodos.
- **QueryID:** elemento opcional. Entero codificado como *string*, que define un identificador para la consulta. Y debería ser enviado de regreso en respuesta a una consulta.
- **SrcPeerID:** elemento obligatorio. Contiene el identificador del nodo que envía la consulta, en formato *JXTA URN*.
- **Query:** elemento obligatorio, Contiene la consulta que se envía al manejador del nodo remoto.

4.3.2.2 Mensaje *Resolver Response*

Este mensaje es usado para responder el mensaje *Resolver Query*, usando el siguiente formato:

```
<?xml version="1.0" encoding="UTF-8"?>
<jxta:ResolverResponse xmlns:jxta="http://jxta.org">
<HandlerName> . . . </HandlerName>
<Credential> . . . </Credential>
<QueryID> . . . </QueryID>
<Response> . . . </Response>
</jxta:ResolverResponse>
```

Este mensaje provee detalles similares al mensaje *Resolver Query*:

- ***HandlerName***: elemento obligatorio, que especifica el nombre del manejador a invocar en el nodo remoto que procesa la respuesta.
- ***Credential***: elemento opcional, que contiene el *token* de autenticación que identifica al nodo que envía la respuesta.
- ***QueryID***: elemento opcional. Define un identificador de la consulta que es respondida con este mensaje.
- ***Response***: elemento obligatorio. Contiene el *string* de respuesta que se envía. Puede ser cualquier cosa.

4.3.3 *Rendezvous*

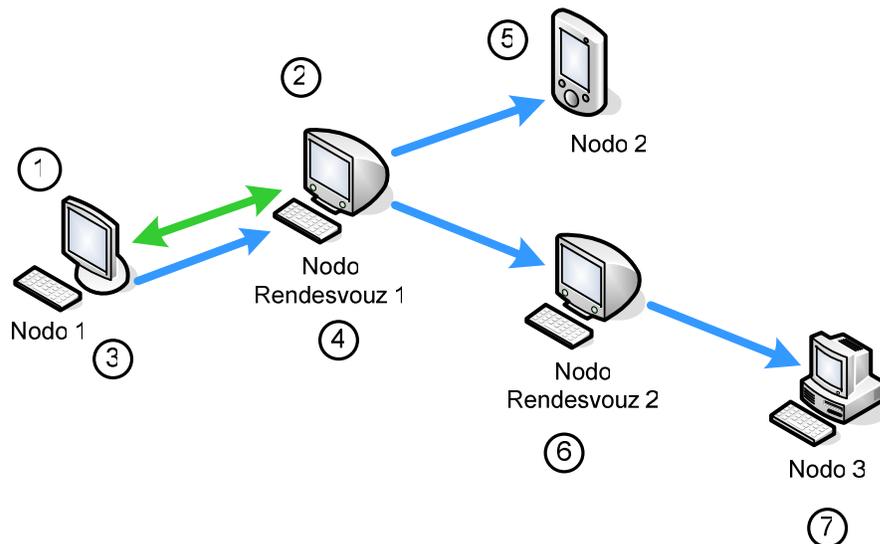
Así como el protocolo *Discovery* se basa sobre el protocolo *Resolver*, el *Resolver* a su vez depende del protocolo *Rendezvous*. Este último provee a servicios a los nodos simples de la red, incluyendo la propagación de mensajes a otros miembros del grupo fuera de la red privada.

Los nodos simples, para poder propagar mensajes mediante un nodo *rendezvous*, deben obtener un permiso. El permiso especifica la cantidad de tiempo que un nodo solicitante puede usar al *rendezvous* antes de renovar la conexión.

Para proveer esta funcionalidad a los nodos simples, se han diseñado tres mensajes:

- **Mensaje Lease Request:** Un mensaje usado por un nodo simple para solicitar un permiso de conexión al nodo *rendezvous*.
- **Mensaje Lease Granted:** Un mensaje usado por el nodo *rendezvous* para aprobar la solicitud de conexión de un nodo simple, y al mismo tiempo proveer el tiempo de conexión.
- **Mensaje Lease Cancel:** Un mensaje usado por un nodo simple para desconectarse del nodo *rendezvous*.

Figura 31. Interacción de mensajes *rendezvous*



Fuente: Brendon J. Wilson. *JXTA*. New Riders Publishing. Primera edición. 2002.

En la figura 31 se puede apreciar el intercambio de mensajes de un nodo simple con un nodo *rendezvous*. A continuación se describen los pasos:

1. El nodo 1 desea propagar un mensaje dentro del grupo de nodos. Primero debe obtener un permiso de conexión enviando un mensaje *Lease Request*.
2. Al recibir el mensaje, el nodo *rendezvous* decide si otorga el permiso de conexión. Si lo hace, genera el permiso y regresa un mensaje *Lease Granted* al nodo 1.
3. El nodo 1 recibe el permiso. Ahora ya puede enviar mensajes al nodo *rendezvous* para propagarlos a otros nodos.
4. Cuando el nodo *rendezvous* recibe un mensaje para propagar, verifica antes si la fuente ya fue autorizada. Si lo fue, propaga el mensaje a todos los nodos que también poseen un permiso de conexión con él. El mensaje también se propagará en la red local usando *multicasting*.
5. Los nodos reciben el mensaje propagado y rutean el mensaje al servicio especificado en el mensaje.
6. Un nodo *Rendezvous* también puede ser un cliente de otro nodo *Rendezvous*. En este caso, el nodo *Rendezvous 1* está conectado al 2, y usa el nodo *Rendezvous 2* para propagar el mensaje a los nodos que están conectados al 2.
7. Inclusive los nodos simples propagan el mensaje, solo que se limitan a propagarlo a la red local.

Antes de conectarse a un nodo *Rendezvous*, un nodo simple debe poder descubrirlo antes al encontrar el anuncio del nodo *Rendezvous*. Los nodos que

desean funcionar como nodos *Rendezvous* deben anunciar sus capacidades publicando un anuncio *Rendezvous* dentro de los nodos de la red P2P.

Este mensaje contiene 3 elementos. ***RdvGroupId*** especifica el identificador del grupo de nodos a los cuales se ofrece el servicio. ***RdvPeerId*** que especifica el identificador del nodo que provee el servicio y ***Name*** que especifica un nombre para el nodo *Rendezvous*.

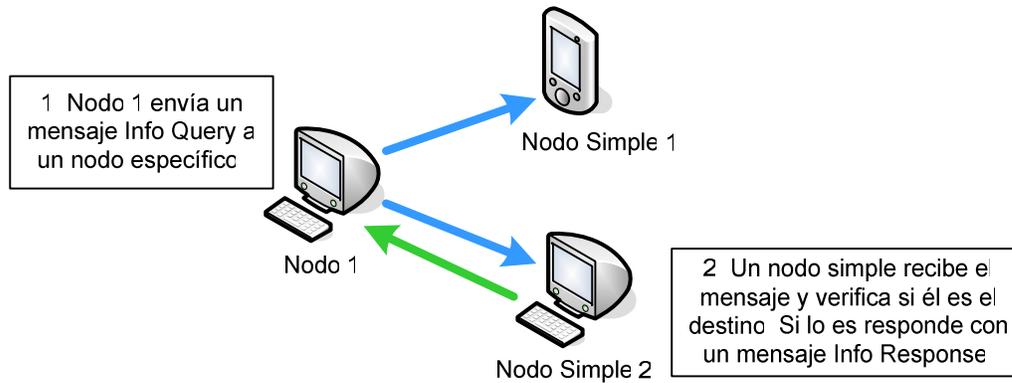
4.3.4 Peer Information

Una vez que un nodo remoto ha sido descubierto utilizando el protocolo *Peer Discovery*, un nodo podría monitorear el estado del nodo remoto para tomar decisiones adicionales acerca de la forma más óptima de usar el nodo remoto. Monitorear es una parte esencial de una red *P2P*, y es lo que permite a los nodos compartir sus recursos.

El protocolo *Peer Information* es un protocolo opcional de *JXTA* que permite a los nodos monitorear un nodo remoto y obtener información del estado actual del nodo remoto. Este protocolo se compone de dos mensajes:

- **Mensaje *Info Query***: un mensaje para consultar el estado de un nodo remoto.
- **Mensaje *Info Response***: un mensaje para proveer el estado de un nodo a otros nodos.

Figura 32. Funcionamiento del protocolo *Peer Information*



Fuente: Brendon J. Wilson. *JXTA. New Riders Publishing*. Primera edición. 2002.

4.3.5 Pipe Binding

Un *Pipe* es un enlace en *JXTA* que permite enviar o recibir datos de un nodo remoto. Antes de utilizar un enlace, éste debe ser vinculado a un nodo. Al vincularlo, se puede crear un *Input Pipe* (enlace de entrada) para recibir datos o un *Output Pipe* (enlace de salida) para enviar datos.

El proceso de vincular un *Pipe* a un nodo, se define por el protocolo *Pipe Binding*. Por definición, estos enlaces son unidireccionales, lo que quiere decir que la información viaja solo en una dirección. Los enlaces también son asíncronos, queriendo decir que los datos pueden ser enviados o recibidos en cualquier momento, lo que permite a los nodos trabajar independientemente sin esperar un estado de sincronización.

Los enlaces son descritos utilizando un anuncio de enlace, utilizando el *XML* que se muestra a continuación:

```

<?xml version="1.0" encoding="UTF-8"?>
<jxta:PipeAdvertisement>
<Id> . . . </Id>
<Type> . . . </Type>
<Name> . . . </Name>
</jxta:PipeAdvertisement>

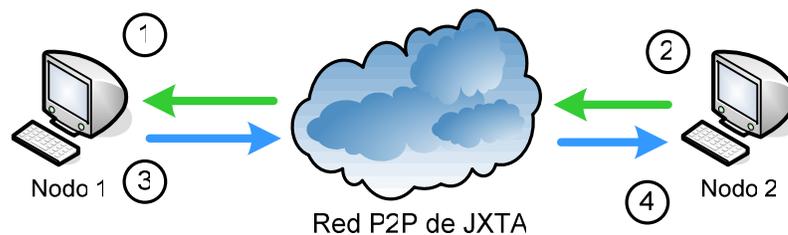
```

Un punto clave aquí, es que el anuncio no contiene el identificador del nodo. Esto permite a varios nodos proveer acceso a un servicio usando el mismo anuncio. Cuando un nodo desea enviar datos usando un enlace, necesita encontrar un nodo que ya tenga un enlace vinculado con el mismo identificador de enlace y que este escuchando datos. El protocolo define dos mensajes que permiten buscar un *Pipe*:

- **Mensaje *Pipe Binding Query*:** un mensaje para consultar a un nodo remoto si posee un *Pipe* o enlace con el identificador dado.
- **Mensaje *Pipe Binding Answer*:** un mensaje para enviar respuestas a las consultas del mensaje anterior.

El funcionamiento de estos mensajes se puede apreciar en la figura 33 que se muestra a continuación.

Figura 33. Funcionamiento del protocolo *Pipe Binding*



Fuente: Brendon J. Wilson. *JXTA*. New Riders Publishing. Primera edición. 2002.

1. El Nodo1 crea un *Pipe* de entrada de un anuncio y espera mensajes.
2. El Nodo2, queriendo enviar un mensaje al Nodo1, usando el mismo anuncio, necesita crear un enlace de salida. Para hacer esto, envía un mensaje *Pipe Binding Query* a todos sus nodos conocidos.
3. El Nodo1 recibe el mensaje, verifica en su cache de *Pipes* para ver si coincide. En caso afirmativo, responde con un mensaje *Pipe Binding Answer* conteniendo el anuncio de su enlace.
4. El Nodo 2 recibe el mensaje de respuesta y extrae la información del anuncio del enlace. La información es usada para crear un enlace de salida. Al hacer esto, el Nodo 2 ya puede enviar mensajes al Nodo 1.

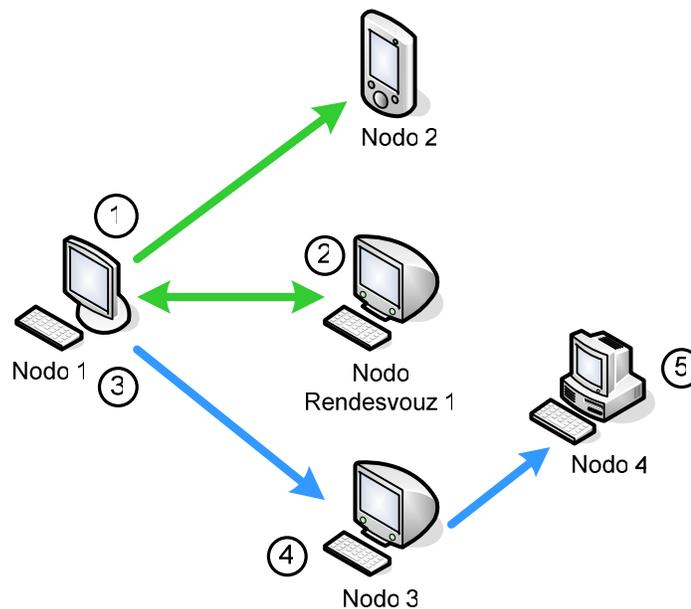
4.3.6 Endpoint Routing

En las redes *P2P*, un mensaje entre dos nodos puede viajar a través de intermediarios. Un intermediario puede ser utilizado para permitir a dos nodos con transportes de red incompatibles comunicarse, al utilizar el intermediario como puerta de enlace. El protocolo *Endpoint Routing* fue diseñado para este propósito y provee un mecanismo para obtener la ruta a un nodo.

Este intermediario o nodo de ruteo, debe ser conocido por los nodos, debido a que en la red, los nodos pueden entrar y salir espontáneamente, los nodos necesitan un mecanismo de ruteo que funcione en estas condiciones en que los nodos de ruteo cambian constantemente.

Si dos nodos no pueden comunicarse directamente, el protocolo *Endpoint Routing* ayuda a los nodos a encontrar una forma de descubrir cómo pueden enviarse mensajes mediante un intermediario.

Figura 34. Funcionamiento del protocolo *Endpoint Routing*



Fuente: Brendon J. Wilson. *JXTA*. New Riders Publishing. Primera edición. 2002.

En la figura 34 se puede apreciar el funcionamiento de este protocolo.

1. El nodo 1 desea enviar un mensaje al nodo 4, pero no puede conectarse directamente con él. Entonces envía un mensaje *Route Query* a todos sus nodos conocidos para tratar de determinar una ruta.
2. El nodo *rendezvous* conoce una ruta al nodo 4, y envía un mensaje *Route Response* al nodo 1 con esta información.

3. El nodo 1 recibe el mensaje y agrega un mensaje *Endpoint Router* al mensaje que desea enviar al nodo 4. Envía este mensaje a la primera dirección del nodo retornada en el mensaje de la ruta.
4. El nodo 3 recibe el mensaje *Endpoint Router* y determina el siguiente nodo en la ruta, y envía el mensaje al siguiente nodo en la lista.
5. El nodo 4 recibe el mensaje *Endpoint Router* y determina que es el destino final del mensaje. El mensaje original es extraído y procesado por el servicio apropiado.

4.4 Ventajas y desventajas

4.4.1 Ventajas

JXTA provee un lenguaje mucho más abstracto para comunicación entre nodos que cualquier protocolo *P2P* antes creado. Lo que permite usar una mayor variedad de servicios, dispositivos y transportes de red.

El lenguaje *XML* provee un formato estándar para estructurar datos que es fácil de entender y transportable. *XML* también tiene la ventaja que es entendible para las personas, lo que facilita a los desarrolladores depurar las aplicaciones.

4.4.2 Desventajas

No se especifica como los servicios (a excepción de los servicios base) son invocados. Existen varios estándares que definen la invocación de servicios, por ejemplo el *WSDL* (*Web Services Description Language*).

La abstracción del transporte de red es demasiada. Ya que si todas las aplicaciones *P2P* actuales usan *TCP* como medio de transporte, ¿porque *JXTA* no especifica directamente a *TCP* como transporte de red?

Todo esto realza la necesidad de los desarrolladores para balancear flexibilidad con rendimiento al implementar sus aplicaciones *P2P*. Puede que *JXTA* no sea la mejor o más eficiente solución para implementar una aplicación

P2P particular, pero *JXTA* provee la plataforma más completa para crear aplicaciones *P2P* que posean la flexibilidad para crecer en el futuro.

El valor de *JXTA* radica en la capacidad de levantar otros servicios *P2P* y habilitar el desarrollo de comunidades *P2P*.

4.5 Comparación con *Jini* y *.NET*

La promesa de interconectar cualquier tipo de dispositivo sobre cualquier tipo de red, puede sonar familiar a los seguidores de la tecnología *Jini* de *Sun*. Sin embargo *Jini* funciona exclusivamente sobre la plataforma *Java*, mientras que *JXTA* no tiene dependencia en un lenguaje de programación en particular. Además *Jini* usa un servidor centralizado para localizar los servicios en la red y depende de *RMI* (*Remote Method Invocation*) para la comunicación remota.

El aspecto de servicios Web de la plataforma *Microsoft .NET*, está basado fuertemente en *XML*, pero el uso *XML* no los hace comparables, ya que *.NET* y *JXTA* tienen objetivos completamente diferentes, por un lado *.NET* se enfoca en la arquitectura tradicional cliente/servidor, mientras que *JXTA* no.

5. IMPLEMENTACIÓN DE LA APLICACIÓN *P2P* ORIENTADA A PYMES, PARA LA OFERTA Y DEMANDA DE PRODUCTOS Y SERVICIOS

5.1 Introducción

El objetivo de este trabajo es el desarrollo de una aplicación *P2P* dirigido a PYMES para la oferta y demanda de sus productos y servicios, haciendo uso de la Ontología de actividades económicas. A continuación describiremos la funcionalidad y arquitectura de la aplicación *P2P* llamada Ontobiz y sus diferentes componentes. Se iniciara explorando lo que motivo la creación de esta aplicación, pasando luego por el análisis de su funcionalidad y finalmente el diseño de la misma.

5.2 Definición del problema

Actualmente no existen aplicaciones basadas en el *P2P* gratuitas que permitan facilitar negocios entre Pequeñas y Medianas Empresas. Seguro que existen portales Web que sirven como intermediarios para conectar negocios, sin embargo, por ser soluciones propietarias, algunos requieren el pago de comisiones por el derecho de pertenecer a ellas, además que por estar basados en el modelo cliente/servidor, presentan ciertas limitantes que pueden afectar la escalabilidad y rendimiento.

Se necesita una aplicación versátil que pueda llenar ese vacío y ayudar a las Pequeñas y Medianas Empresas a establecer relaciones de negocio a nivel latinoamericano.

5.3 Análisis

5.3.1 Objetivos

Según la clasificación dentro del *P2P*, esta aplicación pertenece al modelo híbrido, por lo que utilizara una aplicación servidor que indexara los nodos conectados a la red *P2P* y una aplicación cliente que se usara para ofertar productos/servicios y realizar búsquedas.

5.3.1.1 Objetivos del nodo servidor

- Indexar los nodos que entran y salen de la red *P2P*.
- Facilitar la comunicación entre los nodos de la red.
- Almacenar las transacciones realizadas entre los nodos, incluyendo búsquedas, compras y ventas.
- Reenvío de mensajes de compra/venta a los nodos que se desconectaron de la red.

5.3.1.2 Objetivos del nodo cliente

- Conectarse o desconectarse a la red *P2P*.
- Publicar los productos y servicios de la empresa.
- Realizar búsquedas de productos/servicios en de la red.
- Enviar peticiones de compra a otros nodos de los productos o servicios que estos ofrecen.

- Responder las peticiones de búsqueda y de compra de los otros nodos.

5.3.2 Beneficios

En la siguiente tabla se resumen los beneficios que ofrece la aplicación *OntoBiz* a las empresas.

Tabla X. Beneficios de la aplicación *OntoBiz*

Beneficios	Descripción
Usabilidad	Posee una interfaz amigable que permiten administrar la información de los productos y servicios publicados así como de las compras y ventas realizadas. Cuenta con un agente de búsqueda inteligente que usa una Ontología para encontrar los resultados.
Intercambio y distribución	Las empresas pueden vender directamente a otras empresas sin necesidad de un intermediario, con esto las empresas estarán haciendo comercio electrónico de negocio a negocio, <i>B2B</i> .
Mercadeo y comunicaciones	Permite alcanzar nuevos mercados que no es posible alcanzar utilizando los medios tradicionales, como la radio, televisión y la prensa.
Colaboración	Facilidad de compartir información de los productos y servicios en el grupo <i>OntoBiz</i> . Permite reconocer las nuevas oportunidades que surgen.

5.3.3 Funciones de la aplicación cliente

La aplicación cliente dispone de tres funciones básicas que le permiten interactuar con los demás nodos de la red: publicar anuncios, realizar búsquedas y realizar transacciones.

5.3.3.1 Publicar anuncios

Con esta función, el usuario de la empresa, podrá registrar y administrar todos los productos y servicios que serán publicados en la red. Podrá crear un nombre para el producto o servicio, escribir una descripción general, clasificarlo según la Ontología, opcionalmente puede asignarle una imagen y finalmente publicarlo en la red.

Restricciones

- Los productos y servicios a ofrecer, deben estar dentro del dominio de los conceptos de la Ontología de Actividades Económicas y Productos.
- El nombre asignado a cada producto o servicio debe ser único para la empresa que lo publica, es decir una misma empresa no puede tener más de un producto o servicio con el mismo nombre.

5.3.3.2 Realizar búsquedas

Esta función es la más importante de la aplicación y permite al usuario de la empresa, realizar búsquedas de productos y servicios que otras empresas ofrecen dentro de la red *OntoBiz*.

Restricciones

- Solo se pueden realizar búsquedas sobre los campos indexados de cada producto y servicio, los cuales son: tipo, nombre, marca, descripción, clasificación y precio.
- Los resultados obtenidos al realizar una búsqueda, no corresponden al 100% del contenido de la red *P2P*. Esto se debe a la naturaleza de las redes *P2P*, y a la implementación óptima de búsquedas que ofrece la plataforma *JXTA*. Esto ayuda a no sobrecargar el ancho de banda de la red y a su vez permite escalar la red con muchos nodos.
- Los resultados obtenidos pueden ser distintos al realizar la misma petición de productos o servicios en búsquedas distintas. Esto se debe también a la naturaleza de las redes *P2P* y a la implementación ofrecida por la plataforma *JXTA*.

5.3.3.3 Realizar transacciones

Esta función permitirá a la empresa realizar transacciones de compra y venta de los productos y servicios disponibles en la red *P2P*. Al realizar una búsqueda, si encontramos un producto o servicio de nuestro interés, podemos ver el detalle y enviar una solicitud de compra al nodo ofertante. También podremos responder solicitudes de compra de nuestros productos y servicios provenientes de los otros nodos.

Restricciones

- Al realizar una solicitud de compra, no se garantiza que la transacción se vaya a completar, esto depende de varios factores: si el nodo ofertante no se vuelve a conectar a la red, este nunca recibirá la solicitud, si la recibe puede que nunca la responda, entonces al final la transacción depende del nodo que publica la oferta.
- El nodo vendedor es el responsable de darle seguimiento a todas las solicitudes de compra y este debe perseguir poder realizar la venta. Al recibir una solicitud de compra, la información del nodo comprador es adjuntada a la solicitud, esto permitirá al vendedor ponerse en contacto con el comprador utilizando los medios de comunicación actuales (teléfono e email).
- Si el nodo ofertante se desconecta de la red *P2P* antes de poder recibir alguna solicitud de compra, el sistema *P2P* tiene la capacidad de entregar el mensaje al nodo ofertante a través del servidor, el cual entregara el mensaje cuando el nodo se vuelva a conectar, siempre y cuando no transcurran más de 3 días desde que se envió la solicitud (valor parametrizable en el servidor).

5.3.4 Funciones de la aplicación servidor

La aplicación servidor dispone de tres funciones básicas que le permiten interactuar con los demás nodos de la red: registrar nodos conectados, registrar transacciones y reenvío de solicitudes.

5.3.4.1 Registrar nodos conectados

Esta función es la base para que funcione la red *P2P*, cada vez que un nodo desea conectarse a la red, contacta primero al servidor, esta conexión se establece usando la plataforma *JXTA*, el servidor administra una lista de nodos conectados, y el nuevo nodo queda registrado, si el nodo se desconecta de la red, el servidor lo detecta y lo remueve de su lista.

5.3.4.2 Registrar transacciones

El servidor es capaz de registrar cada transacción que es realizada en la red *P2P*, las transacciones registradas incluyen: búsquedas realizadas por los nodos así como las solicitudes de compras enviadas y las ventas realizadas.

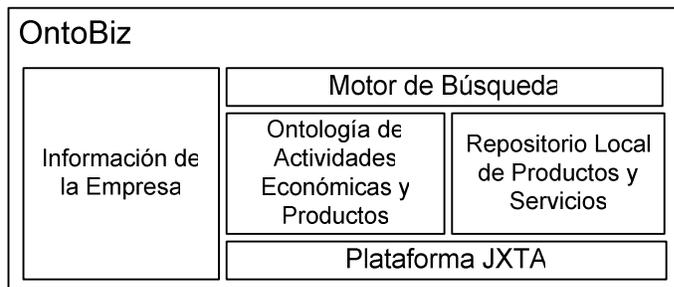
5.3.4.3 Reenvío de solicitudes

Cuando un nodo envía una solicitud de compra o envía una respuesta a una solicitud recibida y el nodo destino ya no se encuentra conectado a la red *P2P*, la aplicación tiene un mecanismo para asegurar la entrega de dicho mensaje, utilizando al servidor como intermediario, el servidor podrá almacenar temporalmente hasta 3 días un mensaje de este tipo y entregarlo al nodo destinatario cuando este se conecte de nuevo a la red *P2P*.

5.4 Diseño

El sistema *OntoBiz* se compone de cinco módulos, el primero de ellos lo constituye la plataforma *JXTA*, este es responsable de todas las funciones relacionadas con el acceso a la red de nodos. El segundo módulo es el registro seguro de la información de la empresa, responsable de encriptar y almacenar los datos generales de la empresa y es utilizado al realizar transacciones entre nodos. El tercer módulo lo constituye el motor de búsqueda, responsable de realizar búsquedas de productos y servicios en la red. El cuarto módulo es la Ontología de Actividades Económicas y Productos, responsable de clasificar los productos y servicios que se ofrecen en la red así como facilitar las búsquedas. Finalmente tenemos el módulo del Repositorio Local de Productos y Servicios, responsable de almacenar la información de productos y servicios que la empresa ofrece como de las transacciones realizadas.

Figura 35. Componentes del sistema OntoBiz.



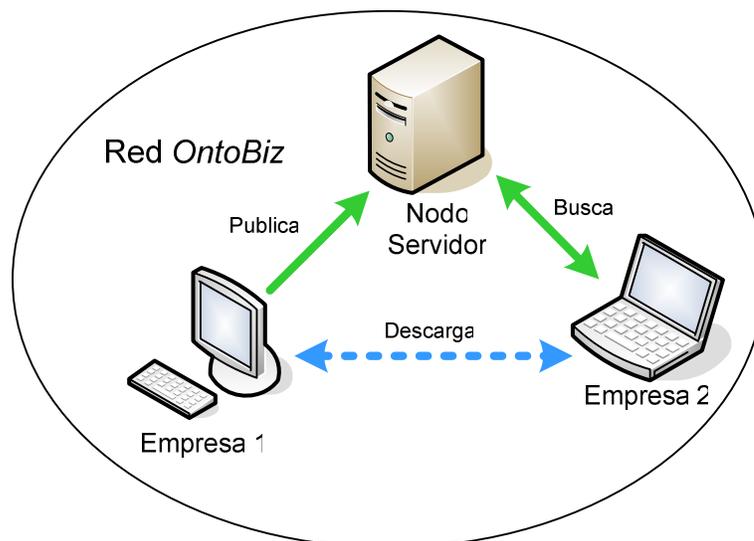
Como se puede observar en la Figura 35, el motor de búsqueda depende de la Ontología y del registro local de productos y servicios para funcionar, éstos a su vez dependen de la plataforma *JXTA*.

5.4.1 Entorno

Para el funcionamiento de la aplicación, se han identificado los siguientes actores:

- Usuario de empresa: responsable de utilizar la aplicación cliente para registrar los datos de la empresa y publicar los productos y servicios a ofrecer.
- Administrador: responsable de administrar el servidor para darle mantenimiento y verificar su correcto funcionamiento.
- Nodo Cliente: representa una aplicación cliente dentro de la red *P2P*.
- Nodo Servidor: representa el nodo servidor que registra los nodos que entran y salen de la red.

Figura 36. Diagrama de la red virtual de *OntoBiz*.



5.4.1 Herramientas de desarrollo

Para el desarrollo de ésta aplicación, utilizamos el conjunto de herramientas que se listan en la tabla XI.

Tabla XI. Herramientas usadas para el desarrollo de la aplicación P2P

Herramienta	Tipo	Descripción
<i>Netbeans</i> 6.1	<i>IDE</i>	Poderoso entorno de desarrollo de aplicaciones Java. Disponible en: http://www.netbeans.org
<i>JSE</i> 1.5.0_06	Plataforma	Plataforma Java para desarrollo de aplicaciones Web y escritorio. Disponible en: http://java.sun.com
<i>JXSE</i> 2.5	<i>API</i>	<i>API</i> de <i>JXTA</i> para el lenguaje de programación Java, incluye las librerías necesarias para el desarrollo de aplicaciones <i>P2P</i> . Disponible en: http://www.jxta.org
<i>QUAQUA</i> v2.9.5	<i>Look n Feel</i>	Librería para Java, que mejora la apariencia de la interfaz grafica de la aplicación. Disponible en: http://www.randelshofer.ch/

Todas estas herramientas son gratuitas y están disponibles en Internet para que cualquier desarrollador pueda rápidamente empezar a crear poderosas aplicaciones multiplataforma.

5.4.2 Elementos usados de JXTA

Se eligió la plataforma *JXTA* para desarrollar la aplicación Peer-to-Peer, ya que provee una poderosa API (*Application Programming Interface*, por sus siglas en inglés) de alto nivel, que permite desplegar aplicaciones *P2P* con diferentes configuraciones, según las necesidades específicas de la red.

Para el desarrollo de la aplicación, utilizamos el proyecto *JXSE 2.5*, que es una implementación de los protocolos *JXTA* sobre la plataforma *Java Standard Edition 5.0 (JSE)* [47].

Para poder crear una aplicación *JXTA*, debemos utilizar ciertos elementos básicos que permiten la interacción dentro de la red, estos elementos básicos son: grupos, nodos, anuncios y canales de comunicación.

5.4.2.1 Grupo *OntoBiz*

Antes que nada, la API de *JXTA* sugiere la creación de grupos, ya que un grupo reúne a uno o más nodos que tienen un objetivo en común, para nuestro caso, nuestro objetivo es permitir a las *PYMES* ofertar y demandar productos y servicios. Para lo cual creamos un grupo llamado *OntoBiz*, y cada Nodo de nuestra red, deberá localizar el grupo y unirse a él.

5.4.2.2 Nodos

Según la definición de *JXTA*, un nodo es una entidad dentro de la red que implementa uno o más protocolos de *JXTA*. Cada instancia de nuestra

aplicación que será ejecutada por una empresa distinta, representa un nodo de la red. Según la clasificación de *JXTA*, nuestra aplicación es un nodo simple. Para poder ofertar productos y servicios, se deben registrar los datos generales de la empresa, y para poder realizar búsquedas, cada nodo debe unirse primero al grupo *OntoBiz*. Los nodos pueden entrar y salir de la red constantemente, sin afectar el funcionamiento de la red *P2P*, sin embargo la cantidad de resultados obtenidos al realizar búsquedas, depende directamente de la cantidad de nodos conectados y de los productos y servicios que estos ofrezcan.

5.4.2.3 Anuncios

Los anuncios son uno de los elementos más poderosos e importantes de la plataforma *JXTA*. Según *JXTA*, un anuncio es un documento *XML* que describe y publica la existencia de un recurso de un nodo en la red. Son esenciales ya que permite localizar nodos y recursos dentro del grupo *P2P*, para el caso de la aplicación, cada nodo conectado es identificado utilizando un anuncio, el cual otros nodos pueden utilizar para comunicarse y poder buscar y localizar el producto o servicio de su interés.

5.4.2.4 Canales de comunicación (*Pipes*)

Los canales de comunicación, conocidos como Pipes según la terminología de *JXTA*, sirven para enviar mensajes de comunicación entre los distintos nodos. Según *JXTA*, un Pipe es un mecanismo de envío de mensajes asíncrono, unidireccional y no-confiable usada para comunicación y transferencia de datos. Estos mensajes son enviados de forma asíncrona, y son utilizados para enviar y recibir información de un nodo a otro.

CONCLUSIONES

1. Los negocios electrónicos juegan un papel muy importante en las empresas hoy en día, desde utilizar el correo electrónico para facilitar negocios hasta la utilización de programas de computación para optimizar procesos internos, esto ha sido gracias al avance de la tecnología, y cada día tienen más impacto en todos los aspectos de nuestras vidas.
2. Las Ontologías son una poderosa herramienta que ayuda a las personas y computadoras a darle un significado a la información, cada día esto cobra más importancia, debido al crecimiento exponencial de la información, necesitamos Ontologías para clasificar y darle significado a esta información y con esto desarrollar herramientas de búsqueda inteligentes. En este caso utilizamos una Ontología de actividades económicas y productos para alcanzar nuestro objetivo.
3. Las redes *Peer-to-Peer* cada vez juegan un papel más importante para los usuarios de Internet, lo que empezó como una violación a los derechos de autor al compartir contenido musical con *Napster*, ahora se ha convertido en una tecnología muy cotizada, debido a su potencial escalabilidad, ahora tenemos sistemas *P2P* para diferentes propósitos, desde usar mensajería instantánea hasta sistemas de búsqueda extra-terrestre.

4. Con toda esta explosión de redes *Peer-to-Peer*, han surgido decenas de protocolos de comunicación, debido a que cada protocolo es propietario del desarrollador, las aplicaciones no pueden colaborar entre sí. Es por eso que utilizamos la plataforma *JXTA*, ya que ofrece una API poderosa y multiplataforma basada en *Java* para el despliegue de aplicaciones *Peer-to-Peer* que puedan colaborar entre sí.

5. El futuro de las aplicaciones orientadas hacia los negocios e Internet, radica en la convergencia de diferentes tecnologías alineadas para conseguir un mismo objetivo. *OntoBiz* pertenece a una nueva generación de aplicaciones que utiliza los avances más recientes de la Web Semántica y las redes *Peer-to-Peer*, con el objetivo de facilitar negocios, ofertando productos y servicios a nivel latinoamericano.

RECOMENDACIONES

1. Para que un negocio sea exitoso y competitivo hoy en día, es necesario que utilice la tecnología y aproveche todas las ventajas que le puede ofrecer en todos los niveles de la empresa, desde utilizar el correo electrónico hasta ofrecer a sus clientes y proveedores servicios en línea que le permitan diferenciarse de la competencia.
2. Lograr que la visión de Tim Berners Lee, el padre de la Web Semántica, se haga realidad, es un reto que absolutamente todos debemos asumir, no basta con saber de ella, debemos colaborar y buscar la forma de hacerla realidad, nos enfrentamos a la Web, que ahora es una montaña de información, y si no tenemos mecanismos que nos permitan manejar y buscar dentro de esta información, de nada nos va a servir toda la información del mundo sino podemos llegar a ella y compartirla.
3. Es necesario desarrollar Ontologías que expliquen los dominios de información que existen en el mundo, y no solo eso, sino que también se puedan relacionar dichas Ontologías entre sí, con el objetivo de que la información sea compartida y las herramientas de búsqueda trabajen de manera conjunta.
4. La red Ontobiz tiene mucho potencial y queda abierta a futuras aplicaciones, una de ellas es la minería de datos con la información recolectada en el servidor, se pueden generar estudios de mercado del comportamiento de las empresas según producto o región. También queda abierta la posibilidad de integrar a la red otros programas para colaborar entre sí y lograr nuevas funciones.

REFERENCIAS

1. Australian Government. **An Australian guide to doing eBusiness on line**. Disponible: <http://www.e-businessguide.gov.au>
2. Portal Intuition. **eBusiness, advice, management and support**. Disponible: <http://www.in-tuition.co.uk>
3. Smith, Eduardo. **Impulsan comercio electrónico**. Prensa Libre. Viernes 14 octubre 2005
4. Cámara de Comercio de Guatemala. Disponible: <http://www.negociosenguatemala.com/ebusiness>
5. Bancasat. Portal Web. Disponible: <http://www.bancasat.com>
6. GuateCompras. Portal Web. Disponible: <http://guatecompras.gt/>
7. Pérez, Leslie. **En busca de una sociedad en línea**. Prensa Libre. Domingo 1 de abril de 2007
8. Deguate. Portal Web. Disponible: <http://www.deguate.com/servicios/index.shtml>. Visitado 27/11/2008
9. Dell Inc. Disponible: <http://en.wikipedia.org/wiki/DELL>. Visitado el 27 de noviembre de 2008
10. Tello, Adolfo Lozano, **Ontologías en la Web Semántica**
11. W3C. **Semantic Web**. Disponible: <http://www.w3.org/2001/sw/>
12. Berners-Lee, Tim. **Weaving the Web**. Harper San Francisco, 1999
13. Berners-Lee, Tim. **Scientific American**, Mayo 2001
14. Daconta, Michael C. y otros, **The Semantic Web, A Guide to the Future of XML, Web Services, and Knowledge Management**
15. Redondo, Peis y otros. **Ontologías, metadatos y agentes: recuperación "semántica" de la información**. Universidad de Granada, España.

16. López, Jorge E. y otros. **Gestión Semántica: Aplicando las Ontologías a la Gestión de Red**
17. Noy, Natalya F., McGuinness, Deborah L. **Ontology Development 101: A Guide to Creating Your First Ontology**. Stanford University
18. D. Fensel. **Ontologies: Silver bullet for knowledge management and electronic commerce**. Springer-Verlag, Berlin, 2001
19. WRC. **OWL Web Ontology Language, Use Cases and Requirements**. Disponible: <http://www.w3.org/TR/webont-req/>
20. OntoWeb. Disponible: <http://ontoweb.aifb.uni-karlsruhe.de/>
21. Agencities. Disponible: <http://www.agencities.org/>
22. Barrón Cedeño, Alberto. **El Web Semántico**. Septiembre 2005
23. Corcho, Oscar y Gómez-Pérez, Asunción. **A Roadmap to Ontology Specification Languages**
24. OIL. **Ontology Inference Layer**. Disponible: <http://www.ontoknowledge.org/oil/>
25. WRC. **OWL Web Ontology Language Overview**. 10 Febrero 2004. Disponible: <http://www.w3.org/TR/owl-features/>
26. Protégé. **Ontology Editor and Knowledge Aquisition System**. Disponible: <http://protege.stanford.edu>
27. Sean Bechhofer y otros. **OilEd: a Reasonable Ontology Editor for the Semantic Web**.
28. Richard MacManus. **10 Semantic Apps to Watch**. Disponible: http://www.readwriteweb.com/archives/10_semantic_apps_to_watch. Visitado el 29 de noviembre de 2008.
29. Iskold, Alex. **Top-Down: A New Approach to the Semantic Web**. Disponible: http://www.readwriteweb.com/archives/the_top-down_semantic_web.php, Visitado el 29 de noviembre de 2008.
30. Freebase. Disponible: <http://www.freebase.com>. Visitado el 29 de noviembre de 2008.

31. Powerset. **Natural Language Search**. Disponible: <http://www.powerset.com>, Visitado el 29 de noviembre de 2008.
32. Twine. **Tie it all together**. Disponible: <http://www.twine.com>. Visitado el 29 de noviembre de 2008.
33. Hakia. **Search for meaning**. Disponible: <http://www.hakia.com> Visitado el 29 de noviembre de 2008.
34. SmartLinks. **Browse smarter**. Disponible: <http://www.adaptiveblue.com>, Visitado el 29 de noviembre de 2008.
35. TrueKnowledge. **Direct answers to humans and machina questions**. Disponible: <http://trueknowledge.com>
36. TripIt. **Organize your travel**. Disponible: <http://www.tripit.com> Visitado el 29 de noviembre de 2008.
37. Spock. **It's about people**. Disponible: <http://www.spock.com> Visitado el 29 de noviembre de 2008.
38. Brendon J. Wilson. **JXTA**. New Riders Publishing. Primera edición. 2002
39. Gnutella. Disponible: <http://en.wikipedia.org/wiki/Gnutella>
40. ICQ. **I Seek You**. Disponible: <http://www.icq.com>
41. Freenet. Disponible: <http://freenetproject.org>
42. Aimster. Disponible: <http://en.wikipedia.org/wiki/Madster>
43. KaZaa. Disponible: <http://es.wikipedia.org/wiki/Kazaa>
44. OpenCola. Disponible: <http://en.wikipedia.org/wiki/Opencola>
45. Civera, Roger. **P2P: Ser Negocio o No Ser**. Disponible: <http://www.improven-consultores.com/pdf/p2p.pdf>
46. SETI@home. **Search for ExtraTerrestrial Intelligence**. Disponible: <http://seti.astroseti.org/setiathome/>
47. JXTA. Sitio Web oficial. Disponible: <https://jxta.dev.java.net/>

BIBLIOGRAFÍA

1. Abian, Miguel Angel, **EL FUTURO DE LA WEB, XML, RDF/RDFS, ONTOLOGÍAS Y LA WEB SEMÁNTICA.**
2. Biddle, Peter y otros. ***The darknet and the future of content distribution.*** Microsoft Corporation. 31 octubre del 2003.
3. Chang, David. **Análisis e implementación de una Ontología de Actividades Económicas y Productos para PYMES de Guatemala.** Guatemala, junio de 2007.
4. Laudon, Kenneth C., Traver, Carol G. ***E-commerce: business. Technology. Society*** (2002)
5. Milojicic, Dejan S. y otros. **Peer-to-Peer Computing.** HP Laboratories.
Disponible: <http://www.hpl.hp.com/techreports/2002/HPL-2002-57R1.pdf>
6. Sun Microsystems Inc. **JXTA Java Standard Edition v2.5: Programmers Guide.** 10 de septiembre de 2007

APÉNDICE

A. Manual de usuario

A.1 Requerimientos mínimos

Para el correcto funcionamiento de la aplicación, se deben cumplir los siguientes requerimientos:

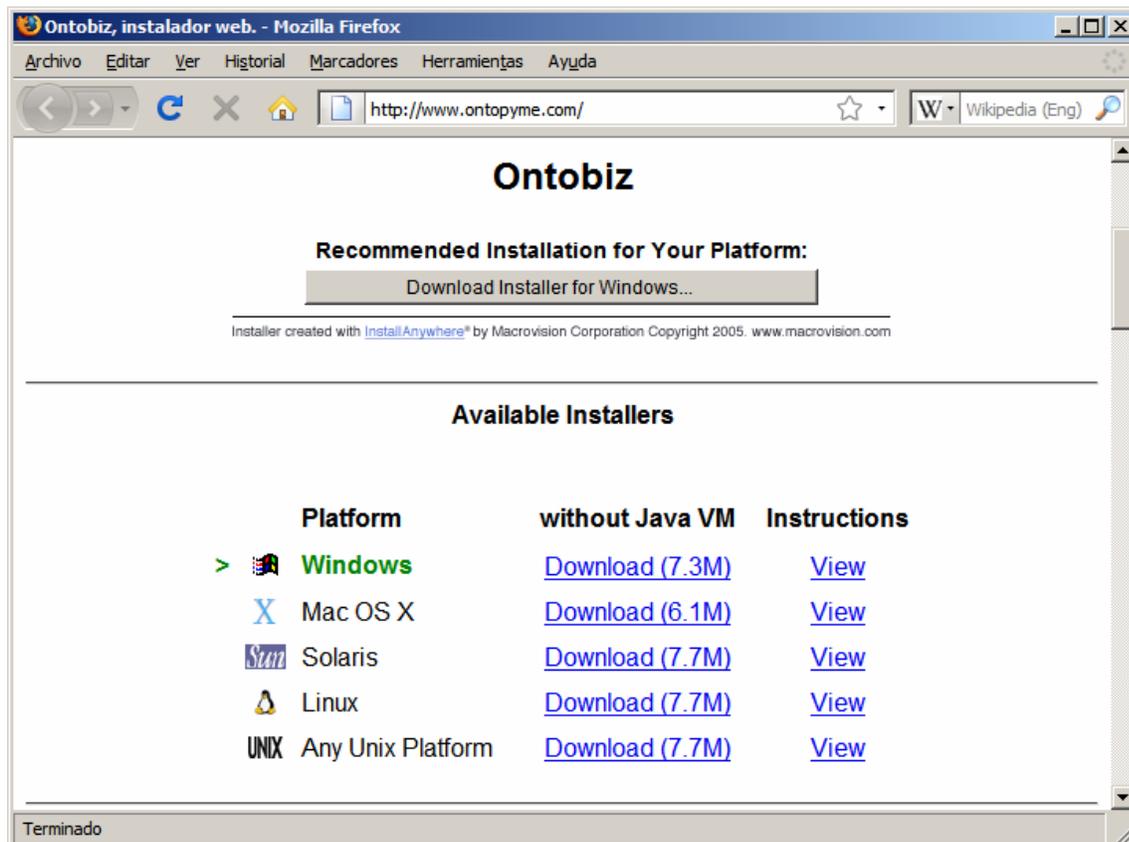
1. Máquina virtual de *Java* versión 1.5 o superior.
2. Sistema operativo: *Windows98/XP/Vista, Linux, Mac OS, o Solaris.*
3. Memoria: 128 *MB*
4. Espacio en disco: 200 *MB*
5. Enlace de Internet: 64 *kbps*

A.2 Instalación

Para instalar la aplicación, primero debe descargar el instalador de Internet, el cual se encuentra disponible en la siguiente dirección web: www.ontopyme.com, puede llegar a esta dirección utilizando su navegador web preferido. Al abrir el sitio, se mostrara la pagina principal donde podrá descargar el instalador apropiado para su sistema operativo, si tiene java instalado, la pagina le sugerirá automáticamente el instalador apropiado, para lo que solo deberá hacer clic en el botón *Download Installer*. Así como se muestra en la figura 37. Si su navegador no tiene soporte para Java, entonces deberá elegir el instalador apropiado manualmente, en la misma página aparece el listado de sistemas operativos soportados, elija el suyo y haga clic sobre el hipervínculo *Download* que se encuentra a la derecha del sistema elegido.

Dependiendo de su navegador, este le puede permitir ejecutar el instalador directamente o guardarlo en una ubicación conocida, si no le permite ejecutarlo, guárdelo y luego diríjase a la ubicación donde lo guardo y haga doble clic sobre el instalador.

Figura 37. Pantalla del instalador web de Ontobiz.



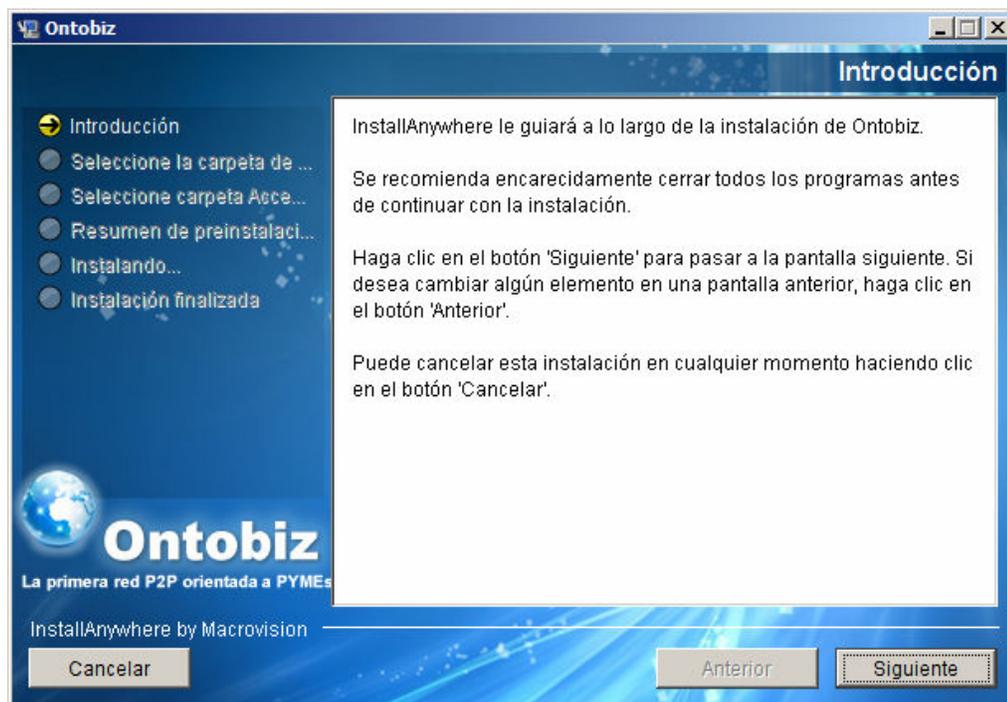
Al ejecutar el instalador, se mostrara en pantalla un asistente que le guiara en el proceso de instalación. Ver Figura 38. La primera pantalla del asistente le permitirá elegir el idioma que el asistente utilizara durante el proceso de instalación.

Figura 38. Pantalla inicial del instalador Ontobiz.



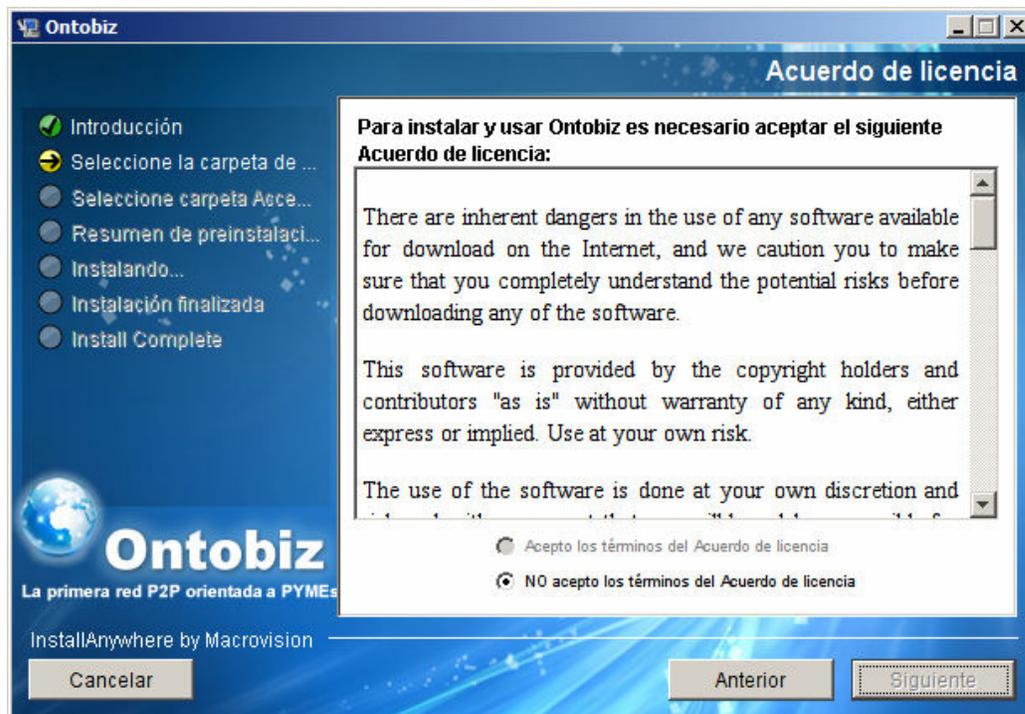
Luego de elegir el idioma del asistente, se mostrará una ventana donde se detallan los pasos del proceso de instalación. Ver Figura 39.

Figura 39. Pantalla de introducción al instalador *Ontobiz*.



Para poder instalar el software deberá aceptar la licencia de uso del mismo, en la figura 40 se puede observar la pantalla del instalador que muestra la licencia. La licencia expresa los términos y condiciones de uso del software aplicables, se recomienda que lea toda la licencia antes de continuar para que esté enterado de antemano las leyes que gobiernan el uso del mismo.

Figura 40. Pantalla de aceptación de la licencia.



Para poder continuar con el proceso de instalación, deberá leer el Acuerdo de Licencia, al llegar al final podrá aceptar la licencia y el botón Siguiente se activara.

A.3 Registro de la Empresa

La primera vez que ejecute la aplicación Ontobiz, deberá llenar un formulario con la información de su empresa, indicando los datos de contacto, así como la clasificación de la empresa según la Actividad Económica a la que pertenece, esta información es necesaria para la compra/venta de productos y servicios. Para su seguridad, la información es encriptada por la aplicación. Ver figura 41.

Figura 41. Pantalla de registro de la empresa.

Registro de Empresa

Datos Generales

Nombre Comercial

Contacto

NIT/NIF Registro Comercial

Actividad Económica 

Tamaño Empresa

Pais

Estado/Región

Provincia

Dirección

Teléfono

Email

Sitio Web

Seguridad

Requerir Autenticación al iniciar aplicación. 

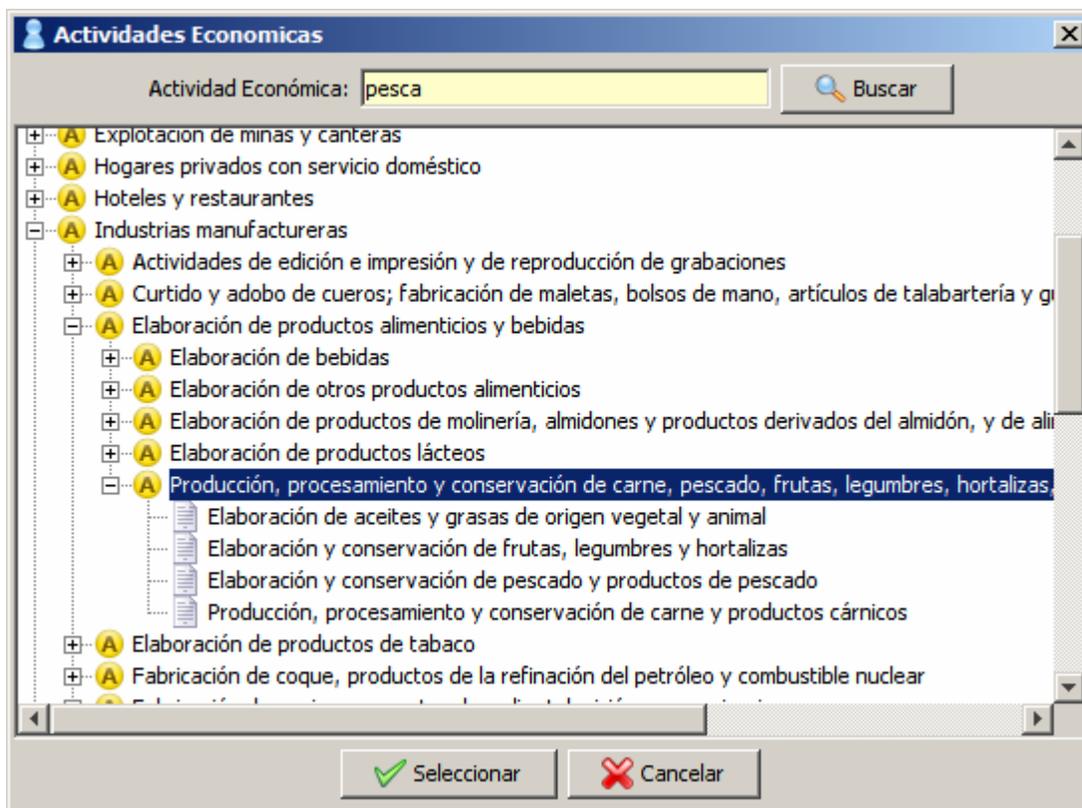
Usuario

Password

A.4 Navegador de Actividades Económicas

La aplicación utiliza la Ontología de actividades económicas que desarrollo David Chang, el navegador permite explorar los conceptos de esta Ontología así como realizar búsquedas de palabras clave para localizar una actividad económica específica. Este navegador es utilizado en las pantallas de Registro de Empresa, administrador de ofertas y en la pantalla de búsqueda. Para ver el navegador de actividades económicas, debe hacer clic en el botón . Ver Figura 42.

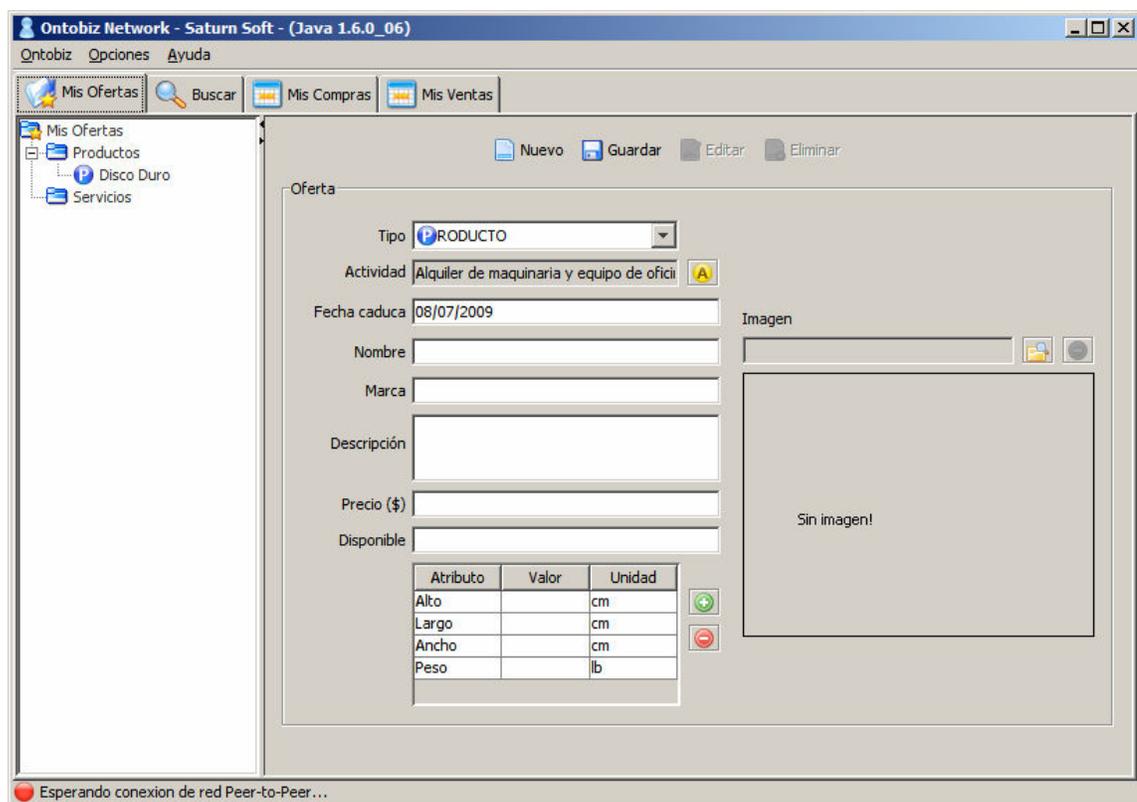
Figura 42. Navegador de actividades económicas.



A.5 Pantalla principal

Una vez ingresada la información de la empresa, se podrá tener acceso a la pantalla principal de la aplicación, la cual se compone de cuatro secciones principales: mis ofertas, buscar, mis compras y mis ventas.

Figura 43. Pantalla principal de la aplicación *Ontobiz*.



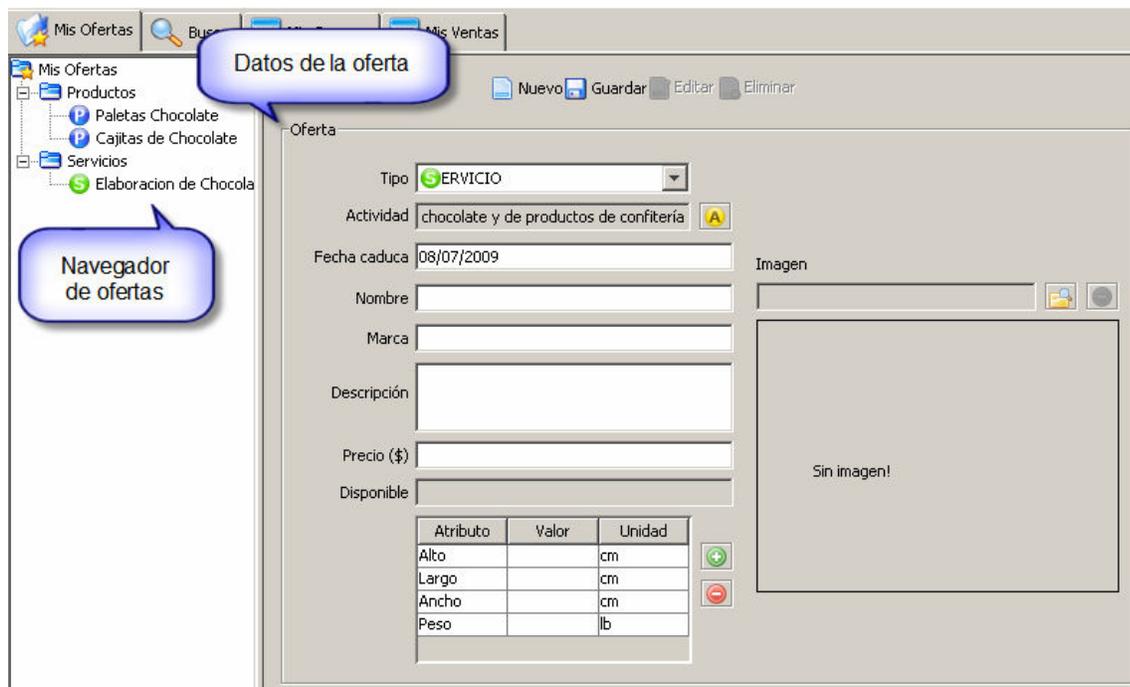
En la parte inferior-izquierda de la aplicación se muestra el status de conexión de la red *P2P*, indicando el status mediante 2 colores:

- (Verde) para indicar que la aplicación está conectada a la red.
- (Rojo) para indicar que la aplicación no está conectada a la red.

A.6 Mis Ofertas

Esta pantalla permite administrar los productos y servicios que la empresa ofertara en la red *P2P*, los cuales estarán disponibles para todos los nodos conectados a la red. Para crear una nueva oferta, primero seleccione el tipo, si es producto o servicio, luego llene la información que se le solicita en pantalla, por default la actividad económica del producto/servicio será el que configuro en el registro de la empresa, sin embargo, lo puede cambiar haciendo clic en el botón . Ver Figura 44.

Figura 44. Pantalla para administración de ofertas.



Datos de la oferta

Navegador de ofertas

Mis Ofertas

Mis Ventas

Nuevo Guardar Editar Eliminar

Oferta

Tipo

Actividad 

Fecha caduca

Nombre

Marca

Descripción

Precio (\$)

Disponible

Imagen

Sin imagen!

Atributo	Valor	Unidad
Alto		cm
Largo		cm
Ancho		cm
Peso		lb

Puede usar el navegador de ofertas para ver y editar las ofertas que ha publicado en la red.

A.7 Buscar ofertas

Esta pantalla es una de las más importantes, ya que nos permite realizar búsquedas de productos y servicios en la red *P2P*. La pantalla presenta algunas opciones que permiten refinar los resultados obtenidos, se pueden realizar búsquedas por los siguientes criterios: tipo de la oferta (producto o servicio), la actividad económica, por el nombre de la oferta, por la marca, por la descripción y por el precio. Ver Figura 45.

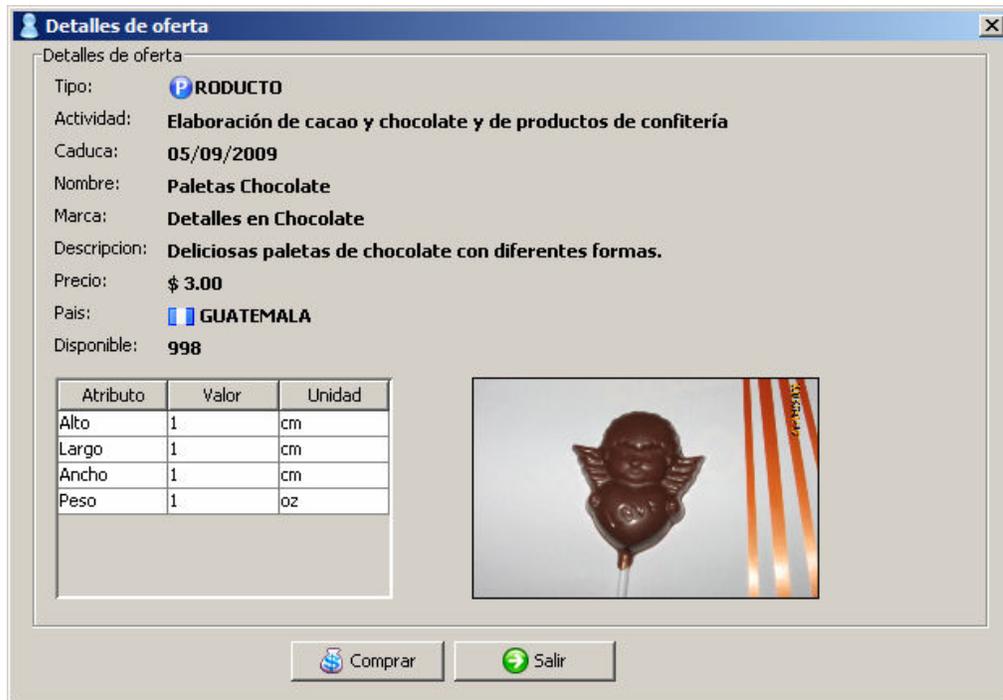
Figura 45. Pantalla para realizar búsquedas.

The screenshot shows a web interface for searching offers. On the left, there is a sidebar with search filters: Tipo (set to PRODUCTO), Actividad, Nombre, Marca, Descripción, and Precio. A 'Buscar' button is located at the bottom of this sidebar. On the right, there is a search results area with a table. Above the table, there are filters for 'busqueda', 'chocolate', and 'chocolate'. The table has columns for Nombre, Marca, Precio, Pais, and Imagen. Two results are shown: 'Paletas Chocolate' with a price of \$ 3.00 and 'Cajitas de Chocolate' with a price of \$ 2.00. A callout bubble points to the search filters with the text 'Opciones de busqueda'. Another callout bubble points to the search results table with the text 'Resultados de las busquedas'. At the bottom of the results area, it says '2 productos'.

Nombre	Marca	Precio	Pais	Imagen
Paletas Chocolate	Detalles en Chocolate	\$ 3.00	II	✓
Cajitas de Chocolate	Detalles en Chocolate	\$ 2.00	II	

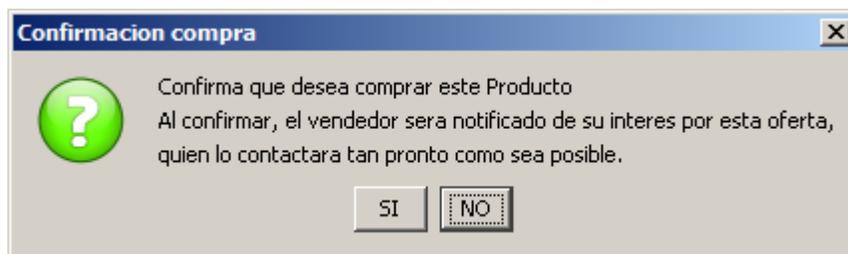
Al realizar una búsqueda, los resultados se mostraran en la sección derecha de la pantalla, podrá ver los detalles de cada oferta encontrada haciendo clic en el botón *Ver detalles*. Ver Figura 46.

Figura 46. Pantalla de detalles de oferta.



Si desea comprar la oferta, puede hacerlo haciendo clic en el botón *Comprar*, la aplicación le mostrará un dialogo solicitándole que ingrese la cantidad a comprar de este producto, finalmente se mostrara un dialogo de confirmación donde puede confirmar o cancelar la compra. Ver Figura 47.

Figura 47. Dialogo de confirmación de compra.



Una vez confirmada la compra, se enviara la solicitud al nodo vendedor y este se pondrá en contacto con usted.

A.8 Mis compras

Esta pantalla permite visualizar el detalle de todas las compras que ha realizado en la red *P2P*. Aquí usted podrá ver el estado de las compras efectuadas recientemente. Ver Figura 48. Cada compra se puede encontrar en alguno de los siguientes estados:

- Pendiente: La compra no ha sido revisada por el nodo vendedor.
- Aceptada: La compra fue aceptada por el vendedor y en cualquier momento este se pondrá en contacto con usted.
- Rechazada: La compra fue rechazada por el vendedor.

Figura 48. Pantalla de visualización de compras.

Fecha	Pais	Producto	Descripción	Cantidad	Total	Estado
08/06/2009	EE	Paletas Chocolate	Deliciosas paletas de chocolate con d...	1	\$ 3.00	Aceptada
08/06/2009	EE	Cajitas de Chocolate	Cajitas con chocolates especiales pa...	1	\$ 2.00	Rechazada
09/06/2009	EE	Elaboracion de Chocolates	Elaboracion de chocolates a la medida.	1	\$ 10.00	Pendiente

Detalle de todas las compras realizadas

3 compras

Enlace de Comunicación creado!

A.9 Mis ventas

Esta pantalla permite visualizar el detalle de todas las ventas que ha realizado en la red *P2P*. Aquí usted podrá ver el estado de las solicitudes de venta recibidas recientemente. Ver Figura 49. Cada venta se puede encontrar en alguno de los siguientes estados:

- Pendiente: Venta pendiente de revisión.
- Aceptada: Venta aceptada y debe ponerse en contacto con el comprador para finalizar la venta.
- Rechazada: Venta rechazada.

Figura 49. Pantalla de visualización de ventas.

Ontobiz Opciones Ayuda

Mis Ofertas Buscar Mis Compras **Mis Ventas**

Ver Venta

Fecha	Pais	Comprador	Producto	Cantidad	Total	Estado
08/06/2009		SATURN SOFT	Paletas Chocolate	1	\$ 3.00	Aceptada
08/06/2009		SATURN SOFT	Cajitas de Chocolate	1	\$ 2.00	Rechazada
09/06/2009		SATURN SOFT	Elaboracion de Chocolates	1	\$ 10.00	Pendiente
09/06/2009		SATURN SOFT	Paletas Chocolate	1	\$ 3.00	Pendiente

Detalle de ventas.

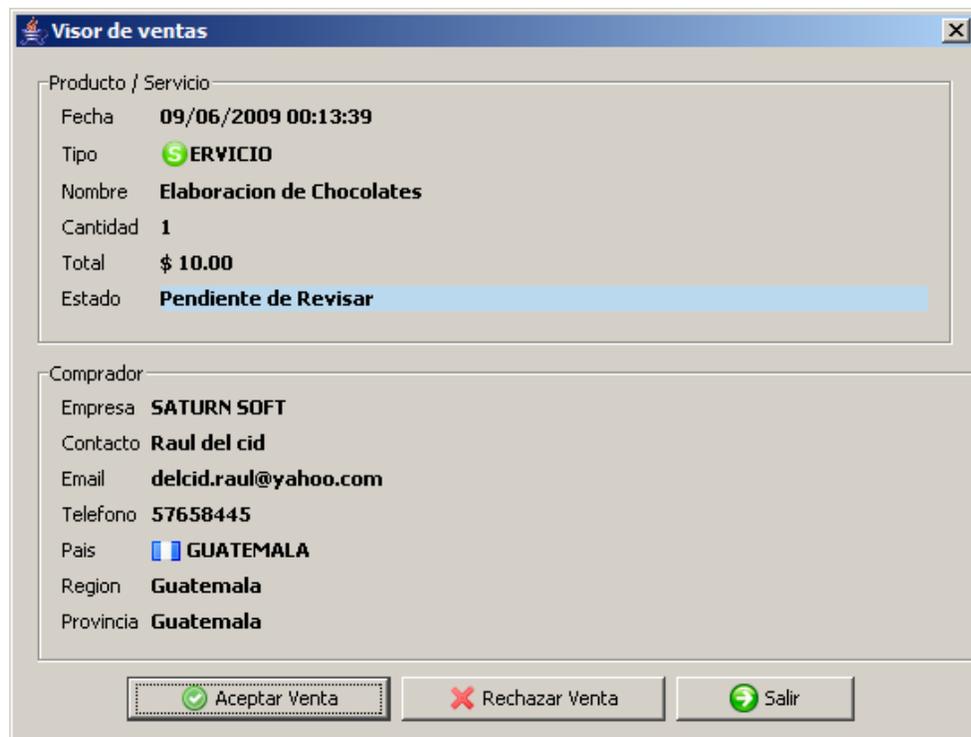
4 ventas

● Enlace de Comunicación creado!

Para ver el detalle de la venta, haga clic en el botón *Ver Venta*.

Se mostrará una ventana donde tendrá la posibilidad de revisar los detalles de la venta, y además podrá modificar su estado, colocando la venta como aceptada o rechazada. Ver Figura 50.

Figura 50. Pantalla de visualización detallada de ventas.



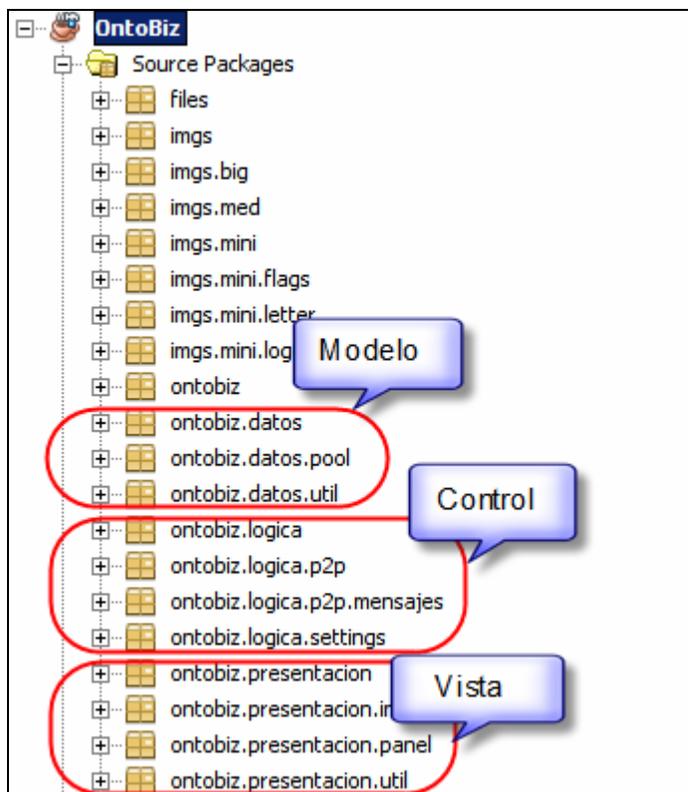
Una vez aceptada o rechazada la venta, se notificara al comprador el estado de la venta, y en este caso es responsabilidad del vendedor ponerse en contacto con el comprador para finalizar la transacción. Con esto se completa el ciclo de funcionamiento de la aplicación.

B. Manual técnico

Para el desarrollo de esta aplicación, se utilizó el patrón de diseño *MVC* (*Modelo Vista Control*), este patrón divide la arquitectura de la aplicación en tres capas: Vista, que representa la parte gráfica de la aplicación, Control que representa la lógica del negocio y el modelo que representa los datos que va a manejar la aplicación. Esta separación de la información con su representación final, ayuda a entender mejor el problema y facilita el mantenimiento del código.

Para lograr este objetivo, el código fuente de la aplicación se dividió en paquetes, cada uno de los cuales corresponde con el modelo *MVC*, esto se puede apreciar mejor en la Figura 51.

Figura 51. Estructura en paquetes del código fuente.



B.1 Paquete datos

El paquete de datos corresponde con la capa modelo del patrón *MVC*, aquí se encuentran todas las clases que representan la información de la aplicación, las cuales se describen a continuación:

Nombre de Clase	Descripción
Actividad	Clase que representa una actividad económica.
Atributo	Clase que representa un atributo asociado a un producto o servicio.
Compra	Clase que representa una compra realizada en la red.
ModeloAtributos	Clase que representa el conjunto de atributos asociados a un producto o servicio.
Oferta	Clase que representa la oferta de un producto o servicio publicado en la red <i>P2P</i> .
OfertaRemota	Clase que hereda de <i>Oferta</i> y agrega la dirección del nodo que publica la oferta, esto nos permite comunicación posterior con el nodo origen.
Pais	Clase que representa un país y permite identificarlo en la aplicación.
<i>Symbol</i>	Clase que sirve para encapsular un <i>String</i> que servirá como llave en un <i>HashMap</i> . Al usar una instancia de <i>Symbol</i> como llave, se mejora el rendimiento en el método " <i>get</i> " hasta en un 30% comparado con usar simplemente un <i>String</i> como llave. Esta clase almacena el <i>hashcode</i> calculado del <i>String</i> para no volver a calcularlo cada vez. Usa patrón: <i>Singleton</i>
Venta	Clase que representa una venta realizada en la red <i>P2P</i> .

B.1.1 Paquete *pool*

Este paquete pertenece a la capa de datos y contiene todas las clases asociadas con la creación de pool de datos que permiten la reutilización de objetos que son caros de crear y que no están siendo utilizados por la aplicación. Esto permite mejorar el rendimiento de la aplicación así como el uso de memoria.

Nombre de clase	Descripción
<i>DocumentPool</i>	Clase que representa un Pool de objetos, permite reutilizar objetos que fueron creados anteriormente y que ya no serán utilizados en el futuro, para sacar un objeto del Pool usar <i>checkOut()</i> , una vez utilizado hay que retornarlo para que pueda ser reutilizado por alguien más usando <i>checkIn()</i>
<i>DocumentBuilderPool</i>	Pool de <i>DocumentBuilders</i> que administra la creación de objetos tipo <i>DocumentBuilder</i> . Subclase de <i>DocumentPool</i> .
<i>StringBuilderPool</i>	Pool de objetos <i>StringBuilder</i> , que mejora el rendimiento de la aplicación al permitir reutilizar instancias que ya no se están utilizando.
<i>StringWriterPool</i>	Pool de objetos <i>StringWriter</i> , que mejora el rendimiento de la aplicación al permitir reutilizar instancias que ya no se están utilizando

B.1.1 Paquete util

Este paquete pertenece a la capa de datos y contiene todas las clases de utilidad asociadas con la manipulación de información.

Nombre de clase	Descripción
<i>TimedListListener</i>	Interfaz usada para notificar cuando un elemento de la lista ha cumplido su tiempo límite.
<i>DateConverter</i>	Clase que posee métodos útiles para conversión de fechas.
<i>FileCacher</i>	Clase que permite almacenar localmente archivos para que siempre estén disponibles sin importar la fuente de donde vinieron
<i>HexConverter</i>	Clase para la conversión de datos de y a hexadecimal.
<i>MessageLogger</i>	Clase útil para escribir log de eventos a un archivo plano, almacenando la fecha y hora del evento.
<i>RatedList</i>	Lista que contiene un conjunto de elementos que pueden ser calificados individualmente.
<i>Secure</i>	Clase que permite encriptación de información utilizando llaves de 128 bits.
<i>StringUtils</i>	Clase con métodos útiles para manipulación de cadenas.
<i>TimedList</i>	<i>ArrayList</i> cuyos elementos poseen un tiempo de Vida.
<i>Zipper</i>	Clase con métodos útiles para compresión y descompresión de datos

B.2 Paquete lógica

El paquete de lógica corresponde con la capa *Control* del patrón *MVC*, aquí se encuentran todas las clases que representan la lógica del negocio de la aplicación, las cuales se describen a continuación:

Nombre de Clase	Descripción
<i>OfferListener</i>	Interfaz para notificar la creación, modificación y eliminación de ofertas a las clases que la implementen y que se registren para dicha notificación.
Actividades	Clase que contiene métodos para cargar la Ontología del disco duro a memoria.
Búsquedas	Clase que administra las búsquedas realizadas en la red <i>P2P</i> .
Compras	Clase que administra las compras realizadas en la red <i>P2P</i> . La lista es almacenada a disco en formato <i>XML</i> .
Ofertas	Lista que permite encapsular un conjunto de objetos tipo <i>Oferta</i> , y además posee métodos para su conversión a formato <i>XML</i> .
<i>OfertasManager</i>	Representa el listado de ofertas de este Nodo, y que son publicadas en la red <i>P2P</i> .
Países	Clase que Carga el Listado de países de disco duro a memoria.
Ventas	Lista administradora de todas las ventas realizadas. La lista es almacenada a disco en formato <i>XML</i> .

B.1.1 Paquete *p2p*

Este paquete pertenece a la capa de lógica y contiene todas las clases relacionadas con la conexión y comunicación de la aplicación con la red P2P y los demás nodos de la red.

Nombre de clase	Descripción
<i>OperationListener</i>	Interfaz para notificar la finalización de Operaciones con el Peer.
<i>PeerDownloadListener</i>	Interfaz para notificar el estado y progreso de la descarga de archivos desde <i>peers</i> remotos.
<i>PeerListener</i>	Interfaz para Notificar Eventos de la red P2P a los objetos que la implementen.
<i>Peer</i>	Representa un peer tipo nodo de la plataforma <i>JXTA</i> , hereda de <i>JxtaPeer</i> , que contiene la base para formar parte de la red <i>OntoBiz</i> . Implementa el patrón de diseño <i>SINGLETON</i> .
<i>PeerStatusEvent</i>	Clase usada por el <i>PeerListener</i> para la notificación del estado de conexión del Peer con la red <i>P2P</i> .
<i>PipeConverter</i>	Clase con métodos especiales para conversión de <i>PipeAdvertisements</i> a formato <i>XML</i> .
<i>Rendezvous</i>	Representa un peer tipo <i>RENDEZVOUS</i> de la plataforma <i>JXTA</i> , hereda de <i>JxtaPeer</i> e implementa la funcionalidad del servidor centralizado de la red <i>P2P</i> .
<i>JxtaException</i>	Clase que hereda de <i>Exception</i> y es usado para lanzar excepción por la clase <i>JxtaPeer</i> .

B.1.2 Paquete mensajes

Este paquete pertenece al paquete *p2p* de la capa de lógica y contiene todas las clases relacionadas con el envío de mensajes entre los nodos dentro de la red *p2p*.

Nombre de clase	Descripción
<i>BuyListener</i>	Interfaz para recibir notificaciones al recibir una solicitud o respuesta de <i>BuyRequest</i> .
<i>JxtaMessage</i>	Interfaz que representa un mensaje XML y que es la base de comunicación de la red <i>p2p</i> .
<i>MessageReplyListener</i>	Interfaz que permite a cualquier clase o método registrarse para escuchar la respuesta (si la hay) de un mensaje enviado hacia un Peer.
<i>BuyReply</i>	Clase que representa el mensaje de respuesta a una solicitud de compra.
<i>BuyRequest</i>	Clase que representa un mensaje de solicitud de compra.
<i>MessageEnvelope</i>	Estructura para almacenar un mensaje y su destino para posterior envío.
<i>MessageReplyEvent</i>	Clase que contiene la descripción del evento que se dispara al recibir un mensaje de respuesta de la red.
<i>MessageReplyManager</i>	Clase que administra el registro de <i>MessageReplyListeners</i> y la invocación de eventos al recibir un Mensaje de respuesta por el <i>PipeMsgListener</i> de la clase Peer.
<i>MessageWrapper</i>	Encapsulador de mensajes que forman el protocolo de comunicación entre nodos <i>Peer-to-</i>

	<i>Peer</i> de la red <i>JXTA</i> .
<i>SaleCompleted</i>	Clase que representa un mensaje de una venta completada.
<i>SearchRequest</i>	Clase que representa un mensaje de búsqueda realizada en la red. Encapsula los criterios de búsqueda ingresados.

B.1.3 Paquete *settings*

Este paquete pertenece a la capa de lógica y contiene todas las clases relacionadas con la administración de las opciones que se pueden configurar en la aplicación.

Nombre de clase	Descripción
<i>CustomSettings</i>	Almacena la Configuración de atributos de personalización de la aplicación.
<i>NetSettings</i>	Almacena la configuración de la red de la aplicación.
<i>PeerSettings</i>	Almacena la configuración de los datos de la empresa de forma segura.
<i>Settings</i>	Clase base para obtener <i>settings</i> del archivo de configuración. Implementa el patrón de diseño <i>Singleton</i> . Los <i>settings</i> son almacenados en formato <i>XML</i> .

B.3 Paquete presentación

El paquete de presentación corresponde con la capa *Vista* del patrón *MVC*, aquí se encuentran todas las clases que representan la interfaz grafica que permiten la interacción con el usuario.

Nombre de Clase	Descripción
<i>diaActividades</i>	Clase que representa la interfaz grafica para Interactuar con las actividades económicas.
<i>diaCustomize</i>	Clase que representa el dialogo de personalización de la aplicación.
<i>diaEmpresa</i>	Clase que representa el diálogo por medio del cual se pueden registrar los datos de una empresa.
<i>diaRed</i>	Clase que representa el dialogo por medio del cual se puede configurar las opciones de red.
<i>frmOntobiz</i>	Clase principal de la interfaz grafica por medio de la cual se puede interactuar con la red <i>P2P Ontobiz</i> .
<i>PaisRenderer</i>	Clase usada para pintar en pantalla los nombres de los países junto con su bandera asociada.