UNIVERSIDAD NACIONAL AUTONOMA DE MEXICO

ESCUELA NACIONAL DE ESTUDIOS PROFESIONALES
"ARAGON"

OPTIMIZACION DEL TIEMPO DE RESPUESTA DE SISTEMAS DE BUSQUEDA EN ACERVOS. EMPLEANDO UN CONJUNTO DE COMPUTADORAS ASOCIADAS EN UN AMBIENTE DE RED TCP/IP.

T E S I S

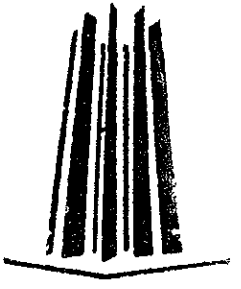
QUE PARA OBTENER EL TITULO DE

INGENIERO EN COMPUTACION

P R E S E N T A :

EDITH VALENCIA MARTINEZ

DIRECTOR DE TESIS: ING. DONACIANO JIMENEZ VAZQUEZ



Dedico esta tesis a la memoria de mi tío
Profesor Huberto Martínez Alarcón.
Gracias por tu gran ejemplo de vida,
y por todo tu amor.
Estoy segura que nos volveremos a
encontrar todos juntos allá algún día.

Un especial agradecimiento a papá Dios por el gran amor que me brindado a lo largo de toda mi vida, por permitirme alcanzar mis sueños y por tomarme de la mano y caminar siempre conmigo. Gracias por el sueño que tuviste en mi cuna.

Doy gracias a mis papis Maria Luisa y Florentino, por su amor, enseñanzas, consejos, compañía y por ser los padres más maravillosos que pude haber soñado

Gracias a mi hermano César Luis, por ser el mejor amigo que puedo tener, por los momentos que hemos compartido juntos, por las peleas, regaños y momentos de alegrías y tristezas.

Gracias a mamá Gallis y a tía Jaquie por quererme tanto, por ser un gran apoyo, y por su gran ejemplo de bondad.

Gracias a todos mis familiares, en especial a mi primo Salomón, por los momentos de tan agradable compañía.

Gracias a todos mis amigos, en especial a Adriana, Juan Luis, José Luis, Armando, Efraín, Ana, Aguilar, Lalo, Israel, Víctor, Sergio, Gonzalo, Rubén, Angélica, Daniel, y un interminable etcétera, que por falta de espacio no es posible mencionar, gracias por darme la oportunidad de aprender a crecer juntos.

Gracias a la Universidad Nacional Autónoma de México y a todo lo que ella representa para mí, Prepa 7, ENEP Aragón, DCAA, DGSCA, CPI con todos los profesores, jefes y compañeros que han hecho posible que sea lo que ahora soy.

Gracias al Ingeniero Donaciano por su ayuda en la elaboración de este trabajo.

Indice

Apéndices

Capítulo 1

Introducción

1.1 Antecedentes

En el presente siglo, el acceso rápido y eficiente a diversas fuentes de información, juega un papel muy importante en todos los ambientes de trabajo, tanto científico, como académico, empresarial e industrial.

Con el nacimiento de las redes de comunicación, y en especial de Internet, la información disponible al usuario aumenta día con día, sin embargo, dicho crecimiento en la información ocasiona, entre otras cosas:

- Problemas al usuario, al no poder encontrar la información de su interés en poco tiempo.
- Demanda de más sofisticados procesamientos de información, y en consecuencia de recursos de cómputo más costosos.

Las razones anteriores provocan que cada vez sea más necesaria la búsqueda e implantación de soluciones económicamente viables, que ayuden al usuario a encontrar rápidamente la información que requiere.

El presente trabajo surge como solución a ésta problemática, considerando como principal objetivo la generación de un sistema rápido y confiable, que tenga la capacidad suficiente para soportar una gran cantidad de usuarios y de información.

En especial, el trabajo se enfoca al sitio WWW de la UNAM, a través del cual, nuestra máxima casa de estudios cuenta desde el año de 1991 con una presencia importante en Internet, en donde podemos encontrar la información más reciente e importante de nuestra Universidad.

La presente tesis está organizada en 7 capítulos los cuales se describen a continuación.

Capítulo 1 Introducción. En este capítulo se define el objetivo de la realización del proyecto.

Capítulo 2 Planteamiento de la problemática a resolver. Está dedicado a la exposición de los problemas que nos llevaron al desarrollo del proyecto.

Capítulo 3 Análisis y diseño de la propuesta. En esta parte se detallará en que consiste el proyecto, se evaluarán las alternativas para ponerlo en marcha y se eligirá la más viable.

Capítulo 4 Configuración e implementación de componentes. Este apartado incluye el desarrollo del proyecto como tal, presentando la interacción entre las partes necesarias para su funcionamiento.

Capítulo 5 Ejecución de pruebas y refinamiento del servicio. En este capítulo se expondrán las pruebas realizadas para la verificación del funcionamiento del sistema.

Capítulo 6 Perspectivas de desarrollo. Se dedica a resaltar la importancia y utilidad que representa el proyecto como una solución eficiente y económica en lugares donde se requieren sistemas de búsqueda con manejo masivo de información.

Capítulo 7 Conclusiones. Por último, se presentan las conclusiones al trabajo que se presenta como tema de tesis.

1.2 Ambientes de Red

Durante las primeras 2 décadas de su existencia, los sistemas de computadoras estuvieron muy centralizados, usualmente en el interior de un cuarto muy grande. Una compañía mediana o una universidad, podían contar con una o dos computadoras, en tanto que las instituciones más grandes tenían a lo sumo una docena de ellos.

La fusión de las computadoras y de las comunicaciones ha tenido una profunda influencia en la forma en que estos sistemas están organizados El concepto de centro de cómputo como un cuarto con una gran computadora, a la cual los usuarios llevaban su trabajo para procesarlo, ha llegado a ser obsoleto. Este modelo no tiene uno, sino al menos 2 aspectos deficientes: primero, el concepto de una sola computadora grande haciendo todo el trabajo, y segundo, la idea de que los usuarios lleven su trabajo a la computadora, en vez de llevar la computadora a donde se encuentran los usuarios.

El viejo modelo de tener solo una computadora para satisfacer todas las necesidades de cálculo de una organización ha sido reemplazando por otro que considera un gran número de computadoras separadas, pero interconectadas, que se dividen el trabajo Estos sistemas se conocen como redes de computadoras

Una red de computadoras es una colección de computadoras autónomas interconectadas; se dice que dos computadoras están interconectadas, si son capaces de intercambiar información. La conexión no necesita hacerse por medio de un cable, también puede hacerse por medio del uso de microondas o satélites de comunicaciones.

1.2.1 Objetivos de las redes

Son muchas las organizaciones que cuentan con un número considerable de computadoras en operación y con frecuencia alejadas unas de otras. Inicialmente cada una de estas computadoras puede haber estado trabajando en forma aislada de las demás pero, en algún momento, la administración puede decidir interconectarlas para tener así la capacidad de extraer e intercambiar la información referente a toda la compañía.

Puesto en una forma más general, el tema aquí consiste en compartir recursos, y el objetivo es hacer que todos los programas, datos y equipo estén disponibles para cualquiera de la red que así lo solicite, sin importar la localización física del recurso y del usuario. Otro aspecto de compartir recursos es el relacionado con la división de la carga de trabajo.

Un segundo objetivo consiste en ofrecer una alta fiabilidad, al contar con fuentes alternativas de suministro, otro objetivo es el ahorro económico, las computadoras pequeñas tienen una mejor relación costo/beneficio, comparada con la ofrecida por las máquinas grandes.

Otro objetivo del establecimiento de una red de computadoras no tiene nada que ver con la tecnología. Una red de computadoras proporciona un poderoso medio de comunicación entre personas que se encuentran muy alejadas entre sí.

1.2.2 Estructura de red

En toda red existe una colección de computadoras destinadas a correr aplicaciones de los usuarios; dichas máquinas son llamadas anfitriones, sistema final o sistema terminal. Los anfitriones están conectados mediante una subred de comunicación. El trabajo de la subred consiste en enviar mensajes entre anfitriones. El diseño completo de la red se simplifica notablemente cuando se separan los aspectos puros de comunicación de la red (subred), de los aspectos de aplicación (anfitriones).

Una subred en la mayor parte de las redes de área amplia consiste de 2 componentes diferentes las líneas de transmisión y los elementos de conmutación Las líneas de transmisión son las encargadas de mover bits entre máquinas, los elementos de conmutación son dispositivos especializados que se utilizan para conectar 2 o más líneas de transmisión En la figura 1.1 se ilustra lo anterior

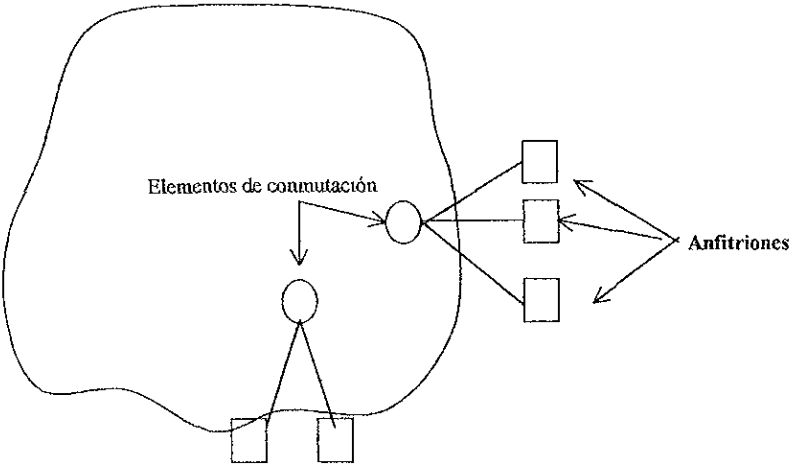


Figura 1.1 Relación entre los anfitriones y la red

En términos generales, puede decirse que hay dos tipos de diseño para la subred de comunicación:

1 Canales punto a punto

2. Canales de difusión

En el primero de ellos, la red contiene varios cables, conectando cada una de ellas un par de elemento de conmutación. Si dos elementos de conmutación desean comunicarse y no comparten un cable común, deberán hacerlo indirectamente a través de otros elementos de conmutación. Cuando un mensaje se envía de un elemento de conmutación a otro o más elementos de conmutación intermediarios, se almacenará ahí y no continuará su camino hasta que la línea de salida necesaria para reenviarlo esté libre. La subred que utiliza este principio, se denomina red punto a punto, de almacenamiento y envío o de conmutación de paquetes.

Un aspecto importante de diseño, cuando se utiliza una subred punto a punto consiste en considerar como deberá ser la topología de interconexión En la figura 1.2 se muestran varias topologías posibles. Las redes locales que se diseñaron como tales, tienen por lo general una topología simétrica A diferencia de éstas las redes de área amplia tienen típicamente topologías irregulares.

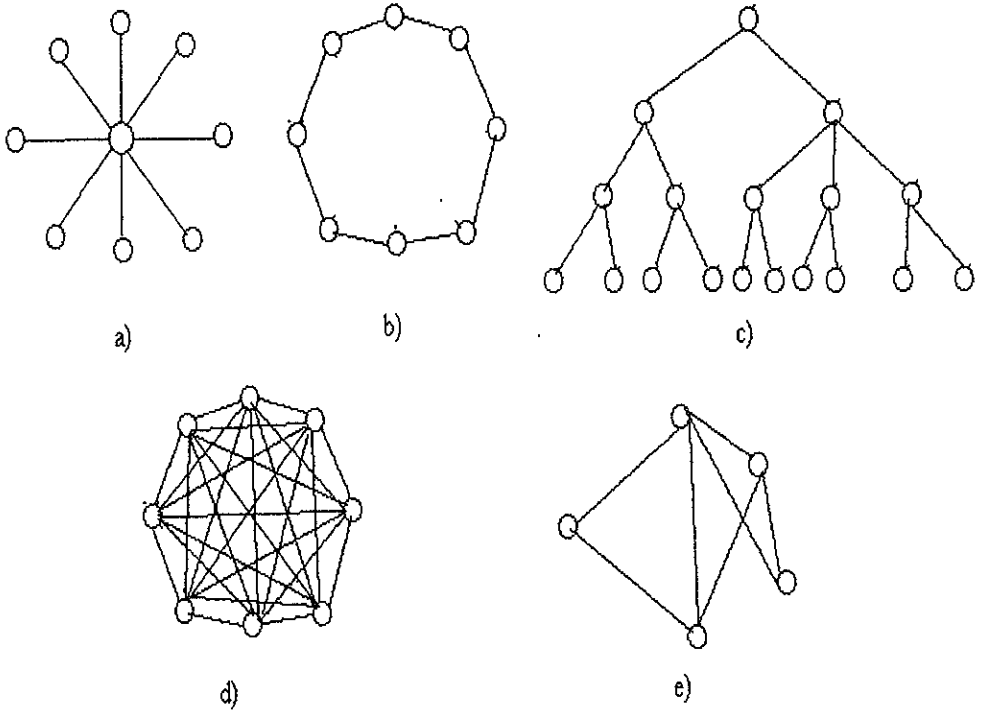


Figura 1.2 Algunas topologías para una subred punto a punto A)Estrella, b)Anillo, c)Arbol, d)Completa, e)Irregular.

La difusión se emplea como un segundo tipo de arquitectura de comunicación, la cual consta de un solo canal de comunicación que es a su vez compartido por todas las máquinas que constituyen la red. Los paquetes que una máquina envía, son recibidos por todas las demás. El campo de dirección, localizado en el interior de un paquete, especifica a quien va dirigido. En el momento en que se recibe un paquete, se verifica el campo de dirección, y si el paquete está dirigido a otra máquina, este simplemente se ignora. En la figura 1.3 se muestran algunas posibilidades de subredes de difusión.

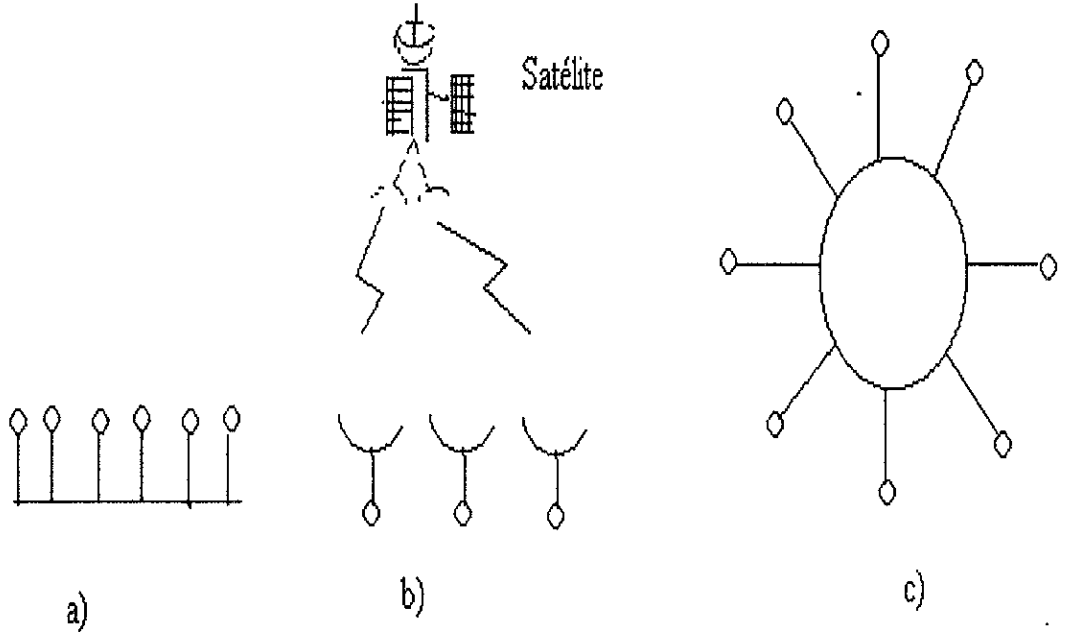


Figura 1.3 Comunicación de subredes de difusión. A)Bus b)Satélite o radio c)Anillo

Capítulo 2

Planteamiento de la problemática a resolver

En el presente capítulo se expone en primer lugar, una breve historia de las comunicaciones y de los servicios de Internet en la UNAM, posteriormente se presenta de una forma general la estructura en la que se encuentran organizados actualmente los sitios WWW que conforman el dominio de la UNAM, y finalmente se hace un análisis y planteamiento de la problemática a la que se enfrentan los usuarios que hacen uso de la información almacenada en estos sitios.

2.1. Las comunicaciones y los servicios de Internet en la UNAM

La etapa de inicio de las comunicaciones telefónicas y de datos en la Universidad Nacional Autónoma de México la encontramos a principios de los años 70, período en el cual se establecieron una gran cantidad de conexiones entre diversos equipos como terminales y estaciones de trabajo hacia un ordenador central, a partir de esa fecha se presentaron diversos avances en materia de comunicaciones y así llegamos al año de 1989 cuando la UNAM establece un convenio de enlace a la red de la NFS en EUA por medio del satélite mexicano Morelos II, es este momento el que marca el nacimiento de las redes de computadoras en la UNAM gracias a las conexiones que se establecieron para la intercomunicación de computadoras a través de fibra óptica, enlaces satelitales y enlaces de microondas.

Es en el año de 1990 cuando la UNAM se incorpora a la red mundial Internet, y comienza a proporcionar diferentes servicios tales como comunicación entre usuarios de cualquier parte del mundo a través de correo electrónico, búsquedas de información a través de menús jerárquicos por medio de gopher, búsqueda electrónica utilizando archie, sesiones remotas por medio de telnet, así como información de diversa índole que podemos encontrar en los diversos servidores de WWW que existen en la UNAM.

Finalmente llegamos a nuestros días en los que la UNAM cuenta con una red integral de telecomunicaciones que enlaza las principales facultades, institutos, centros de difusión, coordinaciones y demás dependencias que conforman a la UNAM

2.2. La importancia de la tecnología WWW en Internet

A partir de la fecha en la que la tecnología de WWW se incorpora a Internet, y consecuentemente al caso que nos ocupa, la red de comunicaciones de la UNAM, la cantidad de información y el número de sitios que la conforman empiezan a experimentar un gran crecimiento, todo esto debido a las características propias de ésta tecnología como son:

a) Capacidad de poder utilizar cualquier texto o imagen como liga que le permite al usuario saltar directamente a la información relacionada, gracias al formato de datos HTML que utiliza el WWW. Este método de recuperación de información es una gran ventaja sobre el tedioso sistema de menús jerárquicos usados por gopher, la navegación de hipertexto contribuye a la facilidad de uso del WWW.

b) Capacidad de integrar funciones de otros servicios de Internet como son· correo electrónico, gopher, ftp, noticias, etc dentro de uña interfaz única, por lo que para los usuarios existe un solo programa cliente que aprender y usar, en vez de un programa separado para cada servicio de Internet

c) Capacidad de incluir técnicas de multimedia como son animaciones, audio y video en los documentos, lo que hace que este medio sea más atractivo para los usuarios.

El primer sitio de WWW con que cuenta la UNAM se desarrolló en el año de 1991, posteriormente las diferentes dependencias que integran a la UNAM inician la construcción de sus propios sitios a fin de ofrecer información de diversa índole a la comunidad universitaria

2.3. Organización de los sitios WWW de la UNAM

Gracias a la red integral de comunicaciones con que cuenta la UNAM se logra tener una telaraña de sitios que actualmente alcanza un total de 257 con 1,568 700 páginas[1], teniendo como punto de entrada a todos ellos, el sitio principal de la UNAM que podemos accesar a través de la dirección http //www unam mx.

Este sitio alberga varios de los aspectos importantes de la máxima casa de estudios de México, tales como su Identidad, organización, la comunidad universitaria que la integra, y su historia e infraestructura.

También podemos encontrar información referente a los servicios que ofrece como son:

- Planes de Estudio
- Educación a distancia
- Servicios de Computo
- Enseñanza a extranjeros
- Deportes y recreación
- Servicios de Telecomunicaciones
- Servicios institucionales
- Servicios de Información

1. Ver apéndice 1 para mayor información

Algunos de los servicios de Internet que proporciona RedUNAM tales como:

- Radio UNAM por medio de Internet.
- Sabueso, que nos permite localizar personas.
- Servicio de FTP anónimo
- Directorio telefónico de la UNAM.
- Servicio de traducción automática
- Incorporación de paginas WWW para dependencias de la UNAM y para instituciones externas
- Búsquedas de sitios de Web en México en catalogo organizado por temas

También cuenta con acervos de información entre los cuales tenemos:

- Noticias de la UNAM, como es la gaceta, y la síntesis informativa
- Noticias de México, con periódicos como La Jornada, El Nacional, El Universal, y El Economista, por mencionar algunos.

El número de sitios con que cuenta la UNAM y la información relevante y variada que se encuentra almacenada en cada uno de ellos ha hecho posible que el conjunto de servidores WWW de la UNAM se convierta en uno de los centros de almacenamiento de información más importantes de México, lo que se puede comprobar con solo verificar las estadísticas de acceso a algunas de las páginas más importantes, las cuales se muestran en la tabla 2.1.

Tabla 2.1 Estadísticas de Acceso del período 31 de Enero al 28 de Febrero de 1998.[2]

Nombre y url asociado	Archivos transmitidos	Bytes transmitidos	Promedio de archivos transmitidos	Promedio de bytes transmitidos
Página principal de la UNAM. http //www.unam mx	361356	3189289273	32851	289935388
Gaceta UNAM http //www.unam.mx/gaceta	16775	201569737	699	8398739
Síntesis Informativa http://www unam.mx/sintesis	10888	48299760	454	2012490
Servicios Hemerográficos http://www.unam.mx/serv_hem	143049	2008040405	5960	83668350
Periódico el Nacional http //www.unam mx/nacional	17184	60501579	716	2520899
Periódico La Jornada Http://www unam.mx/jornada	3459976	22122376903	138399	884895076
Periódico el Universal http //www.unam mx/universal	143758	533391673	5990	22224653

2. Fuente de Información: Estadísticas WWW de la UNAM http://www.unam mx/estadisticas

2.4. Problemática de la información en lo Servidores WWW de la UNAM

Los 257 sitios WWW que conforman a la UNAM no pueden ser administrados centralmente, debido a que cada uno de ellos contiene información relacionada con la dependencia a la que pertenece y debe existir una persona encargada de mantener el sitio, quien es el único autorizado a modificar, añadir o eliminar información para que se pueda conservar una estructura coherente en el interior de cada uno de ellos; sin embargo, debido a ésta administración descentralizada de los Sitios WWW, no se cuenta con una estructura organizada de la información que se tiene en todos los sitios WWW de la UNAM, lo que representa una problemática a los usuarios, para los cuales es difícil encontrar la información que les interesa por medio del método convencional de búsqueda, que consiste en iniciar en una página, y navegar de documento en documento a través de los hiperenlaces hasta llegar al recurso deseado, situación que se agrava debido a la gran cantidad de documentos almacenados en los sitios.

Por estas razones, se requiere de la implementación de una herramienta especializada que ayude al usuario a llevar a cabo la tarea de búsqueda de información de una manera rápida y eficiente

Dicha herramienta debe cumplir al menos con las siguientes características.

a) Contener un acervo general en donde se tenga almacenada centralmente la información de los 257 sitios WWW de la UNAM

b) Recuperación rápida de los documentos que se requieren con base en una cadena de búsqueda.

c) Alto rendimiento para mantener un nivel de respuesta aceptable, aún cuando se tenga una gran cantidad de usuarios.

Capítulo 3

Análisis y diseño de la propuesta

3.1 Características del sistema de búsquedas propuesto

Existen varios factores que influyen directamente sobre la eficiencia de los sistemas de búsquedas, algunos de ellos son:

a) El tamaño, contenido y actualización del índice de recursos.

b) La velocidad de la búsqueda.

c) Las opciones de búsqueda.

d) El diseño de la interfaz al usuario.

El sistema que se planea obtener durante el transcurso de la elaboración del presente trabajo, que surge como respuesta a la problemática que representan a los usuarios las búsquedas de información en los sitios WWW de la UNAM, no pretende resolver todos los problemas anteriores, no es el desarrollo de un nuevo algoritmo para mejorar la herramienta de recopilación de información, o la creación de un nuevo motor de búsqueda con nuevas opciones de lenguaje para lograr mayor relevancia y exactitud en los resultados obtenidos.

Lo que se propone en el presente trabajo, es la construcción de un sistema de búsquedas con el que se tratará de solucionar la problemática de los largos tiempos de respuesta al momento de realizar las consultas, por medio del uso de alguna de las herramientas de recopilación de información y de motores de búsqueda que existen actualmente, después de haber realizado el análisis para la elección de la herramienta más adecuada.

3.2 Sistemas de búsquedas

Las herramientas que se han desarrollado para lograr que la información publicada en Web se pueda encontrar de una manera eficiente son llamadas sistemas de búsqueda, y se pueden definir como un conjunto de programas que realizan las siguientes funciones:

a) Recorren la telaraña de sitios WWW recuperando recursos que usualmente son páginas HTML.

b) Almacenan la información recopilada en una base de datos previamente definida.

c) Ejecutan búsquedas en la base de datos con base en palabras clave o frases que el usuario introduce.

d) Presentan al usuario los documentos que cumplen con la cadena de búsqueda.

Los sistemas de búsquedas generalmente constan de 2 componentes:

- Herramienta de recopilación de información
- Motor de búsqueda

A continuación se describen a detalle las funciones que realizan los 2 elementos anteriores dentro de los Sistema de búsquedas.

Herramienta de recopilación de información

Una herramienta de recopilación de información es el programa que recorre la estructura de hipertexto de los sistemas basados en html en varios sitios de Internet automáticamente, cada vez que obtienen un documento, se dirigen a los enlaces que éste contiene y obtienen los documentos correspondientes en forma recursiva

Es importante hacer notar que la palabra "recursiva" no esta limitada a la definición de un algoritmo especifico, aún si una herramienta de recopilación de información aplica algún heurístico para la selección y orden de los documentos a visitar y las peticiones de documentos están espaciadas por períodos largos de tiempo, sigue siendo una herramienta de recopilación de información.

Las herramientas de recopilación de información pueden ser utilizadas para diferentes propósitos.

a) Obtener información estadística.

Gracias a que las herramientas de recopilación de información generan índices de los recursos que encuentran en los sitios WWW visitados, éstas herramientas pueden ser utilizadas para la generación de estadísticas

Algunas de éstas estadísticas incluyen:

1. Número total de recursos por servidor o por dominio.
2. Páginas más frecuentemente visitadas.
3. Tamaño promedio de una página de Web.
4. Número total de imágenes, páginas html y programas ejecutables almacenados en determinado sitio WWW.

b) Establecimiento de sitios espejo

Para la creación de los sitios espejo, las herramientas de recopilación de información inician en un directorio raíz de la máquina remota y a partir de éste copian el árbol de directorio completo al sistema de archivos en la máquina local.

Para mantener el sitio espejo actualizado, las herramientas de recopilación de información visitan el sitio remoto en periodos de tiempo predeterminados, y copian los documentos que han sido modificados

c) Mantenimiento.

Una de las principales dificultades en mantener una estructura de hipertexto es que las referencias a otras páginas pueden convertirse en "ligas muertas" cuando la página referenciada ha sido cambiada de lugar o eliminada, una herramienta de recopilación de información que verifica referencias puede ayudar al administrador a localizar estas ligas muertas.

Las herramientas de recopilación de información pueden ayudar a mantener el contenido y la estructura de un sitio de Web por medio de verificaciones en el contenido de páginas HTML, actualizaciones regulares, etc.
Este es el contraste al mantenimiento manual de documentos, donde la verificación es frecuentemente esporádica y no comprensible.

d) Generación de índices

Tal vez una de las aplicaciones más importantes de las herramientas de recopilación de información es su uso en el descubrimiento de recursos.

Varias herramientas de recopilación de información, recorren grandes partes de la telaraña de información y almacenan los recursos encontrados en un índice que posteriormente puede ser utilizado para localizar información específica, aún si el índice no contiene la palabra exacta que se desea recuperar, es seguro que contendrá referencias a páginas relacionadas, las cuales en turno pueden referenciar al tema deseado.

Las herramientas de recopilación de información forman una parte importante en los sistemas de búsqueda, al ser utilizados para la generación de índices de recursos

Para la elección de la información que se va a almacenar en los índices, las herramientas de recopilación de información se basan en diversos factores, algunos de los cuales son:

1. Palabras que aparecen al principio del documento.
2. Palabras que aparecen repetidas veces.
3. Todas las palabras del documento
4. Palabras que aparecen en ciertas partes del documento, como son, el encabezado, el cuerpo, el título, los hiperenlaces, etc.

Una vez que se cuenta con las palabras elegidas, éstas son almacenadas en el índice de recursos, incluyendo apuntadores a las direcciones reales en donde residen los documentos de donde provienen éstas palabras.

Motor de búsqueda

Los motores de búsqueda son los programas que ejecutan búsquedas sobre los índices generados por las herramientas de recopilación de información, basados en las consultas que el usuario especifica.

En un inicio los motores de búsqueda realizaban consultas elementales, como localizar en todo el índice la palabra exacta que el usuario introducía. Sin embargo los resultados que éstos programas arrojaban no eran lo que el usuario esperaba, varios de éstos problemas están directamente relacionados con el lenguaje en el que se realiza la consulta, algunos de los cuales son:

1 Homónimos
2. Plurales
3. Distinción de mayúsculas y minúsculas
4. Tiempos de verbos
5. Palabras relacionadas pero no mencionadas

Debido a los problemas anteriores, los desarrolladores de motores de búsqueda han ido incorporando en sus productos diversas opciones para que los resultados obtenidos sean más exactos, algunas de las cuales incluyen:

1. Búsqueda por texto completo
2. Operadores booleanos AND, OR, NOT para
3. Búsquedas por frases
4. Uso de paréntesis para agrupar expresiones de búsqueda
5. Búsquedas por campos
6. Proximidad (una palabra junto a otra)
7 Distinción de mayúsculas y minúsculas
8. Caracteres en ISO-latín como acentos y diacríticos

3.2.1 Servicios de búsquedas en WWW

En la actualidad existen varios programas que se han implementado para ayudar a los usuarios a encontrar información específica a través de Internet de una manera más fácil, las características principales de algunos de ellos se detallan a continuación.

3.2.1.1 Altavista

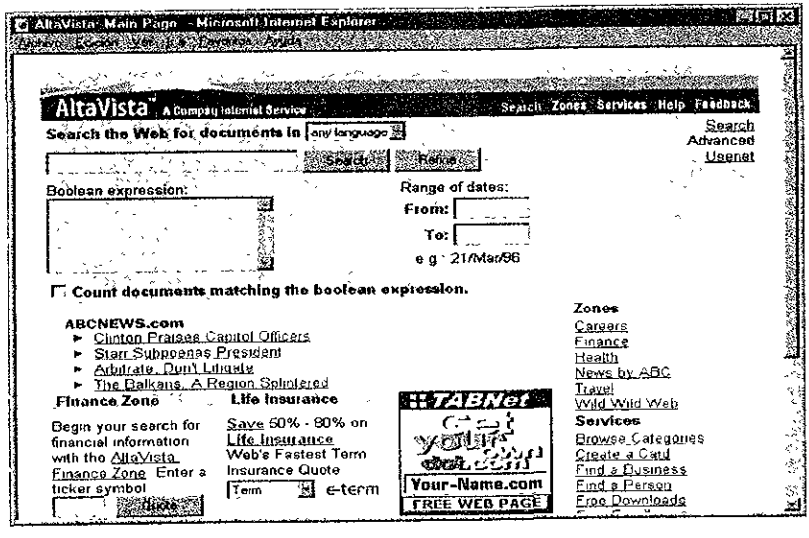


Figura 3 1 Pantalla de búsquedas avanzadas de Alta Vista

El servicio se puede accesar a través de la dirección. http //altavista digital com/

Este servicio es extremadamente rápido, actualizado y general. Presenta búsquedas en texto completo de páginas de Web y artículos de listas de discusión.

Alta Vista es apropiado para buscadores iniciales, y también presenta varias características de búsquedas avanzadas Con una simple búsqueda, los usuarios pueden buscar por frases exactas, palabras requeridas o rechazadas, búsquedas en el campo de título del documento HTML, búsquedas por documentos que contienen una liga a un URL particular, hacer uso de palabras clave, emplear palabras mayúsculas y minúsculas y hacer uso de paréntesis para agrupar expresiones de búsqueda.

La propiedad de búsquedas avanzadas permite el uso de operadores booleanos (AND, OR, NOT, NEAR) y le permite al usuario limitar búsquedas por fecha.

Los resultados que presenta se pueden ver en forma estándar, compacta y detallada, los criterios para la presentación de los resultados se pueden especificara través de la página para búsquedas avanzadas.

Alta Vista fue creado para demostrar el hardware y el software de Digital, aunque actualmente el software se está portando a otras plataformas, la primera de ellas en la que el software puede ser instalado son las máquinas de Sun Microsystems bajo el sistema operativo Solaris. El creador del Software es Digital Equipment Corporation

3.2.1.2 Infoseek

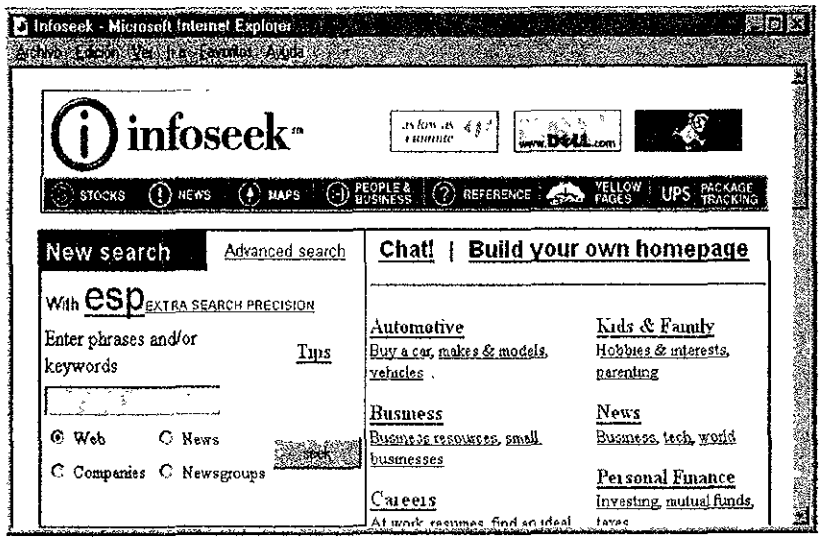


Figura 3.2 Página de entrada al Sistema de Búsquedas de Infoseek

La página del servicio es http //guide infoseek com/

Este servicio de búsquedas rápido y de calidad profesional recupera arriba de 100 resultados.

Los usuarios pueden buscar en páginas de Web, artículos de listas de discusión y en listas de preguntas frecuentes por medio de palabras clave o por medio de frases.

Las características de búsqueda del servicio son búsquedas en el texto completo, contiene una lista de palabras comunes que se excluyen, distinción de mayúsculas y minúsculas, reconoce símbolos y números, búsquedas por frase, proximidad, uso de los símbolos + y – para incluir y excluir términos.

Los resultados de la búsqueda son presentados en orden de mayor a menor importancia. algunos resultados incluyen una lista de temas relacionados y grupos de noticias

La Guía de Infoseek también incluye un directorio selectivo de sitios de Internet en el cual se pueden realizar búsquedas y también se puede navegar Este directorio incluye referencias cruzadas, un diccionario en el cual se pueden realizar búsquedas, y se encuentra integrado con la base de datos de sitios de Web no revisados, es posible navegar a través del directorio jerárquico, encontrar un sitio de Web relevante, y después seleccionas la liga de páginas similares, lo cual mostrará páginas similares en una base de datos más grande

Un servicio separado gratuito, llamado Infoseek profesional provee arriba de 200 resultados en búsquedas de Web y permite accesar otras bases de datos, incluyendo perfiles de compañías, libros, revistas y videos.

Creadores: Infoseek Corporation

3.2.1.3 Yahoo

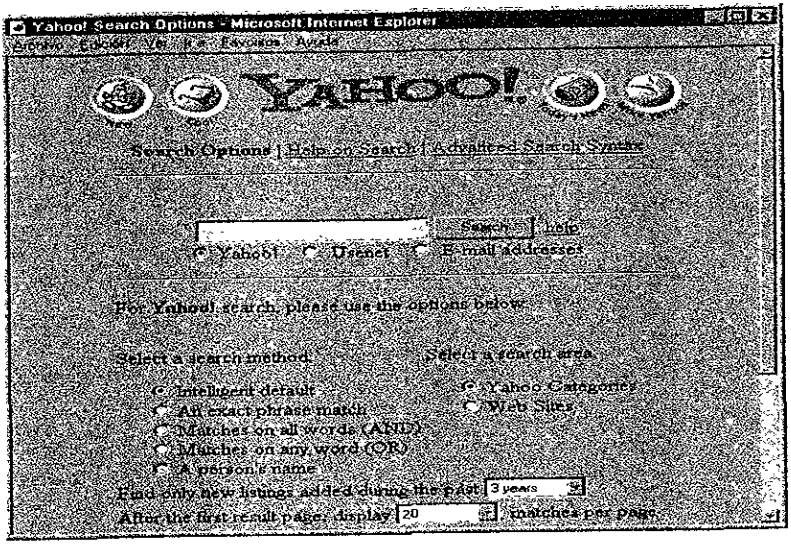


Figura 3 3 Servicio de búsquedas Yahoo

Página de inicio del servicio http //search yahoo com/bin/search/options

Yahoo es el catálogo de Internet más utilizado, en contraste a los sistemas de búsqueda mencionados anteriormente, que utilizan los programas de computadoras para recuperar los documentos de Web automáticamente, Yahoo cataloga los sitios de Web manualmente, dependiendo de los envíos que hacen los usuarios.

La página principal es una lista alfabética de 14 áreas divididas por temas Los usuarios pueden seguir la estructura jerárquica hasta que encuentren el tema específico que requieren Los temas listados en el catálogo son en su mayoría páginas de Web, así como algunos artículos de listas de discusión. Algunas entradas incluyen una frase descriptiva, pero la mayoría incluyen solamente la liga

Los usuarios pueden navegar la estructura jerárquica, utilizar un sistema de búsquedas que realiza las búsquedas en los URLs, títulos y comentarios dentro de Yahoo, buscar los documentos que contienen todos, o al menos uno de los términos de búsqueda, seleccionar subcadenas o palabras completas, seleccionar el número de resultados que se van a desplegar por página, no hace distinción de mayúsculas y minúsculas.

Los resultados que el servicio presenta son una lista de categorías y sitios que contienen los términos que el usuario introduce.

Creadores: Yahoo Corporation.

3.2.1.4 Excite

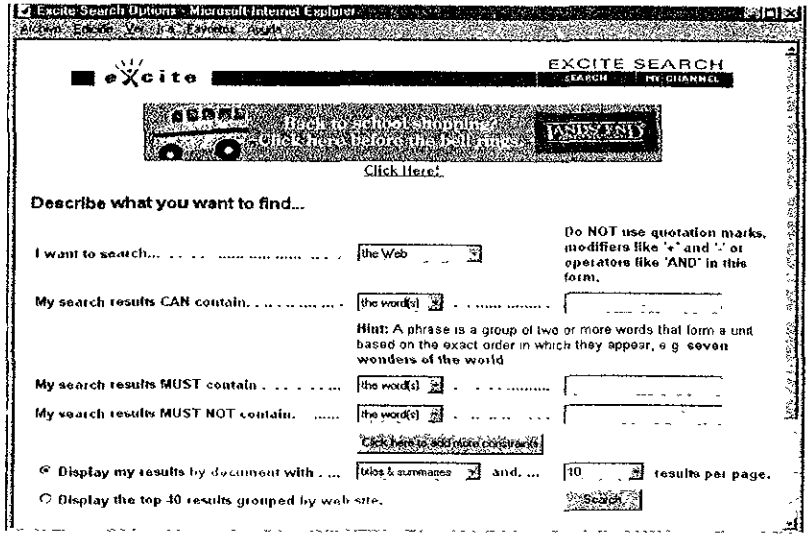


Figura 3 4 Sistema de búsquedas Excite

Página de inicio del servicio http //www excite com/

Excite ofrece una interfaz al usuario elegante y configurable, para buscar en páginas de Web y en las 2 semanas anteriores de artículos enviados a listas de discusión

Las características de búsqueda son texto completo de páginas WWW, lista de palabras comunes que se excluyen, uso de operadores booleanos AND, OR, NOT, paréntesis para agrupar porciones de cadenas de búsquedas booleanas, si al usuario le gusta un resultado de la búsqueda, la opción "búsquedas por ejemplo" le permite recuperar más documentos parecidos o relacionados con éste Características adicionales incluyen un catalogo de sitios revisados (las ligas incluyen entre 1 y 4 líneas que describen el sitio) y noticias actuales.

Es posible utilizar el sistema de búsquedas de Excite para sitios particulares, Netscape utiliza que esta disponible de forma gratuita en el sitio

Creadores: Architext Software

3.2.1.5 Lycos

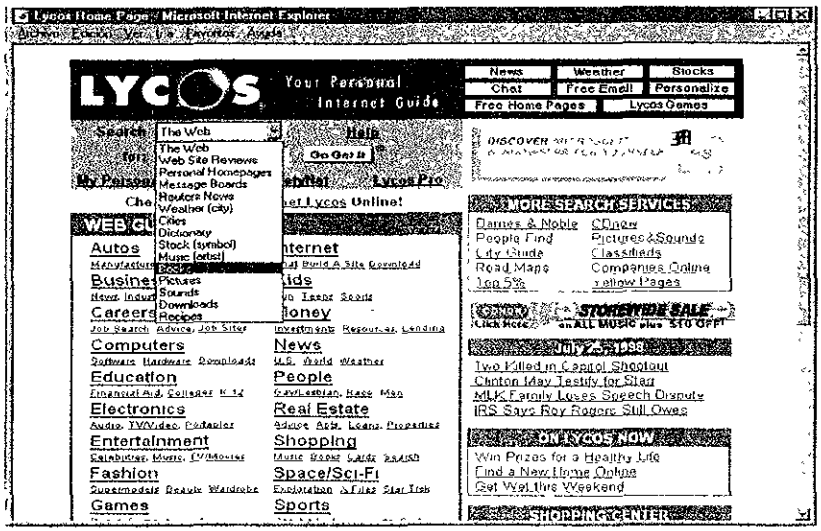


Figura 3 5 Servicio de Búsquedas Lycos

Página de inicio del servicio http://www.lycos.com/

La base de datos del servicio de búsqueda Lycos, está compuesta por páginas WWW, algunas ligas de FTP y Gopher, y sitios visitados con mayor frecuencia ordenados por tema.

Los usuarios pueden ejecutar búsquedas simples con base en palabras clave y utilizar los operadores booleanos (AND, OR), establecer la característica de relevancia más cercana o más lejana para recuperar de esta manera un mayor o menor número de documentos, y configurar como van a ser desplegados los resultados

El sistema ignora palabras comunes, realiza las búsquedas en el titulo, encabezado, ligas y las primeras 20 líneas de las páginas de Web, para las búsquedas por default asume OR entre los términos, las opciones de búsquedas personalizadas permiten especificar búsquedas de todos o ningún término, y tres niveles de detalles en la página de resultados, también permite búsquedas en sonidos e imágenes

Lycos también cuenta con Point Communications, que ofrece un servicio de noticias y un catálogo por temas de Sitios de Internet revisados, cada revisión incluye una descripción de una sentencia y rangos numéricos basados en el contenido, presentación y experiencia total.

Creadores· Carnegie Mellon University; Lycos, Inc

3.2.2 Características de productos comerciales para implantación de sistemas de Búsquedas

Una vez revisados algunos de los principales servicios de búsquedas que podemos encontrar en Internet, se presenta una descripción de las características principales de algunos productos diseñados para la implantación de servicios de búsqueda de localización de documentos a través de sitios WWW particulares, dichos productos se muestran en la tabla 3.1

Nombre	Características
Compass Server 3 0	Permite crear categorías organizadas en orden jerárquico Por medio de My Compass permite crear perfiles para entrega de Información personalizada a los usuarios, por email, netcaster o páginas de web. Para la defnición de los indices utiliza el formato RDM (Resource Description Messaging)
Excite for Web Servers	Utiliza métodos estadísticos avanzados, cada colección de documentos es analizado para correlación estadística entre términos y documentos, estas correlaciones son usadas para definir los conceptos que facilitan el proceso de recuperar información. No tiene robot, solo sirve para indexar documentos en el servidor de web en el que esta instalado.
Ultraseek Server	Permite la modificación de la interfaz al usuario por medio de la modificación directa a los archivos. Permite la administración del servidor Ultraseek a través del Web, así como la revisión del funcionamiento del robot en general. Reindexación automática y en tiempo real, lo que impide tener problemas como documentos duplicados, y urls no existentes.
Lycos Intranet Spider	Se pueden coleccionar 3 tipos de información acerca de un documento· Información del contenido (abstracta o texto completo), propiedades HTTP o propiedades Microsoft Office. El catálogo de información de Inmagic es una base de datos de texto, lo que permite anadir metadatos, borrar o añadir urls. El software consta de 3 productos. 1.El Spider en sí 2 DB/TextWebPublisher . Es el producto que permite realizar las busquedas, las cuales pueden ser locales o a través de la red por medio de un browser. 3.DB/TextWorks Permite elaborar las pantallas de presentación para el usuario (formas html y aplicaciones locales), crear la estructura de los campos que va a contener la base y elegir la información que se va a indexar.

Tabla 3.1 Características principales de productos comerciales para implantación de sistemas de búsquedas

3.3 Alternativas de solución para optimizar las búsquedas

La problemática de las búsquedas de información en WWW se puede resolver por medio del uso de alguna de las herramientas mencionadas anteriormente, sin embargo la eficiencia de éstas herramientas está íntimamente ligada a la capacidad de procesamiento del equipo en el que se encuentra implementado, y cuando la cantidad de información o el número de usuarios que hacen uso de éstas herramientas se incrementan, se sobrepasa la capacidad de los equipos, disminuyendo su rendimiento y en consecuencia el de los sistemas de búsqueda, dando lugar a largos tiempos de respuesta al momento de llevar a cabo las consultas y en general a un rendimiento deficiente del sistema de búsquedas.

Para solucionar el problema planteado tenemos 2 alternativas.

La opción inmediata es intentar resolverlo por medio de computadoras con procesadores más veloces, con mayor cantidad de memoria y con el espacio suficiente para poder almacenar la información que va a componer el sistema.

La segunda opción consiste en utilizar varias máquinas para resolver el problema, dividiendo la información entre ellas, de esta forma podemos pensar en conjuntar el rendimiento individual de los equipos para que de esta manera podamos obtener la capacidad de procesamiento suficiente para que los sistemas de búsqueda trabajen de una manera eficaz.

Para poder realizar la evaluación de estas dos alternativas, es necesario revisar antes algunos conceptos relacionados con la forma en la que pueden ser implantadas las opciones anteriores, dichos conceptos están enfocados al cómputo paralelo y distribuido.

3.4 Cómputo Paralelo y Distribuido

3.4.1 Cómputo Paralelo

A diferencia de otras tecnologías que han excitado la imaginación de la humanidad en el pasado como la energía solar, la energía nuclear, los viajes espaciales, etc , que han pasado fugazmente de moda, las computadoras han transformado la sociedad y seguramente lo continuará haciendo por muchas décadas

Dos grandes inventos auspiciaron esta revolución· el concepto del programa almacenado y el transistor El programa almacenado permite dar flexibilidad a las computadoras, por lo que se pueden realizar gran variedad de tareas con el mismo hardware. El transistor permitió la creación de electrónica pequeña, robusta y rápida.

La unión de estos dos conceptos en un sólo circuito integrado trajo como consecuencia el nacimiento del microprocesador. Desde su creación en 1971 los avances de la tecnología han permitido que su velocidad haya aumentado 25000 veces y la escala de integración más de 1 millón de veces.

La tendencia en computación, desde la década de los 60's había sido la de incrementar el rendimiento de los procesadores y con esto aumentar la capacidad de cómputo. El problema de esta tendencia era que, hacer mejoras en el rendimiento de los procesadores implicaba, entonces y hoy en día, un incremento en el costo de los mismos, el cual al principio de cada generación de procesadores era pequeño comparado con las mejoras en el procesamiento (poco incremento en el costo y gran mejora en procesamiento), mas se llega a un punto en el que los pocos avances en procesamiento tienen un costo excesivo, lo que hace incosteable el seguir la investigación de dicha familia de procesadores

A partir de la década de los 90's el costo de los procesadores permite incorporar la idea de paralelismo en computación como una opción para aumentar la capacidad de cómputo sin la restricción del costo de las mejoras en el rendimiento de los procesadores.

La idea de paralelismo no es nueva, en computación consiste en utilizar más de una unidad de procesamiento para resolver simultánea y coordinadamente diferentes partes de un problema de cómputo de manera que el rendimiento se incremente en forma lineal con el número de procesadores

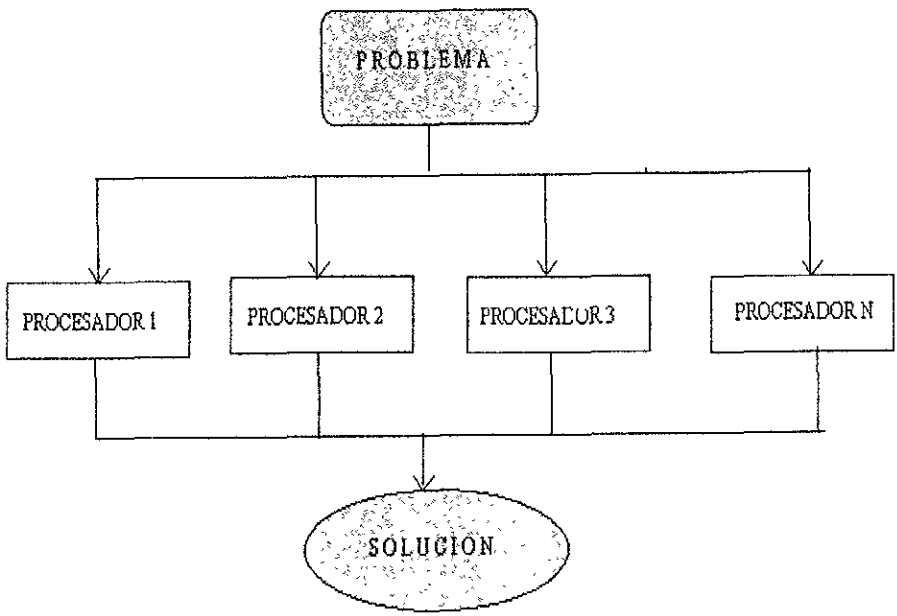


Figura 3.1 Paralelismo

Una computadora paralela es una colección de elementos de procesamiento que pueden trabajar concurrentemente, y comunicarse a través de una interconexión de alta velocidad El paralelismo es atractivo porque provee una alternativa para lograr un mejor rendimiento de la máquina y de la memoria, independientemente de las mejoras en la tecnología.

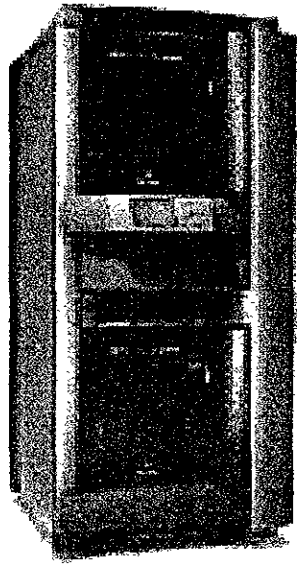


Figura 3.2. Supercomputadora Origin 2000 Gabinete con 40 procesadores MIPS R10000, a 195 Mhz. 390 Millones de operaciones de punto flotante por segundo (Mflops) por procesador .

Existen varias aplicaciones que pueden ser implementadas por medio del cómputo paralelo, algunas de ellas son:

a) Resolver problemas matemáticos y de Ingeniería más rápidamente.
b) Ejecutar tareas independientes simultáneamente.
c) Mejorar la ejecución de las simulaciones.
d) Incrementar la respuesta de programas en tiempo real
e) Hacer las búsquedas en BD más rápidamente.

En las aplicaciones anteriores identificamos dos razones por las cuales se debe utilizar el cómputo paralelo:

a) Atacar problemas largos o más ambiciosos, que requieren más tiempo, más memoria, o ambos
b) Dar solución el mismo problema en menos tiempo

3.4.1.1 Paradigmas de paralelización

Las herramientas de software y hardware existentes generalmente toman uno de los dos mayores enfoques de ejecución de programas paralelos:

a) Paso de mensajes.
b) Memoria compartida

Estos dos paradigmas difieren en varios aspectos, pero lo más importante es la forma en la que almacenan los datos que son compartidos entre los varios componentes del programa paralelo y la forma de poner los datos disponibles a los componentes que se necesitan para que el programa corra.

Paso de mensajes

El paso de mensajes es un modelo que surge de la arquitectura de multiprocesadores de memoria distribuida y redes de estaciones de trabajo.

Para la programación en memoria distribuida, se debe considerar que el sistema es multicomputadoras y en el cada procesador tiene su propia memoria privada, además de estar distribuidos todos los procesadores a través de una red de interconexión entre ellos.

Una característica importante de los sistemas de memoria distribuida es su fácil escalabilidad, pero con la desventaja de que se debe programar toda la comunicación entre procesadores

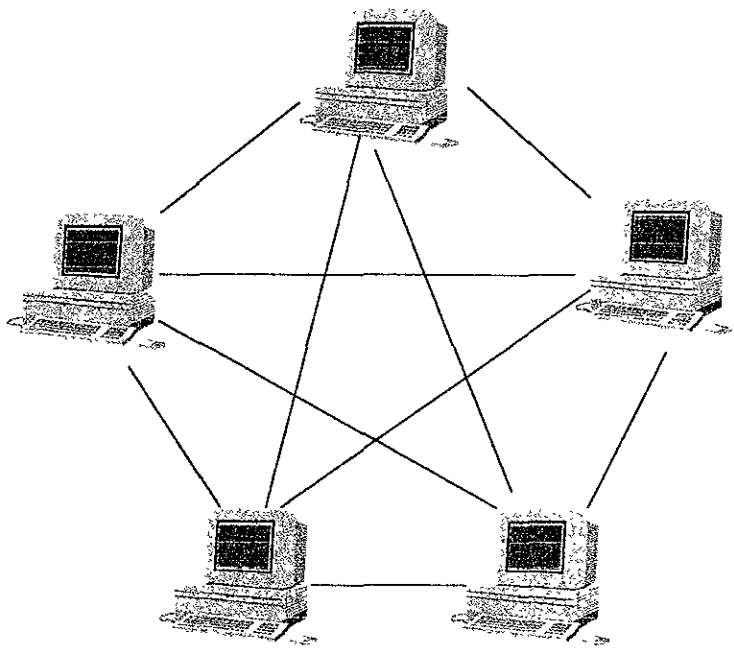


Figura 3.4 Ambiente basado en paso de mensajes.

Memoria compartida

En el caso de memoria compartida se supone que todos los procesadores están conectados a la memoria principal del sistema multiprocesadores y cada uno puede leer y escribir sobre esta memoria global, una de las características de este caso es que la programación es más sencilla pero es muy difícil y caro el tratar de escalar este tipo de sistemas

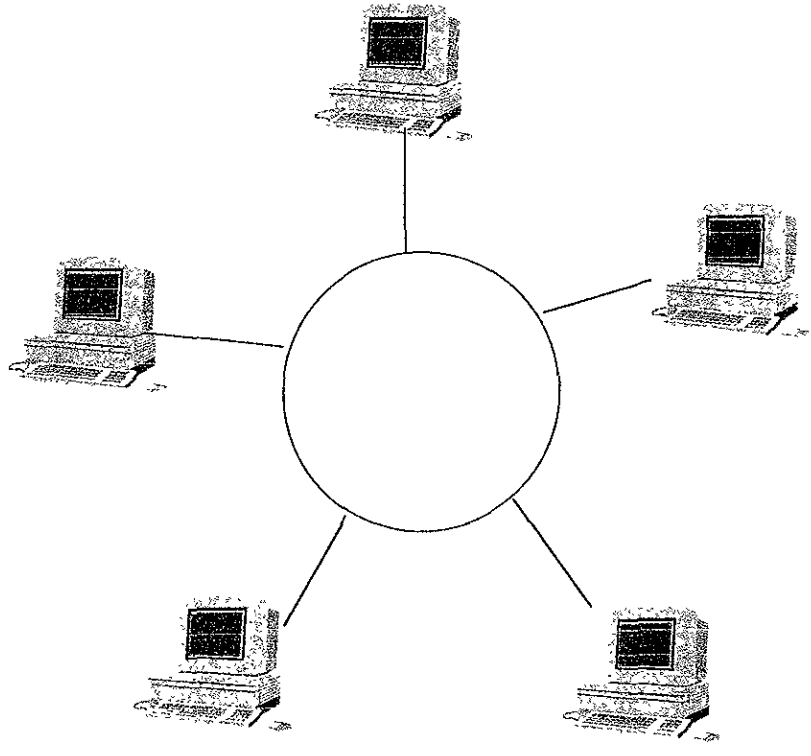


Figura 3.4 Ambiente basado en memoria compartida

3.4.1.2 Arquitecturas Paralelas

Cualquier computadora, ya sea secuencial o paralela, ejecuta instrucciones sobre datos. Un flujo de instrucciones (el programa) le indica a la computadora qué hacer en cada paso, y éstas instrucciones afectan a un flujo de datos (las entradas del programa). De acuerdo a la forma según la cual el conjunto de datos es afectado por el conjunto de instrucciones, pueden ser definidos diferentes modelos de computadoras, y es así como Flynn definió los siguientes cuatro modelos, dependiendo de si existen uno o múltiples flujos de instrucciones o datos, ejecutados u operados por un procesador.

a.) Simple flujo de Instrucciones - Simple flujo de Datos (SISD)
b.) Múltiple flujo de Instrucciones - Simple flujo de Datos (MISD)
c) Simple flujo de Instrucciones - Múltiple flujos de Datos (SIMD)
d) Múltiple flujos de Instrucciones - Múltiple flujos de Datos (MIMD)

a) Computadoras SISD

Una computadora SISD consiste de una sola unidad de procesamiento recibiendo una secuencia de instrucciones que opera sobre una secuencia de datos En cada paso, la unidad de control emite una instrucción que opera sobre datos obtenido de la unidad de memoria Este es el modelo Von Neumann, al cual corresponden la mayoría de las computadoras actuales, no presenta ningún paralelismo.

Figura 3 5 Computadoras SISD

b) Computadoras MISD

En este modelo p procesadores (p>1), cada uno con su propia unidad de control, comparten una unidad de memoria común donde residen los datos Existen p secuencias de instrucciones y una secuencia de datos En cada paso, un dato recibido desde la memoria es operado por todos los procesadores simultáneamente, cada uno con las instrucciones que el procesador recibe de su unidad de control. Así el paralelismo es alcanzado realizando diferentes operaciones sobre el mismo dato No existen computadoras comerciales que se ajusten a este modelo.

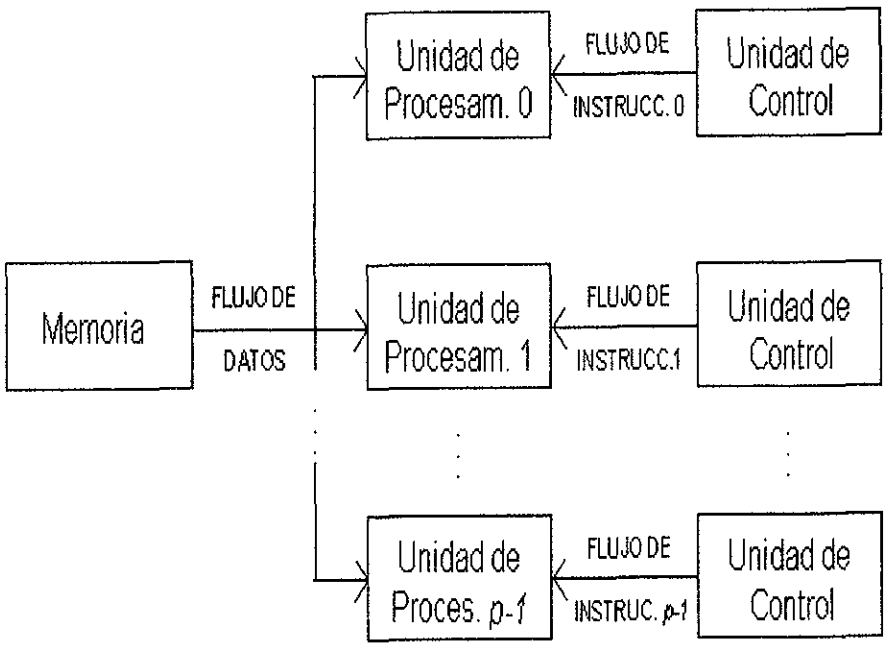


Figura 3.6 Computadoras MISD

c) Computadoras SIMD

En este modelo una computadora paralela consiste de p procesadores idénticos (p>1), cada uno operando con su propia memoria local Todos los procesadores operan bajo el control de una sola secuencia de instrucciones emitida por una unidad de control central.Existen p secuencias de datos, una por procesador, los cuales operan de manera síncrona. En cada paso, todos los procesadores ejecutan la misma instrucción cada uno sobre un dato diferente

En la mayoría de las aplicaciones interesantes que se quieran resolver sobre este tipo de computadoras, es deseable que los procesadores puedan comunicarse entre sí durante el cálculo a fin de intercambiar datos o resultados intermedios. Esto puede ser realizado de dos maneras diferentes, a través de una memoria común (computadoras SIMD a memoria compartida) o a través de una red de interconexión (computadoras SIMD a memoria distribuida).

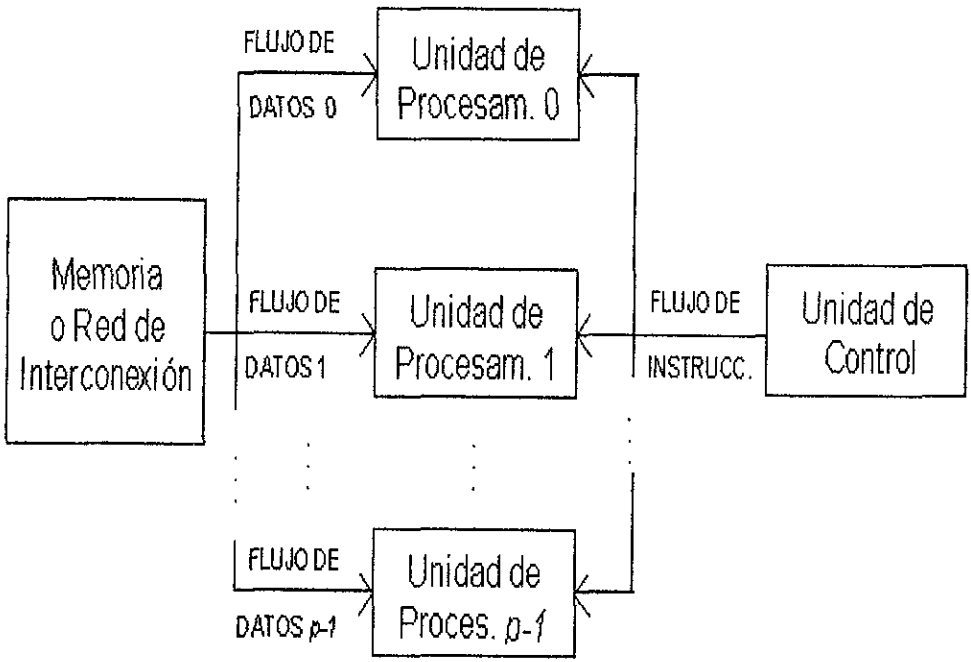


Figura 3 7 Computadoras SIMD

d) Computadoras MIMD

Esta clase de computadoras es la más general y más poderosa en el paradigma de la computación paralela que los clasifica de acuerdo a la secuencia de datos y/o instrucciones Este tipo de computadoras poseen p procesadores (p>1), cada uno operando bajo el control de una secuencia de instrucciones emitida por su propia unidad de control Así los procesadores están potencialmente todos ejecutando diferentes programas sobre datos diferentes. Esto significa que los procesadores operan de manera asíncrona, sin embargo, los algoritmos asíncronos son difíciles de diseñar, evaluar e implantar.

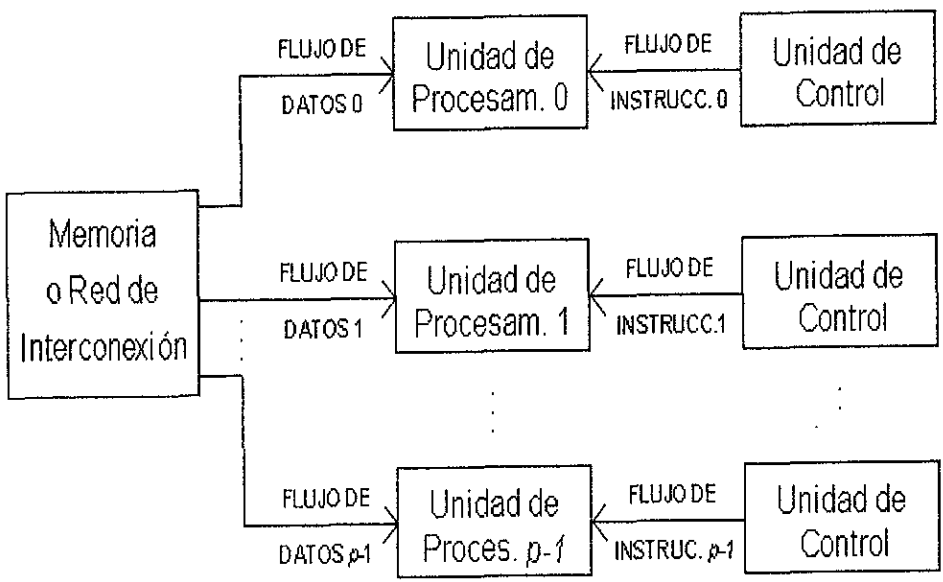


Figura 3 8 Computadoras MIMD

d.1) Arquitecturas MIMD a Memoria Compartida

Esta clase de computadoras es también conocida como modelo de máquina paralela de acceso aleatorio (PRAM) Aquí p procesadores (p>1) comparten una memoria común Cuando dos procesadores quieren comunicarse lo hacen a través de esta memoria común. Si se desea transmitir un dato desde el procesador Pi al procesador Pj , esto lo realiza en dos pasos, en el primero, el procesador Pi escribe el dato en una dirección de memoria conocida por el procesador Pj. En el segundo paso el procesador Pj lee esa localización.

El modelo básico permite a todos los procesadores acceder la memoria compartida simultáneamente, si la posición de memoria que ellos están tratando de escribir es diferente. Sin embargo, el tipo de memoria puede dividir este modelo en cuatro subclases dependiendo de si dos o más procesadores pueden acceder a la misma posición de memoria simultáneamente [AKL-89], a saber·

1. Lectura-Exclusiva, Escritura-Exclusiva.
2. Lectura-Concurrente, Escritura-Exclusiva
3. Lectura-Exclusiva, Escritura-Concurrente
4. Lectura-Concurrente, Escritura-Concurrente

El permitir múltiples lecturas simultáneas sobre la misma posición de memoria no debe ocasionar ningún problema. Conceptualmente, si cada procesador requiere leer desde una misma posición de memoria copia el contenido de esa posición y lo almacena en su memoria local. Sin embargo, si varios procesadores requieren escribir simultáneamente diferentes datos sobre la misma posición de memoria, debe existir una manera determinística de especificar el contenido de esa posición de memoria una vez realizadas las escrituras En estos casos (subclases 3 y 4), los conflictos de escritura se resuelven por hardware, mientras que para las subclases 1 y 2, es el sistema operativo que los resuelve, dando como resultado que en diferentes ejecuciones de un mismo programa pueden tener resultados distintos, dada la aleatoriedad descrita

d.2) Arquitecturas MIMD a Memoria Distribuida

La otra forma de comunicación entre los procesadores es a través de una red de interconexión. En este modelo la memoria es dividida entre el conjunto de procesadores, para su acceso local. Además, cada procesador es conectado con sus vecinos a través de una línea bidireccional de comunicación, la cual le permite enviar o recibir datos en cualquier instante de tiempo, habiéndose desarrollado una amplia variedad de topologías que permiten abarcar una gran cantidad de problemas de manera eficiente, tales como. el arreglo lineal, el anillo, la malla, el toroide, el árbol, el fat-tree, y el hipercubo [RUK-94], que tienen una baja cantidad de enlaces entre procesadores, de manera tal que cuando sea necesario comunicar un mensaje entre dos procesadores que no tienen conexión directa, debe encaminarse o enrutarse dicho mensaje por procesadores intermedios entre éstos dos

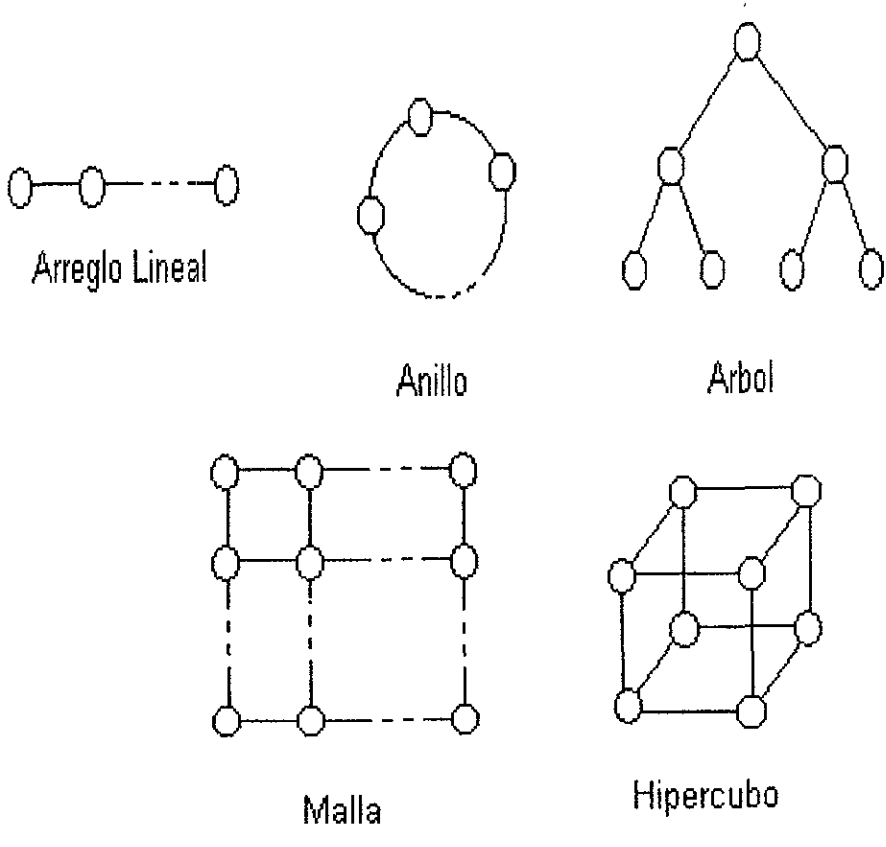


Figura 3.9 Arquitecturas MIMD a Memoria Distribuida

Otro representante importante de este tipo de arquitectura son las máquinas basadas en transputers. La palabra Transputer, formada de la unión de los términos TRANsistor y comPUTER, señala el interés de sus creadores de proponer un componente que, al igual que los transistores en los circuitos eléctricos, sirva de base para los sistemas masivamente paralelos [RUK-94]. Un transputer es un microcomputador que contiene un procesador, memoria local y capacidad de comunicaciones a través de enlaces que permiten conectarlo con otros transputers u otros dispositivos, todo en un solo chip.

Cada transputer cuenta con cuatro enlaces de comunicación que le permiten formar parte de redes de procesadores en diversidad de topologías Sin embargo, se dipone de switches programables que permiten conectar cualquier canal de un transputer con diferentes canales en diferentes instantes de tiempo, y además, permiten que la comunicación entre canales pueda ser realizada entre transputers ó entre un transputer y otro dispositivo. Al conectar varios procesadores entre sí de acuerdo a alguna topología, se crean grandes máquinas de procesamiento paralelo a un costo relativamente bajo.

Generalmente, es el transputer 0 el considerado "raíz", por estar conectado a la computadora anfitrión ó Host, pero, como se dijo, cualquier transputer puede además estarlo, y ese hecho varía de una arquitectura a otra.

3.4.2 Cómputo Distribuido

El cómputo distribuido es un proceso en donde un conjunto de computadoras conectadas entre sí a través de una red, son usadas colectivamente para resolver un gran problema.

En un sistema distribuido, la existencia de múltiples computadoras autónomas es transparente para el usuario. El usuario puede teclear un comando para correr un programa, y observar que corre. El hecho de seleccionar el mejor procesador, encontrar y transportar todos los archivos de entrada al procesador y poner los resultados en el lugar apropiado, depende del sistema operativo o de programas secundarios controladores.

Debido a que más y más organizaciones tienen redes de área local de gran velocidad interconectando varias estaciones de trabajo de propósito general, los recursos computacionales combinados pueden exceder el poder de una sola computadora de alto rendimiento. En algunos casos, varias MPPs han sido combinados usando computo distribuido para producir un poder de procesamiento sin igual.

Algo que es común entre el cómputo distribuido y MPP es la noción de paso de mensajes. En todo procesamiento paralelo, los datos deben ser intercambiados entre tareas que cooperan unas con otras, y el modelo más ampliamente utilizado es el paso de mensajes, debido a la perspectiva del número y variedad de multiprocesadores que soporta, así como los términos de las aplicaciones, lenguajes, y el software que lo utilizan.

En un MPP cada procesador es excatamente como todos los demás, en capacidad, recursos, software, y velocidad de comunicación, no es así en una red. Las computadoras disponibles en una red pueden estar hechas por varios vendedores, o tener diferentes compiladores. Por este motivo, cuando un programador desea explotar una colección de computadoras en red, se encuentra con varios problemas de heterogeneidad·

a) Arquitectura
b) Formato de datos
c) Velocidad de la computadora
d) Carga de la máquina
e) Carga de la red

El conjunto de computadoras disponible puede incluir una gran variedad de tipos de arquitecturas, tales como 386/486, estaciones de trabajo de alta velocidad, multiprocesadores de memoria compartida, supercomputadoras y aún mas, grandes MPPs.
Los formatos de datos son, la mayoria de las veces, incompatibles, esta incompatibilidad es un punto importante en el cómputo distribuido, debido a que los datos enviados de una computadora pueden ser ilegibles para la computadora que los recibe. Los paquetes de paso de mensajes desarrollados para ambientes heterogéneos deben asegurarse de que todas las computadoras van a entender los datos que se intercambian, aún si el conjunto de computadoras está conformado por un grupo de estaciones de trabajo con el mismo formato de datos, pueden existir problemas de heterogeneidad debido a las diferentes velocidades de los procesadores.

Al igual que la carga en la máquina, el tiempo necesario para enviar un mensaje a través de la red puede variar dependiendo de la carga de la red que depende de otros usuarios, este tiempo de envío se vuelve importante, cuando una tarea esta bloqueada esperando un mensaje, y es aún más importante, cuando el algoritmo paralelo es sensitivo al tiempo de llegada del mensaje.

Por lo anterior, en el cómputo paralelo, la heterogeneidad puede aparecer dinámicamente, aún en las tareas más simples.

A pesar de las numerosas dificultades causadas por la heterogeneidad, el cómputo distribuido tiene varias ventajas.

a) Usando el hardware existente, el costo de este tipo de cómputo puede ser muy bajo.
b) El rendimiento se puede optimizar asignando cada tarea individual a la arquitectura más apropiada.
c) Podemos explotar la naturaleza heterogénea de la computación, una red de computadoras heterogéneas, no solamente es una red de área local conectando estaciones de trabajo Por ejemplo, provee acceso a diferentes bases de datos o a procesadores especiales para las partes de la aplicación que pueden correr solamente en determinada plataforma.
d) Los recursos de la computadora virtual pueden crecer por pasos y tomar ventaja de las tecnologías de cómputo y de red más novedosas.
e) El desarrollo de los programas se puede mejorar por medio del uso de ambientes de desarrollo familiares, los programadores pueden utilizar editores, compiladores, y depuradores que están disponibles en las máquinas individuales.
f) Las computadoras y las estaciones de trabajo individuales son usualmente estables y la experiencia en su uso ya se encuentra disponible.
g) La tolerancia a fallas al nivel del usuario o del programador puede ser implementada con poco esfuerzo en la aplicación o en el sistema operativo.
h) El cómputo distribuido facilita el trabajo colaborativo.

Todos estos factores se traducen en un menor tiempo de desarrollo y depuración, costos reducidos y posiblemente implantaciones de las aplicaciones más efectivas.

3.5 Análisis de alternativas de solución

Como se mencionó en el capítulo 2, el número de sitios del dominio WWW de la UNAM actualmente alcanza un total de 257[1] con 1,568 700 de documentos, con un promedio de 4.32 KB por documento[3]

Con base en la información anterior, se requiere alrededor de 6.77 GB de espacio de almacenamiento, para contener todos los documentos y poder realizar las búsquedas de información sobre los mismos.

Para la implantación de la solución 1 propuesta en el punto 3.3, correspondiente al uso de una computadora más robusta, se requiere de una inversión para la adquisición de dicha computadora, debido a que actualmente, en la dependencia no se cuenta con el equipo de cómputo que pueda albergar dicha cantidad de información; además de los costos de mantenimiento que, por el hecho de utilizar computadoras más robustas, se incrementan.

Para la implantación de la segunda alternativa concerniente al uso de cómputo distribuido, al que hemos hecho referencia en el punto 3.4, la coordinación cuenta con 9 estaciones de trabajo, de las cuales 6 son de la marca Sun con Sistema Operativo Solaris 2.6 y las 3 restantes son Silicon Graphics con Sistema Operativo Irix 6.3.

Es necesario mencionar que las 6 computadoras son de producción y tienen sistemas en operación, sin embargo, dichos sistemas no consumen toda la capacidad de dichas máquinas.

En total, reuniendo el espacio libre solamente de 4 de las estaciones de trabajo Sun, cubrimos las necesidades de espacio para el almacenamiento del total de los documentos

Tomando en cuenta los factores anteriores, y debido a que existen varios productos que nos permiten unir varias computadoras, conectadas en red, para lograr ejecutar aplicaciones en un ambiente distribuido, el proyecto se implantará a través de la segunda alternativa

3. Fuente de información Estadísticas generadas por el programa de recopilación de información "scooter" del producto Altavista Search IntraNet Extensions

3.6 Elección del software para la implantación del sistema distribuido

A continuación se analizarán algunos productos que poseen características que pueden ser de utilidad para implementar el sistema distribuido sobre el cual se instalará el sistema de búsquedas, asimismo se presenta la elección de la aplicación más adecuada para el sistema que nos ocupa

1. Nombre de la aplicación:
Linda

Descripción :

Es un lenguaje de coordinación que suple el "pegamento" necesario para unir varios procesadores independientes en un programa paralelo, provee una memoria virtual que es compartida por todos los procesadores de un programa en paralelo. Provee un conjunto de comandos que permite la creación, sincronización y comunicación de procesos.
Los procesos en un programa paralelo linda ejecutan intercambio de datos por medio de objetos en la memoria virtual.

Esquema que utiliza:
Memoria Virtual

Software comercial.

2. Nombre de la aplicación:
Sfgate.

Descripción :

SFgate es una aplicación que sirve de comunicación entre el Web y Wais escrito en Perl SFgate puede accesar a cualquier numero de bases de datos en cualquier lugar de la red al mismo tiempo. Solo se deben especificar en la forma HTML las diversas fuentes en donde se van a llevar a cabo las consultas.

SFgate no realiza llamadas a los programas nativos de Wais que realizan las búsquedas, el mismo programa se conecta a los servidores y ejecuta las búsquedas.

Esquema que utiliza:
CGI escrito en perl.

Software libre

3. Nombre de la aplicación:
Condor.

Descripción :

Otro enfoque para capturar el poder de conjuntos de estaciones de trabajo, ha sido el desarrollo de sistemas de administración de recursos distribuidos (RM). Un sistema RM provee una interfaz la cual le permite a los usuarios encontrar recursos fácilmente en los cuales correr sus trabajos y de monitorear el estado de los trabajos al momento de la corrida Típicamente los sistemas RM usan una interfaz de batch la cual le permite a los usuarios mandar una colección de trabajos y poder correr estos trabajos en cualquier lugar en el que los recursos estén disponibles Los esfuerzos que han sido guiados por este método se han enfocado en mecanismos y políticas para identificar recursos disponibles en un conjunto de computadoras, para poner en ellos los trabajos encolados.

Condor permite ejecutar trabajos en un conjunto de estaciones de trabajo corriendo bajo la plataforma UNIX. Los trabajos son encolados y ejecutados remotamente en las estaciones de trabajo en periodos de tiempo en los que las máquinas se encuentran inactivas, es decir que no están ejecutando procesos.
Los trabajos migran de computadora en computadora sin la intervención del usuario.

El programa que nos permite mandar trabajos para que se ejecuten en las maquinas remotas, lee un archivo de descripción que contiene comandos que permiten controlar el encolado de los trabajos.

Esquema que utiliza:
Paralelismo adaptivo.

Software libre.

4. Nombre de la aplicación:

LAM/MPI (Local Area Multicomputer/Message Passing Interface)

Descripción :

Es un ambiente de programación MPI para computadoras heterogéneas en una red. Con LAM, un conjunto de procesadores dedicado o una infraestructura de red puede actuar como una computadora paralela resolviendo un problema.

LAM corre en cada computadora como un demonio, estructurado como un nano-kernel y procesos virtuales hand-threaded.

Una característica importante de LAM es que tiene control de la multicomputadora. Hay muy pocas cosas que no se pueden hacer en tiempo de corrida. Un conjunto inicial de nodos LAM es iniciado con las herramientas que el sistema presenta, Sin embargo el administrador de recursos puede ajustar las computadoras que se encuentran en una sesión de LAM en tiempo de corrida.

Cuando alguno de los nodos de la red se cae, el evento es detectado por LAM y todas las aplicaciones sobrevivientes son informadas de que un nodo se ha caído. La librería MPI reacciona invalidando todas las comunicaciones que incluyan procesos del nodo muerto.

Esquema que utiliza:

Paso de mensajes.

Software libre.

5. Nombre de la aplicación:
PVM (Parallel Virtual Machine)

Descripción :

El sistema esta compuesto de 2 partes, la primera parte es un demonio que reside en todas las computadoras que forman la maquina virtual. Esta diseñado para que cualquier usuario con una cuenta en una máquina lo pueda instalar. Cuando un usuario quiere correr una aplicación PVM, primero debe de crear una maquina virtual inicializando PVM La aplicación PVM puede ser inicializada desde el prompt de unix de cualquiera de las computadoras que forman el conjunto.

La segunda parte consiste de un conjunto de librerías de rutinas de interfaces de PVM Contiene un repertorio completo de primitivas que son necesarias para la cooperación entre tareas de una aplicación. Esta librería contiene rutinas que pueden ser llamadas por el usuario para el paso de mensajes, expansión de procesos, coordinación de tareas, y modificación de la maquina virtual.

Soporta el paralelismo funcional y el paralelismo de datos

Esquema que utiliza:
Paso de mensajes.

Software libre.

Finalmente, y después de haber presentado las características principales de los productos anteriores, en la tabla 3. Se presenta la evaluación de dichos productos

Nombre de la aplicación	Comentario
Linda	El software es comercial, y existe software disponible de forma gratuita, que realiza funciones parecidas a Las de Linda. Las ventajas que ofrece el sistema Linda están basadas en el uso de la memoria Virtual, sin embargo debido a que en el sistema que se va a desarrollar no existe un intercambio de mensajes intensivo, las ventajas presentadas por este sistema no son relevantes.
Sfgate	El esquema que presenta SFgate no implementa de forma adecuada el control de errores, cuando por alguna razón alguna de las máquinas que forman parte del sistema de búsquedas queda fuera de servicio, el sistema la deja fuera y no asigna el trabajo a otro procesador
Condor	Por la naturaleza del sistema, es útil para ejecutar trabajos que requieren procesamiento largo tipo batch, tiene mecanismos de chequeo, por que los trabajos se ejecutan en las máquinas que se encuentran ociosas dependiendo de los parámetros que el usuario especifica en el archivo de descripción, cuando la maquina se ocupa, el trabajo es detenido y en el archivo de chequeo se establece el grado de avance y el estado del trabajo, para moverlo a otra maquina. No es apropiado para tareas que requieran respuesta inmediata en tiempo real

LAM/MPI	MPI no fue diseñado para ser un sistema completo, que provee la infraestructura necesaria para ser utilizado en cómputo distribuido, no incluye características como administración de procesos, configuración de la máquina virtual y soporte de entrada/salida.
PVM	La librería de PVM presenta una gran tolerancia a fallas, debido a que si una máquina falla, lo notifica, la quita del conjunto y nos da la posibilidad de agregar otra máquina que supla las funciones de la que falló. Con PVM logramos obtener interoperabilidad entre sistemas heterogéneos y presenta funciones más sencillas a las que presenta LAM/MPI. Permite la creación de procesos dinámicos una vez que la máquina virtual ya fue inicializada.

Tabla 3.2 Evaluación de las aplicaciones presentadas.

Con base en la información presentada anteriormente, el producto a utilizar en la implantación del sistema distribuido, es la biblioteca de envío de mensajes PVM, debido a la tolerancia a fallas que presenta, la facilidad de instalación y de inicialización de la máquina virtual, y el dinamismo presentado en el tiempo de corrida de la máquina virtual, lo que se refleja en un control efectivo de los recursos que forman parte del conjunto de computadoras que forman parte de la aplicación.

3.7 Esquema del sistema de búsquedas propuesto

La forma en la que se integran las aplicaciones con la biblioteca de envío de mensajes PVM, para lograr que dichas aplicaciones se ejecuten en el ambiente distribuido, es a través de la compilación de las aplicaciones junto con la librería que provee PVM, para lograrlo, se debe modificar el código fuente de las aplicaciones para hacer llamadas a las funciones de PVM para paso de mensajes.

Los productos comerciales de servicios de búsqueda presentados en la tabla 3.1, distribuyen solamente los archivos en forma binaria, por lo que no es posible la integración con PVM, por esta razón, es necesario utilizar un sistema que nos permita tener acceso a los archivos con el código fuente de la aplicación

Tomando en cuenta que, debido a la forma en la que están constituidos los sistemas de búsquedas, en donde se distinguen los 2 componentes principales descritos en el tema 3 2, que son la herramienta de recopilación de información y el programa de búsquedas, es posible, en nuestro caso dividir éstas 2 funcionalidades para la implantación del sistema, teniendo, por una parte el programa de recolección de los documentos HTML del dominio WWW de la UNAM, y por la otra el programa de búsquedas, que es el que se va a incorporar en el sistema distribuido

Siguiendo con el esquema anterior, el uso de software libre es una opción económica y útil, por lo que, haremos uso de éste, por medio del programa de recolección de información MOMSpider, y del indexador freeWAIS-sf, que servirá como programa de búsquedas.

La elección de freeWAIS-sf está basada en la gran variedad de opciones para la especificación de la cadena de búsquedas que presenta al usuario, y en que la distribución del software es por medio de los archivos fuente, con la necesidad de la compilación respectiva, requerida para su uso, además de que es posible instalarlo en máquinas con sistema operativo UNIX.

Una vez que se ha presentado el análisis de cada una de las partes que van a componer el sistema y ya que se han elegido los elementos necesarios para la integración del mismo, en la figura 3.10, se presenta un esquema general donde podemos apreciar la interacción que existirá entre ellos.

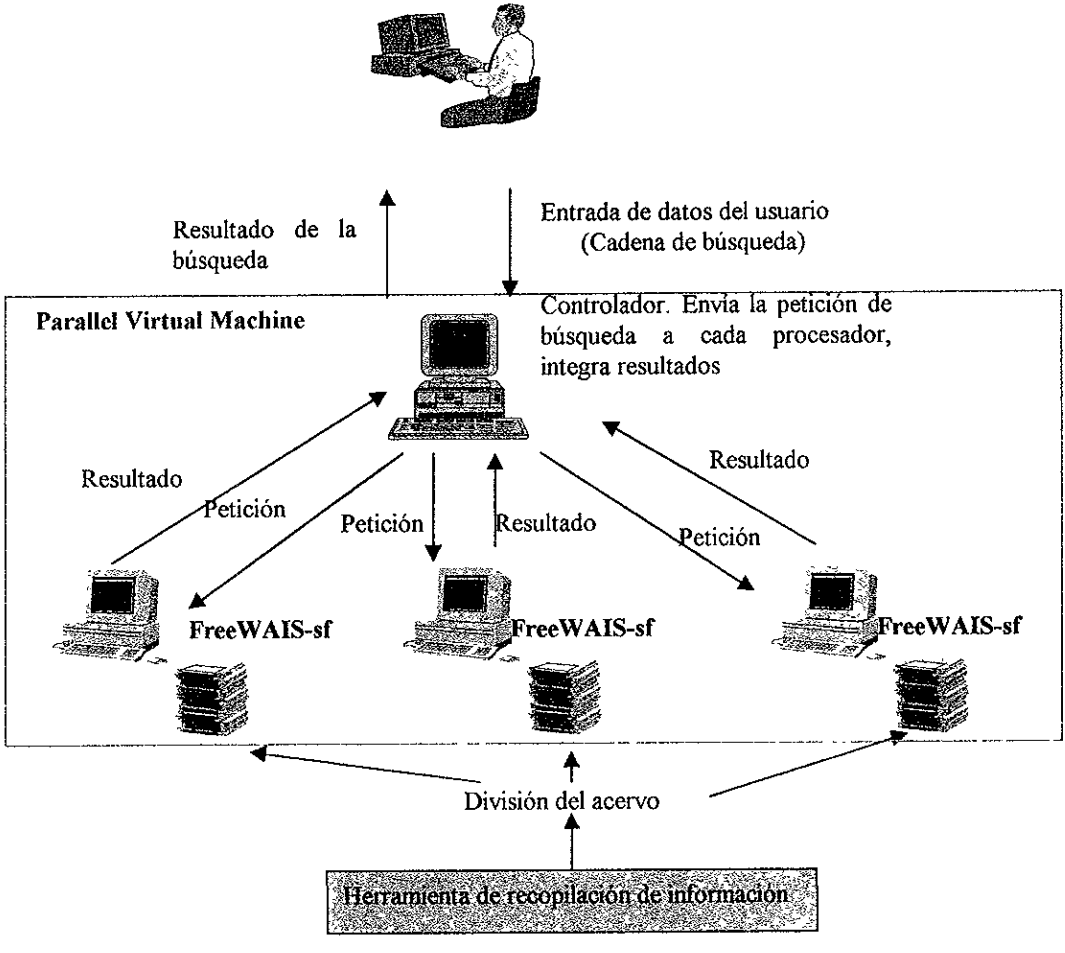


Figura 3.10 Esquema general del sistema propuesto

Capítulo 4

Configuración e implementación de componentes

4.1 Sistema Distribuido

4.1.1 Descripción general de Parallel Virtual Machine

PVM es el resultado de un proyecto de investigación de cómputo en redes heterogéneas. Las metas generales de este proyecto son investigar temas y desarrollar soluciones para el cómputo concurrente y heterogéneo

PVM es un conjunto integrado de herramientas de software y librerías que simulan un ambiente de trabajo en cómputo heterogéneo y concurrente de propósito general, formado por computadoras de diversas arquitecturas, interconectadas entre sí El objetivo principal del sistema PVM es permitir que esa colección de computadoras pueda ser utilizada de forma cooperativa para lograr la concurrencia o el cómputo paralelo.

Los principios sobre los cuales se basa PVM incluyen lo siguiente:

1) Conjunto de computadoras configuradas por el usuario. Las tareas de las que están constituidas las aplicaciones se ejecutan en un conjunto de máquinas, las cuales son seleccionadas por el usuario, dichas máquinas son necesarias para que el programa PVM inicie su ejecución Pueden formar parte del conjunto de máquinas desde computadoras personales hasta máquinas de múltiples procesadores (incluyendo máquinas de memoria compartida y de memoria distribuida). El conjunto de máquinas puede ser alterado añadiendo o eliminando máquinas durante el tiempo de ejecución, lo cual es una característica importante para tolerancia a fallas.

2) Acceso transparente al hardware Los programas de aplicación pueden ver el ambiente de hardware como una colección de elementos virtuales de proceso, o pueden explotar las capacidades de máquinas específicas que forman parte del conjunto, enviando ciertas tareas a las computadoras más apropiadas

3) Cómputo basado en procesos. La unidad de paralelismo en PVM es una tarea, una hebra de control secuencial e independiente, que alterna entre comunicación con otras tareas y proceso independiente No existe un mapeo de proceso a procesador en PVM, particularmente, múltiples tareas pueden ser ejecutadas en un solo procesador.

4) Modelo de paso de mensajes explícito El conjunto de tareas de cómputo, cada una ejecutando una parte de la carga total de trabajo, se comunica a través del envío y recepción de mensajes entre tareas. El tamaño del mensaje es limitado solamente por la cantidad de memoria disponible.

5) Soporte de heterogeneidad. El sistema PVM soporta heterogeneidad en términos de máquinas, redes, y aplicaciones. En lo que respecta al paso de mensajes, PVM permite que los mensajes que contienen más de un tipo de datos puedan ser intercambiados entre máquinas que tienen diferentes representaciones de datos

6) Soporte de multiproceso. PVM aprovecha las facilidades del paso de mensajes que existe en los multiprocesadores para tomar ventaja del hardware sobre el cual se encuentra instalado, frecuentemente los vendedores proveen sus propias versiones de PVM optimizadas para sus sistemas, dichas versiones pueden comunicarse con la versión pública de PVM.

El sistema PVM está compuesto de 2 partes·

La primera parte es un demonio, llamado pvmd3 que reside en todas las computadoras que conforman la máquina virtual. (Un ejemplo de un programa denominado demonio es el programa de correo que siempre se está ejecutando y administra todos los correos electrónicos que entran y salen de la computadora). pvmd3 está diseñado para que cualquier usuario con una cuenta válida en alguna máquina con Sistema Operativo Unix pueda instalarlo. Cuando un usuario desea ejecutar una aplicación PVM, primero crea una máquina virtual iniciando PVM. La aplicación PVM puede ser iniciada desde el prompt de Sistema Operativo Unix en cualquiera de las computadoras que van a formar parte de la máquina virtual.

La segunda parte del sistema es una librería de rutinas de interfaz de PVM. Contiene un repertorio de primitivas que son necesarias para la cooperación entre las tareas que conforman una aplicación. Esta librería contiene rutinas del usuario para el paso de mensajes, creación de procesos, coordinación de tareas y modificaciones a la máquina virtual

El modelo de cómputo de PVM está basado en la noción de que una aplicación consiste de varias tareas Cada tarea es responsable de una parte de la carga total de trabajo

En ocasiones, una aplicación es paralelizada a través de sus funciones, esto es, cada tarea ejecuta una función diferente, por ejemplo, entrada, establecimiento del problema, solución, salida y despliegue. Este proceso es frecuentemente llamado paralelismo funcional.

Un método más común de paralelizar una aplicación es llamado paralelismo de datos. En este método todas las tareas son las mismas, pero cada una de ellas se encarga solamente de una pequeña parte de los datos.

PVM soporta cualquiera de los dos métodos de paralelismo o una mezcla de ambos. Dependiendo de sus funciones, las tareas se pueden ejecutar en paralelo y pueden necesitar sincronización o intercambio de datos, a pesar de que no siempre sea este el caso

El siguiente diagrama ejemplifica el modelo de cómputo de PVM

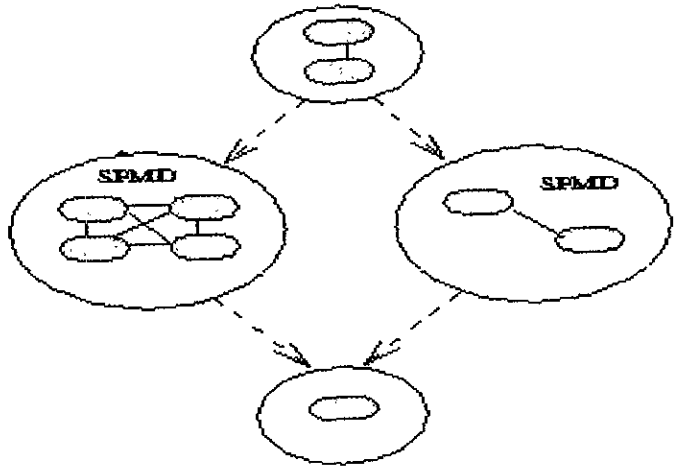


Figura 4 1 Modelo de Cómputo de PVM

El sistema PVM soporta los lenguajes C, C++ y Fortran Estos lenguajes han sido elegidos, debido a que son los que predominan en la mayoría de las aplicaciones

Las librerías de interfaz para el usuario de PVM en los lenguajes C y C++, fueron implementadas como funciones, siguiendo las convenciones generales utilizadas por la mayoría de los sistemas en C, incluyendo los Sistemas Operativos al estilo Unix

Los programas de aplicación escritos en C y en C++ accesan las funciones incluidas en la librería de PVM, a través del ligado del programa de aplicación, con la librería libpvm3.a que es parte de la distribución estándar.

Todas las tareas de PVM son identificadas por un valor entero llamado identificador de tarea (TID). Los mensajes son recibidos por y enviados hacia los identificadores de tarea. Debido a que los identificadores de tarea no se deben repetir en toda la máquina virtual, éstos son asignados por el programa pvmd, y no son elegidos por los usuarios. A pesar de que PVM codifica la información dentro de cada TID, es de esperarse que el usuario los trate como simples identificadores enteros. PVM contiene varias rutinas que retornan valores de los TIDs, para que de esta manera, la aplicación del usuario pueda identificar otras tareas en el sistema.

Existen aplicaciones en donde es natural pensar en grupos de tareas, por el contrario, hay casos en los que el usuario necesitará identificar sus tareas por los números 0 – (p -1), donde p es el número de tareas.

PVM incluye el concepto de grupos creados por el usuario. Cuando una tarea se une a un grupo, se le asigna un número de instancia único en ese grupo. Los números de instancia inician en 0 y se van incrementando. Siguiendo con la filosofía de PVM, las funciones para manejo de grupos son diseñadas para ser muy generales y transparentes para el usuario. Por ejemplo, cualquier tarea de PVM puede unirse o dejar cualquier grupo cuando así lo requiera, sin tener que informar a otras tareas en los grupos afectados También los grupos se pueden superponer y las tareas pueden enviar mensajes a diversos grupos de los cuales no son miembros. Para utilizar cualquiera de las funciones para el manejo de grupos, el programa se debe ligar con la librería libgpvm3.a.

El paradigma general para programar aplicaciones con PVM es la siguiente

El usuario escribe uno o más programas secuenciales en C, C++ o Fortran 77, dicho programa contiene llamadas a la librería de PVM. Cada programa corresponde a una tarea que compone toda la aplicación.

Los programas son compilados para cada arquitectura que conforma la Máquina Virtual, y los archivos objetos resultantes son colocados en un lugar accesible a todas las computadoras de la máquina virtual.

Para ejecutar una aplicación, el usuario típicamente inicia la copia de una tarea a mano (usualmente la tarea amo) desde una de las máquinas dentro de la máquina virtual. Este proceso inicia subsecuentemente otras tareas de PVM, resultando, eventualmente en una colección de tareas activas que procesan localmente e intercambian mensajes con otras tareas para resolver los problemas.

El siguiente programa ilustra los conceptos básicos de la programación con PVM, este programa está diseñado para invocarse manualmente, después de imprimir su identificador de tarea (el cual se obtiene por medio de la función *pvm_mytid()*), inicia una copia de otro programa llamado *hola_esclavo*, utilizando la función *pvm_spawn()*. Una iniciación exitosa, ocasiona que el programa ejecute una recepción por bloqueo utilizando *pvm_rec*. Después de recibir el mensaje, el programa imprime el mensaje enviado por su contra parte y también su TID, el mensaje es extraído del buffer utilizando *pvm_upstr*. La llamada final *pvm_exit* retira el programa del sistema PVM

```
main()
{
  int cc, tid, msgtag;
  char buf[100];

  printf("Yo soy:  %x\n", pvm_mytid()),

  cc = pvm_spawn("hola_esclavo", (char**)0, 0, "", 1, &tid);

  if (cc == 1) {
      msgtag = 1;
    pvm_recv(tid, msgtag);
    pvm_upkstr(buf);
    printf("De %x: %s\n", tid, buf),
  } else
    printf("No puedo iniciar hola_esclavo \n"),

  pvm_exit(),
}
```

El siguiente programa, es un listado del programa esclavo, su primer acción en PVM es obtener el TID del amo, utilizando la llamada *pvm_parent.* Posteriormente el programa obtiene el nombre de la máquina en la que se está ejecutando, y la envía al amo utilizándola secuencia de tres llamadas, *pvm_initsend* que inicializa el buffer que va a enviar los datos, *pvm_pkstr* para colocar la cadena dentro del buffer de envío de una forma independiente de la arquitectura, y *pvm_send* para transmitir el mensaje al proceso destino, especificado por *ptid*, etiquetando el mensaje con el número 1.

```
#include "pvm3.h"

main()
{
  int ptid, msgtag;
  char buf[100],

  ptid = pvm_parent();

  strcpy(buf, "Hola a todos de: ");
  gethostname(buf + strlen(buf), 64),
   msgtag = 1;
  pvm_initsend(PvmDataDefault);
  pvm_pkstr(buf),
  pvm_send(ptid, msgtag);

  pvm_exit();
}
```

4.1.2 Instalación de la Máquina Virtual

Una de las razones principales de la popularidad de PVM es que es simple de configurar y de utilizar. PVM no requiere privilegios especiales para ser instalado. Cualquier persona con una cuenta válida en una máquina Unix puede hacerlo.

El primer paso para poder establecer la máquina virtual es obtener e instalar la librería de PVM. La dirección de ftp anónimo por medio de la cual se puede obtener la versión más actual del código fuente de PVM es ftp.netlib org en el directorio pvm3. Si contamos con un navegador de WWW podemos bajar la librería de la dirección:
http //www netlib org/pvm3/index html

El nombre del archivo que debemos bajar depende de las plataformas que vamos a incluir en el conjunto de computadoras que van a integrar la máquina virtual En este punto es importante recordar que la máquina virtual puede estar formada de un conjunto de máquinas heterogéneas.

Las plataformas que soporta la librería de PVM son

PCs

Pentium II, Pentium Pro, Pentium, Duals and Quads	Win95, NT 3.5.1, NT 4.0 Linux, Solaris, SCO, NetBSD, BSDI, FreeBSD
MAC	NetBSD
Next	
Amiga	NetBSD

Estaciones de trabajo y servidores con memoria compartida.

SUN3, SUN4, SPARC, UltraSPARC	SunOS, Solaris 2.x
IBM RS6000, J30	AIX 3.x, AIX 4.x
HP 9000	Hpux
DEC Alpha, Pmax, microvax	OSF, NT-Alpha
SGI	IRIS 5 x, IRIS 6.x

Computadoras paralelas.

Cray YMP, T3D, T3E, Cray2
Convex Exemplar
IBM SP2, 3090 NEC SX-3
Fujitsu
Amdahl
TMC CM5
Intel Paragon
Sequent Symmetry, Balance

En el caso que nos ocupa, y como ya se mencionó anteriormente, el sistema se va a implementar en máquinas SPARC con Sistema operativo Sun Solaris versión 2 5.1, por lo que el archivo que necesitamos es pvm3.4.beta6.tar.gz

Una vez que tenemos el software almacenado localmente en cada una de las máquinas que formarán parte de la máquina virtual, debemos proceder a la instalación, que consta de varios pasos, los cuales se detallan a continuación.

Descomprimir el archivo.
servicio% gzip -d pvm3.4 beta6.tar.gz | tar xvf -

Lo cual nos creará un directorio raíz llamado pvm3, con varios subdirectorios, que contienen, entre otras cosas, códigos fuente para obtener los ejecutables de PVM, algunos ejemplos de programas, archivos de configuración Make para todas las arquitecturas que soporta PVM, y documentación.

Para poder continuar con la instalación de PVM es necesario establecer 2 variables de ambiente, las cuales son utilizadas por PVM al momento de inicializar y de ejecutar la máquina virtual, cada usuario necesita establecer estas dos variables para utilizar PVM.

La primer variable es PVM_ROOT, cuyo valor es la ruta del directorio en donde se descomprimió el software de PVM, la segunda variable es PVM_ARCH, la cual le indica a PVM la arquitectura de la máquina, para que de esta manera PVM utilice los archivos ejecutables correspondientes que se encuentran en el directorio establecido por PVM_ROOT.

El método más fácil es establecer estas variables en el archivo cshrc (en caso de estar utilizando csh), en nuestro caso las variables se establecieron de la siguiente manera:

```
setenv PVM_ROOT /home/servicio/pvm3
setenv PVM_ARCH SUN4SOL2
```

El siguiente paso es compilar los códigos fuente que se encuentran incluidos en la distribución, para lograrlo, solamente es necesario teclear make en el directorio pvm3, (el directorio donde se descomprimieron los archivos), Makefile compila y genera el programa deminio ejecutable pvmd3, la librería de C libpvm3 a, la librería de Fortran libfpvm3.a, y el programa cliente pvm.

Una vez que se ha terminado de compilar, las librerías y ejecutables se encuentran instalados en /home/servicio/pvm3/lib/SUN4SOL2.

4.1.3 Configuración y ejecución de la máquina virtual

Para iniciar la consola y el demonio de PVM, vamos a utilizar los scripts pvm y pvmd que se encuentran en el directorio /home/servicio/pvm3/lib/. Ellos determinan la arquitectura de la máquina y ejecutan los programas que se encuentran en /home/servicio/lib/ARCH, para que éstos archivos se puedan ejecutar, es necesario agregar el directorio /home/servicio/lib/ARCH a la variable PATH de csh

Para que PVM se ejecute en todas las computadoras que van a formar parte de la máquina virtual, es necesario tener acceso a todas ellas por medio del comando rsh (remote shell), para lo cual debemos permitir el acceso a cada una de las máquinas, esto se logra creando un archivo sobre el directorio raíz del usuario en las máquinas, llamado .rhosts que contiene la dirección IP de las máquinas a las que se le va a permitir el acceso, así como el login específico con el cual se va a conectar el usuario.

Para nuestro caso:

```
servicio% more .rhosts
132.248.71.82        servicio
132.248.71.87        servicio
132 248.71.81        servicio
132.248.71.4         servicio
```

4.2 Herramienta de recopilación de información

4.2.1 Descripción general de MOMSpider

MOMSpider es una herramienta de recopilación de información que se especializa en el mantenimiento de las estructuras de información distribuidas basadas en hipertexto, que conforman el World Wide Web.

MOMSpider utiliza una lista de instrucciones que detallan qué estructuras de información debe recorrer, a quien le notificará los problemas, y en donde se debe colocar la información recopilada. Con base en esta lista, el programa recorre cada una de las estructuras de información y cubre los requerimientos listados.

El diseño de MOMSpider se enfoca en cumplir con los requerimientos necesarios para el mantenimiento de Sitios de Web, y en minimizar los efectos que pueda producir en los servidores que visita y el ancho de banda de las redes.

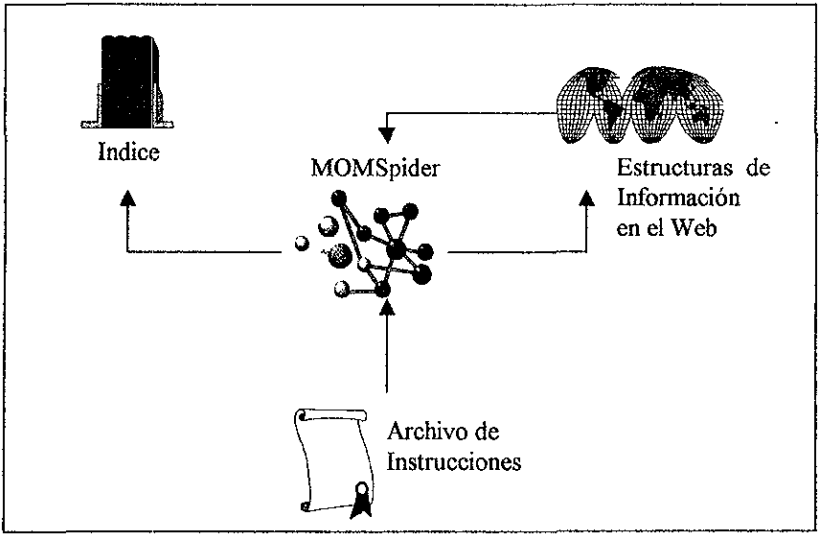


Figura 4.2 Elementos relacionados con MOMSpider

MOMSpider obtiene sus instrucciones leyendo un archivo de texto que contiene una lista de opciones y tareas que deben ser ejecutadas. Cada tarea describe una estructura de Web específica, que debe ser abarcada por el programa en su proceso de recorrido.

Una instrucción de tareas incluye el tipo de recorrido, el nombre de la estructura de información, el URL raíz en donde se va a iniciar el recorrido, la localidad en donde se va a colocar la salida indexada, una dirección de correo electrónico que corresponde al dueño de la estructura de información, y un conjunto de opciones que determinan qué acciones justifican el envío de un mensaje de correo

Para cada tarea, MOMSpider recorre la estructuras del Web, partiendo del documento especificado como raíz y siguiendo cada una de las ligas que se encuentran en el documento hasta llegar a un nodo hoja. Un nodo hoja se define como un objeto de información que no es de tipo HTML (por lo tanto no puede contener ligas, deteniéndose aquí el proceso de recorrido) o que está afuera del dominio de la estructura de información especificada.

MOMSpider determina los límites de una estructura de información, de acuerdo al tipo de recorrido especificado, que puede ser Sitio, Arbol o Dueño.

El recorrido por sitio especifica que cualquier URL que apunte a un sitio diferente al que se especifica en el documento raíz es un nodo hoja.

El recorrido por árbol especifica que cualquier documento que no se encuentre al nivel o por debajo del documento raíz es un nodo hoja, donde el nivel es determinado por la ruta que se encuentra en el URL.

El recorrido por dueño especifica que cualquier documento fuera de la raíz que no contenga información del dueño en el encabezado igual al nombre de la estructura de información es un nodo hoja.

La información generada por cada tarea se almacena en un índice con formato tipo HTML, en el archivo especificado en las instrucciones. El índice contiene la siguiente información:

- Información correspondiente a cómo y dónde fue generado el índice.
- Ligas de hipertexto a la versión anterior del documento de índices.
- Lo siguiente por cada documento que no es del tipo hoja, que es accesible a través de la raíz:
  - Una liga que conduce al documento real
  - Información del documento que se encuentra en el encabezado del mismo(Título, Fecha de modificación, fecha de expiración, etc)
  - Una lista de todas las referencias de hipertexto únicas hechas por el documento, cada referencia incluye:
    a) El tipo de referencia (liga, pregunta, imagen, etc.)
    b) Una liga que duplica la referencia
    c) Información del documento que se encuentra en el encabezado del mismo(Título y Fecha de modificación)
    d) Si el objeto referenciado está dentro de la actual estructura de información, se incluye una liga adicional para poder saltar a su propia entrada en el documento de índices.
- Una lista de ligas de referencias cruzadas que apuntan a cambios interesantes como se muestra en las entradas del índice.

Uno de los puntos principales en el diseño de MOMSpider es la eficiencia, particularmente en lo que se refiere al uso del ancho de banda. Sería irresponsable desarrollar una herramienta de recopilación de información que consuma los recursos de red que existen en Internet. Por esta razón, MOMSpider minimiza la carga en el ancho de banda de la red por medio del uso de la petición HEAD para probar ligas, manteniendo un listado de los nodos que ya han sido recorridos, agrupando múltiples tareas dentro de una misma ejecución, y permitiendo al usuario restringir el recorrido de ciertos URLs. También permite que el usuario especifique varios prefijos de URLs que siempre se deben evitar

Un segundo punto en el diseño de MOMSpider es minimizar el impacto en los proveedores de información y al mismo tiempo maximizar los beneficios indirectos que se reciben del proceso del recorrido. Todas las peticiones de HTTP son similares a:

HEAD /path HTTP/1 0
User-Agent. MOMspider/0 1
From: user@machine.sub.dom.ain
Referer: http //www site edu/current/document html

Lo anterior le permite reconocer a los administradores de los servidores la fuente de la petición, y si es necesario, poner restricciones sobre una herramienta de recopilación de información en particular. También provee información útil, incluyendo la forma de contactar a la persona que está ejecutando el programa.

Como una precaución adicional, MOMSpider revisa periódicamente y cubre cualquier restricción mencionada en el documento /robots.txt, que es el estándar propuesto por Martijn Koster, para impedir el acceso a un sitio de Web a determinados programas, especificados por el administrador del sitio.

4.2.2 Instalación de MOM Spider

MOMSpider está escrito en lenguaje PERL, por lo cual, para poder instalarlo, es necesario tener instalada la distribución binaria de PERL así como las librerías. En nuestro caso contamos con la versión 4.5004_01 de PERL para Unix.

También debemos tener la librería de PERL libwww-perl, la cual se puede obtener de las direcciones:

http //www cis ufl edu/perl/
http //web nexor co uk/perl/perl.html

Una vez que tenemos los requisitos anteriores, el siguiente paso es obtener la distribución de MOMSpider, a través de los sitios.

http //www ics uci.edu/WebSoft/MOMspider/
ftp //liege ics uci edu/pub/arcadia/MOMspider/

El archivo que debemos bajar es MOMspider-1.00.tar.gz, el cual se debe descomprimir una vez que se tiene almacenado en la máquina local por medio del comando.

servicio% gzip –d MOMspider-1.00.tar.gz
servicio% tar xvf MOMspider-1.00.tar

El directorio creado por el comando tar, contiene programas en PERL para la correcta ejecución de MOMSpider, así como documentación del producto y ejemplos de los archivos de instrucciones.

Los programas escritos en PERL incluidos en la distribución hacen referencia al directorio /usr/local/bin/ que es donde se debe encontrar el intérprete de PERL, la ruta del directorio se debe modificar para que apunte a la dirección del ejecutable de PERL en la máquina local, en nuestro caso, la ruta se modificó a #!/opt/leng/perl5.004/bin/perl

También debemos establecer la variable de ambiente MOMSPIDER_HOME cuyo contenido debe ser el directorio en donde se encuentra la distribución de MOMSpider

servicio% setenv MOMSPIDER_HOME /home/servicio/MOMspider-1.00

Aseguramos que el programa momspider tenga permisos de ejecución

servicio% chmod 755 momspider

Por último, debemos establecer las opciones de configuración por default editando el archivo momconfig.pl y asignando valores a las diferentes variables en PERL que se indican en el archivo, algunas de las opciones que debemos establecer son:

- Dominio de la red local
- Rutas de archivos en donde se especifican los sitios que no son permitidos, y los sitios en donde se debe verificar si existen restricciones de acceso.
- Máxima profundidad del recorrido, máximo número de tiempo de respuesta en segundos, y dirección del url inicial

4.2.3 Configuración del proceso de recorrido

Como se mencionó anteriormente, MOMSpider utiliza varios archivos que determinan su comportamiento en el proceso de recorrido, dichos archivos son, el archivo de instrucciones, el archivo donde se especifican los sitios iniciales, los sitios no permitidos y los sitios hojas

Archivo de Instrucciones

El archivo de instrucciones consiste de una serie de directivas globales, seguidas de series de tareas de recorrido

MOMSpider establece las opciones de configuración asociadas con las directivas y procede a ejecutar cada una de las tareas listadas en el orden indicado

El formato del archivo de instrucciones es bastante rígido. Las líneas en blanco y las líneas que inician con # son ignoradas. Todas las demás directivas deben estar en una sola línea y no existen caracteres de continuación de línea. Las instrucciones de tareas tienen la forma general <TYPE >.

Todas las instrucciones son sensibles a mayúsculas y minúsculas.

Las directivas globales se deben listar en la parte más alta del archivo de instrucciones, una por línea.

Las tareas de recorrido consisten de un conjunto de directivas entre los signos de mayor y menor, y el tipo de recorrido. Las tareas son ejecutadas en el orden especificado en el archivo. En general, es más útil listar las tareas ordenadas por su jerarquía.

Las directivas de tareas y las directivas globales permitidas se explican en el apéndice II.

Los archivos de configuración utilizados en el sistema, se muestran en el apéndice III.

4.2.4 Recolección de información

El siguiente paso, una vez establecidas todas las opciones de configuración, es poner a funcionar el robot, para el proceso de recolección de información, lo cual se hace a través del programa momspider

Lo ideal en nuestro caso es establecer los horarios en los que se va a ejecutar el programa de forma periódica, para que de esta manera siempre obtengamos información actualizada Los horarios elegidos, son los fines de semana, evitando así saturar la red en días normales de trabajo.

Para lograrlo, se agrega una entrada en el crontab, que es un programa utilizado para manipular las tablas usadas para controlar el programa demonio cron, que nos permite ejecutar programas de forma regular.

Servicio% crontab -e

Y en el editor que nos presenta, se debe agregar una entrada de acuerdo a los siguientes campos.

comando minutos horas fecha_dia fecha_mes dia_semana

Para nuestro caso

momspider 0 0 - - 0

De esta manera, obtenemos la información que el robot recupera en sus proceso de recorrido, iniciando en el sitio www.unam.mx , que es el sitio de entrada al conjunto de sitios html que conforman el dominio de la UNAM

Lo siguiente para la consecución final del sistema, es proceder a la fase de indexado y búsqueda en este acervo de información.

4.3 Programa de búsqueda

4.3.1 Descripción general de freeWAIS-sf

El protocolo WAIS Servicio de Información de área amplia (Wide Area Information Service) está basado en el estándar ANSI Z39.50 Versión 1, el cual especifica un servicio de la capa de aplicación OSI para permitir que una aplicación en una computadora realice una búsqueda en una base de datos que se encuentra en una máquina remota. Sin embargo, WAIS solamente utiliza un subconjunto de características del Z39.50, no utiliza peticiones que requieren que el servidor preserve algún estado

La implementación original de WAIS trataba a los documentos como conjuntos de términos uniformes. Pero debido a que la mayoría de los documentos presentan estructuras internas, se hicieron mejoras en la implementación original, para permitir que el usuario utilice las estructuras en sus búsquedas

FreeWAIS-sf es una extensión del software freeWAIS distribuida por "Clearinghouse for Networked Information Discovery and Retrieval (CNIDR)". El sufijo SF en el nombre del software significa Campos Estructurados (Structures Fields), que es una característica de indexado y de búsqueda, que distingue este software de sus predecesores.

Esta basado en la versión 0.202 de este software, pero incluye y mejora varias de las características que freeWAIS-sf contiene.

Algunas de las extensiones más importantes de freeWAIS-sf incluyen:

1. Introducción estructuras de texto, fechas y campos numéricos dentro de un documento.

2. Soporte de búsquedas boleanas complejas.

3. El código fonético y el proceso de reducir una palabra a su palabra raíz, se pueden habilitar o desabilitar, para cada campo individual.

4 Para cada expresión, se puede definir una categoría semántica utilizando el operador = para texto y ==, <, > para números.

5. Soporte de conjuntos de caracteres específicos de países.

6. La definición del formato y distribución del documento de los encabezados son configurables por medio de una nueva especificación de lenguaje basada en expresiones regulares No se debe escribir código en C para indexar nuevos tipos de documentos

7. El proceso de instalación solamente requiere correr un script en shell y responder varias preguntas.

Todos los cambios están restringidos al indexador y al servidor para permitir que los clientes existentes puedan hacer peticiones a las bases de datos de freeWAIS-sf.

4.3.2 Arquitectura Cliente/Servidor de freeWAIS-sf

Un sistema WAIS consiste de clientes comunicándose con un servidor a través de una red TCP/IP utilizando el protocolo WAIS. Los servidores responden las peticiones de búsquedas de los clientes utilizando estructuras de datos auxiliares llamados índices. Estos índices son generados a partir de los documentos originales por el programa waisindex. El servidor retorna las respuestas a las búsquedas recuperando parte de los archivos originales. Al conjunto de documentos con sus índices asociados se le denomina una base de datos de WAIS

El esquema 4.3 ejemplifica lo anterior

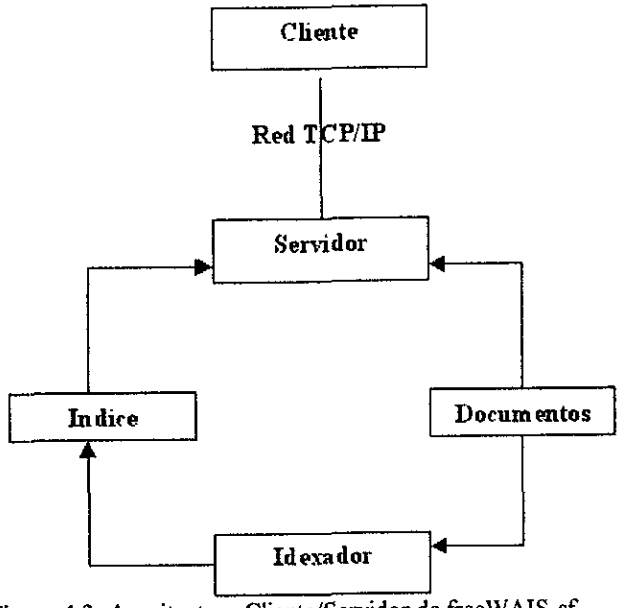


Figura 4.3 Arquitectura Cliente/Servidor de freeWAIS-sf

4.4 Integración de las partes que componen el sistema

4.4.1 Generación de índices

freeWAIS-sf crea 3 diferentes tipos de índices:

a)Locales. Los términos indexados en esta forma se pueden encontrar solo haciendo una búsqueda dentro de un campo especifico.

b)Globales. Los términos indexados en esta forma pueden ser encontrados haciendo una búsqueda global similar a la que se puede hacer con el WAIS tradicional.

c)Ambos. Los términos indexados de esta forma pueden ser encontrados de 2 formas: haciendo una búsqueda dentro de un campo especifico o haciendo una búsqueda global.

Hay algunas limitaciones concernientes a los índices de las bases de datos con freeWAIS-sf, que se deben conocer para evitar resultados inesperados

1. Los términos que tienen un solo carácter NO son indexados.
2. Los términos con más de 20 caracteres NO son indexados.
3. Hay una lista previamente construida de palabras de términos comunes en Ingles que NO son indexadas.
4. El número máximo de puntos que podemos recuperar NO esta claramente definido. Para un punto de vista practico el máximo parece ser alrededor de 250.
5. Se puede utilizar operadores boleanos en Inglés, Francés, Alemán, Italiano, Español, y Portugués y obtener los mensajes de regreso de la búsqueda en el idioma preferido.
6. Es mejor hacer las búsquedas en minúsculas.
7. Los caracteres ASCII que son indexados por default son A-Z , a-z y 0-9.
8. Los siguientes caracteres NO se pueden indexar en ningún caso: < > = ( ) , { } / y espacio.

Para generar los índices debemos utilizar el programa waisindex. Una lista completa de las opciones que presenta el programa, se presenta en el apéndice IV.

El programa waisindex utiliza un archivo de formato de descripción (Ver apéndice V. Formato de descripción de waisindex, para una descripción general del formato), el archivo utilizado en el sistema se muestra a continuación:

Archivo UNAM fmt

```
---------------------------------------------------
record-sep: /^\n/

layout:
headline: /^TITLE: / /^URL:/ 50 /^TITLE  */
headline: /^URL. / /\n:/ 70 /^URL: */
end:

region: /^TITLE: / /^TITLE: */
title "Titulo" SOUNDEX LOCAL
end: /^URL:/

region: /^URL: / /^URL: */
url "Url" TEXT LOCAL
end. /^\n:/
```

En nuestro caso la generación de índices se realizó por medio del comando waisindex con las opciones mostradas a continuación; en cada una de las máquinas que forman parte de la máquina virtual, en las que se van a llevar a cabo las consultas de forma distribuida.

```
servicio% waisindex -d /home/servicio/indices/UNAM -export -nocat -t fields
/home/servicio/info/UNAM.txt
```

Obteniendo así el conjunto de índices almacenado en /home/servicio/indices/.

Para una descripción más amplia de los archivos utilizados y generados por waisindex, ver apéndice VI.

4.4.2 Búsquedas en los Indices.

Con freeWAIS-sf tenemos una semántica más rica y nueva para especificar las cadenas de búsqueda, lo cual significa que la cadena debe tener una sintaxis especifica, por lo que el usuario puede tener errores al momento de hacer la búsqueda, por esta razón las categorías a ser buscadas se deben presentar con opciones de selección para cada termino.

A continuación se presenta una síntesis del lenguaje de las cadenas de búsqueda.

1. Expresiones atómicas de búsquedas
   - Términos (como biología)
   - Términos con metacaracteres (como biolo*)
   - Frases (como "biología molecular")

2. Raíz de palabras
   El uso de raíces de palabras es transparente para el cliente. Los términos buscados en una categoría de este tipo, son buscados usando su palabra raíz automáticamente. Para metacaracteres todos los términos de todas las palabras del diccionario que cumplan con la frase son usados como términos de búsqueda Las búsquedas de frases buscan las palabras en la cadena. Al menos uno de los términos de la frase debe ser un termino indexado. Entonces el servidor busca el documento conteniendo esta palabra para una cadena que cumpla con la frase completa. Esto significa que la búsqueda de cadenas trabaja solo si el servidor tiene acceso a los documentos.

3. Soundex y phonix

   Se permiten los operadores de prefijo soundex o phonix para convertir la cadena de búsqueda en su código Soundex/Phonix. Esto es muy útil por ejemplo cuando buscamos en una agenda de teléfonos y no sabemos exactamente como se escribe el nombre de la persona.

4. Operadores boleanos
   Se permite utilizar una combinación arbitraria de operadores boleanos "y", "o", y "no" (donde "no" significa "y no") Se pueden utilizar paréntesis para agrupar éstos operadores.

5 Categorías
   Para cada expresión, una categoría semántica (campo) puede ser definida usando el operador "categoria pred", donde pred puede ser:

   - Categorías de texto
     = (igual a)
   - Categorías numéricas
     == (igual a)
     <= (igual o menor a)
     < (menor a)
     > (mayor a)
     >= (igual o mayor a)

4.4.3 Interfaz al usuario

El último paso es la elaboración de la interfaz al usuario, la cual le presentará las pantallas de búsquedas con las diferentes opciones que presenta freeWAIS-sf y que se han mencionado en la sección previa.

Dicho sistema consiste de páginas html en conjunto con programas CGI elaborados en PERL.

Diversas pantallas que muestran el funcionamiento del servicio, se presentan a continuación.

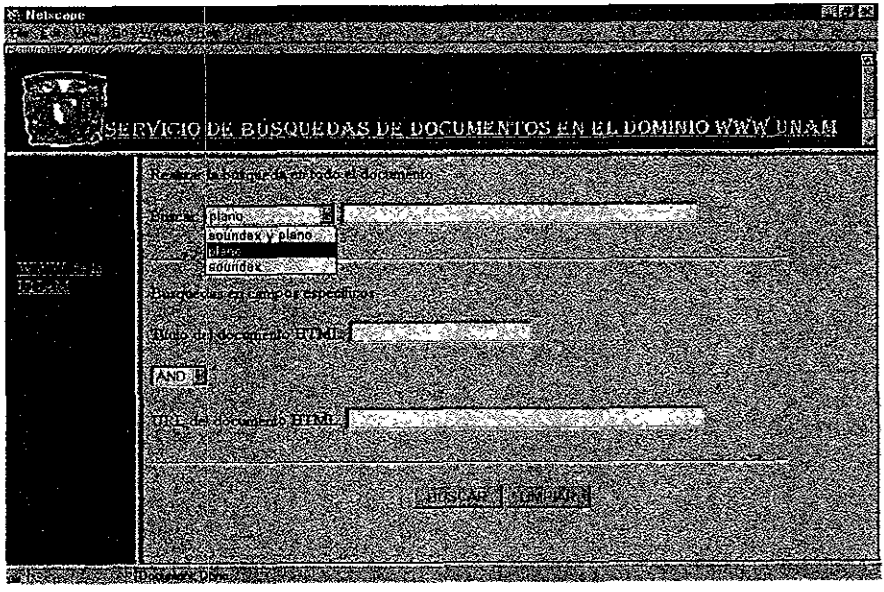


Figura 4.4 Página principal del servicio de búsquedas.

En la figura 4.4 podemos apreciar el sistema final, en el que tenemos las opciones de búsquedas siguientes:

Búsquedas en el título, en el URL o en todo el documento HTML.

Cuando realizamos una búsqueda en todo el documento, tenemos la opción de emplear la característica "parecido a" (soundex), tal como está escrito (plain) o una combinación de ambas características, así como también podemos utilizar los operadores boleanos AND, OR, NOT.

Si especificamos cadenas de búsqueda en el título y/o URL, podemos utilizar también los operadores boleanos, para restringir más nuestra búsqueda.

A continuación se presentan algunos ejemplos del uso de las características del servicio, y los resultados obtenidos.

La primera búsqueda se realizó en todo el documento y se especificó la cadena de búsquedas enep aragón, la petición de búsqueda se presenta en la figura 4.5 y los resultados obtenidos se presentan en la figura 4.6

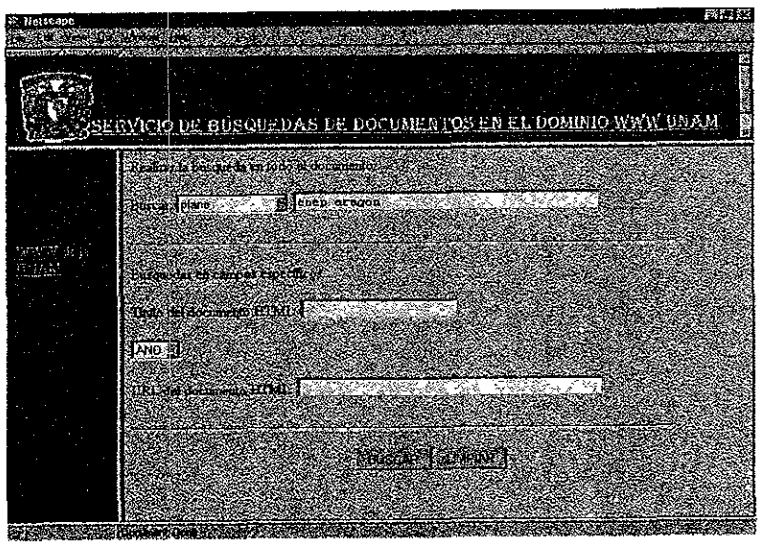


Figura 4.5 Petición de la cadena de búsquedas enep aragón en el documento completo

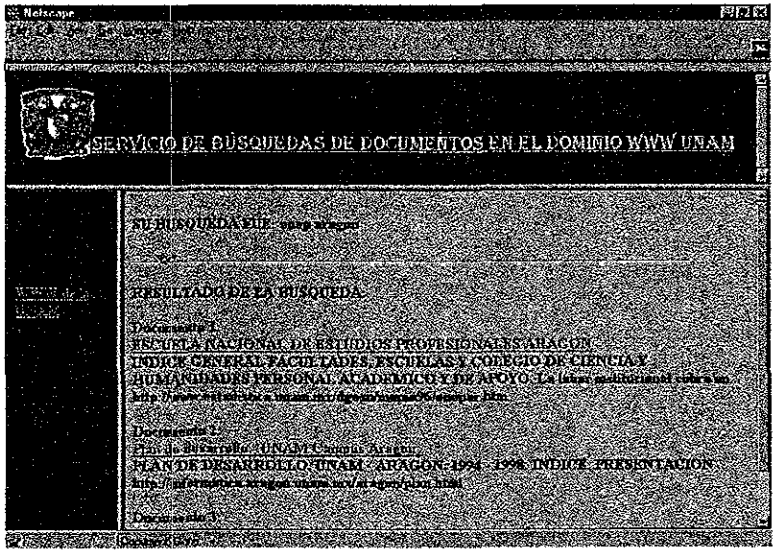


Figura 4 6 Resultados de la cadena de búsquedas enep aragón en el documento completo

La siguiente búsqueda se realizó solamente en el título del documento enviando la misma cadena de búsquedas enep aragón, la petición de búsqueda se presenta en la figura 4.7 y los resultados obtenidos se presentan en la figura 4.8.

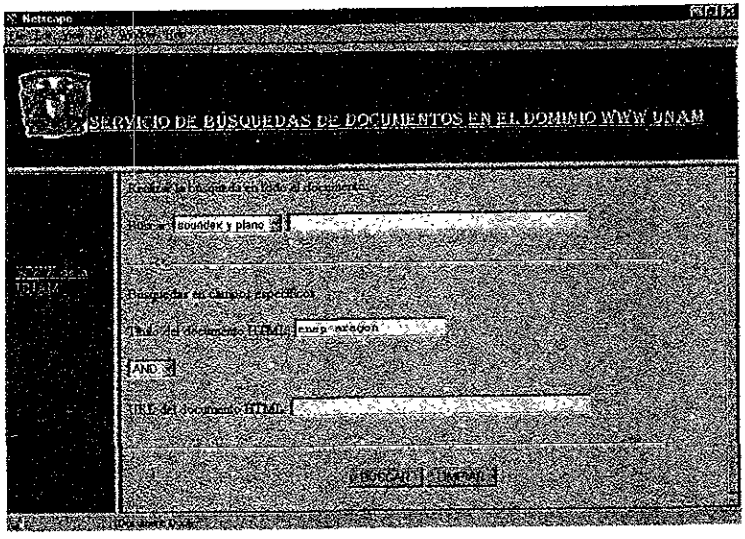


Figura 4.7 Petición de la cadena de búsquedas enep aragón en el título del documento

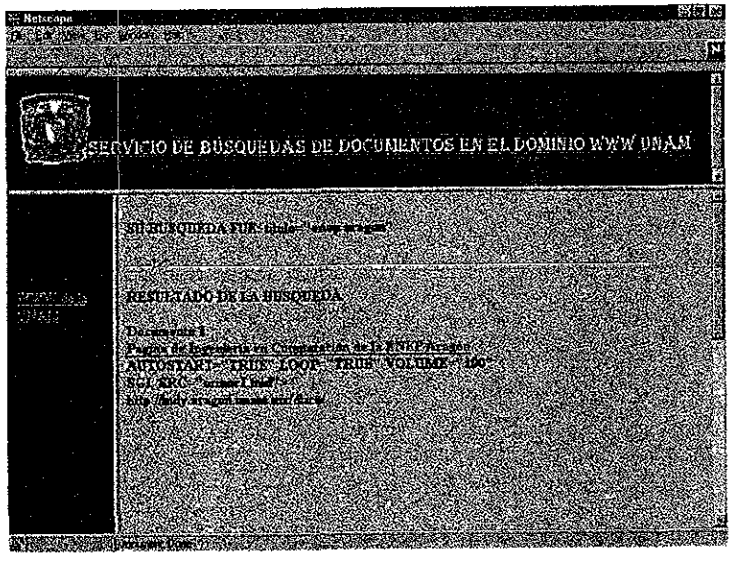


Figura 4 8 Resultados de la cadena de búsquedas enep aragón en el título del documento

Finalmente, en la figura 4.9 y 4 10, podemos observar, respectivamente la petición de búsquedas y los resultados presentados por el sistema para la búsqueda realizada solamente en el url del documento enviando la cadena de búsquedas aragon.unam

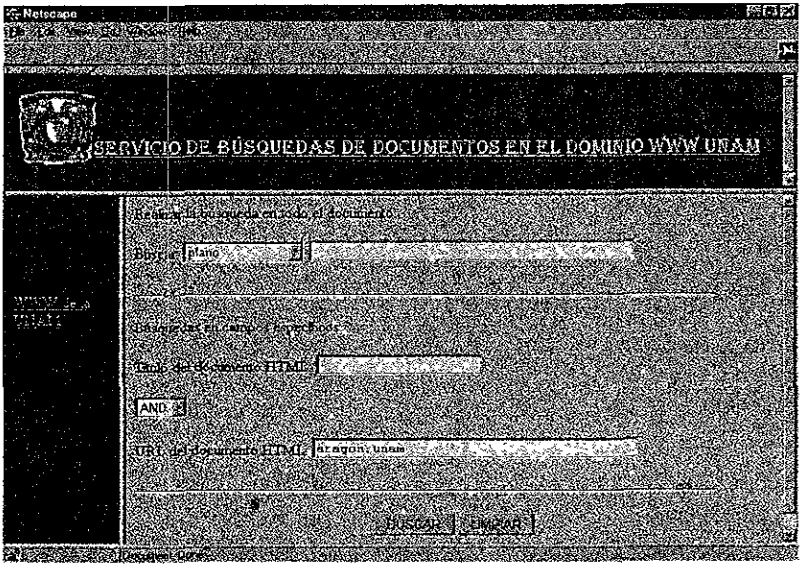
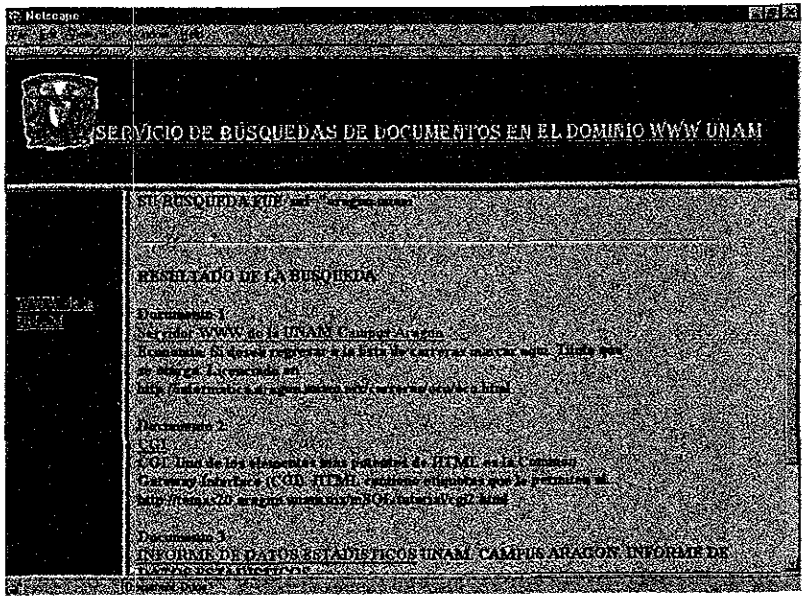


Figura 4 9 Petición de la cadena de búsquedas aragon.unam en el url del documento

Figura 4.10 Resultados de la cadena de búsquedas aragon.unam en el url del documento



Capítulo 5
Ejecución de pruebas y refinamiento del servicio

5.1 Comparación de los servicios de búsqueda de Altavista, Infoseek y el sistema propio.

Para poder tener un punto de comparación del servicio de búsquedas del presente trabajo, se instalaron temporalmente los sistemas que las compañías Infoseek y Altavista tienen en el mercado, utilizando las versiones libres, que se pueden tener a prueba por 1 mes, con ciertas restricciones, como la de Ultraseek Server, que solamente permite indexar 100,000 documentos.

Con lo que respecta al precio de los 2 productos, el costo del producto Ultraseek Server de la compañía Infoseek es de $35,000 para 100,000 documentos más 15% para servicio y actualizaciones. La compañía Altavista cotiza su producto con licencia ilimitada de Altavista Search en $103, 995.00.

Instalación y configuración de los servicios.

En los 2 sistemas se especificó como raíz para el proceso de recorrido del robot el sitio WWW de la UNAM http //www unam mx, al igual que la configuración del sistema propio.

En el sistema de Altavista se obtuvieron un total de 117,232 páginas indexadas, en el servicio de Infoseek 100,000 por las restricciones de licencia mencionadas con anterioridad, y en el servicio propio 1,568,700

En la tabla 5.1 se muestra una tabla comparativa del espacio ocupado por cada documento indexado y en la figura 5.1 se muestra la gráfica de barras correspondiente; en la tabla 5 2 se puede ver una comparación de la velocidad de indexado, y en la figura 5 2 la gráfica correspondiente.

Sistema de Búsquedas	Espacio en disco necesario
Altavista Search Intranet Extensions 97	5.32 Kb/documento
Ultraseek Server	6.90 Kb/documento
Servicio Propio	4.32 Kb/documento

Tabla 5.1 Comparación del espacio en disco por documento

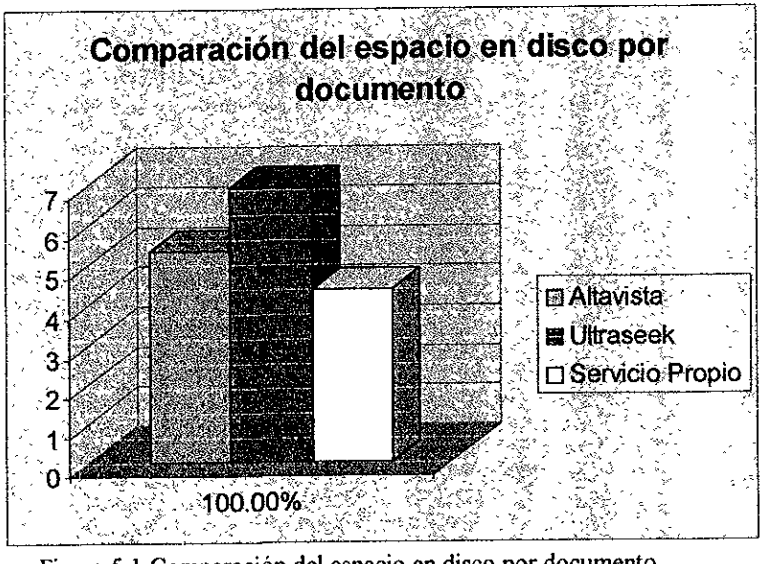


Figura 5.1 Comparación del espacio en disco por documento

Sistema de Búsquedas	Velocidad de Indexado
Altavista Search Intranet Extensions 97	1,600 documentos/hora
Ultraseek Server	2,800 documentos/hora
Servicio Propio	3,500 documentos/hora

Tabla 5 2 Comparación de la Velocidad de Indexado

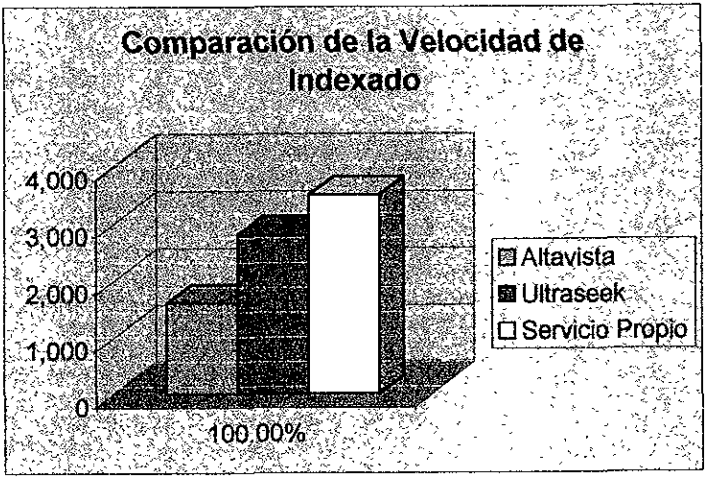


Figura 5.2 Comparación de la Velocidad de Indexado

La instalación del producto Ultraseek server es más sencilla y rápida, la instalación del producto de Altavista sobre la plataforma Sun Solaris 2.5 1 requiere de la instalación del parche para la librería libresolv.so.2, lo que hace que el proceso de instalación ocupe más tiempo.

La creación de la interfaz para el usuario es más complicada en el software de Altavista debido a que se deben modificar alrededor de 10 páginas html, en contraste a las 2 páginas requeridas en Ultraseek Server

La instalación y creación de interfaz al usuario en el servicio propio, está en función del número de computadoras que van a formar parte de la máquina virtual, y como se especifico en el capítulo 4, este proceso consta de varios pasos que se encuentran relacionados entre sí, iniciando en la configuración de la máquina virtual, hasta llegar a las pantallas de consultas.

Para evaluar la parte de las consultas en los 3 sistemas, se realizaron varias búsquedas simples y avanzadas, los resultados obtenidos mostraron que la velocidad para presentar los resultados es ligeramente mayor en el sistema propio que en los otros 2 sistemas evaluados, el hecho de que la velocidad de respuesta en el servicio propio comparada con los sistemas de Infoseek y de Altavista no sea lo suficientemente alta, es debido a que la red en la que se encuentran las computadoras que componen la máquina virtual es de tipo Ethernet a una velocidad de 10Mbit/s, otro de los factores que influyen para que esto suceda, es que las computadoras de la máquina virtual no están dedicadas al servicio de búsquedas y comparten el segmento de red con máquinas de producción que utilizan ancho de banda.

5.2 Situaciones en las que el sistema no podrá ser utilizado

Debido a que una de las partes que compone el sistema de búsquedas es freeWAIS-sf, el cual, como se mencionó en el capítulo 4, crea sus índices a partir de acervos de texto plano, que tienen una estructura de campos definida, el sistema no se puede utilizar para indexar documentos con formato binario como pdf, office, postscript, etc.

Otra de las restricciones del sistema de búsquedas propio, es que si uno de los nodos que constituyen la máquina virtual se pierde por alguna razón, la parte del acervo que se asignó al nodo en cuestión, no podrá ser utilizada en las búsquedas, hasta que el nodo se vuelva a incorporar.

Capítulo 6
Perspectivas de Desarrollo

El sistema está constituido de diversas partes, mostradas a lo largo del presente trabajo, las cuales se presentan en la figura 6.1

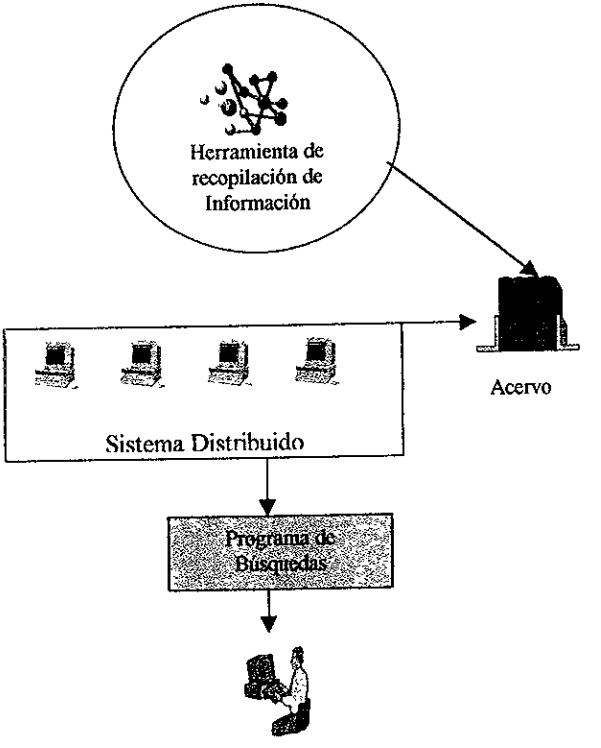


Figura 6.1 Elementos que constituyen el sistema de búsquedas

Como consecuencia de la organización y del diseño del sistema, es posible hacer uso de éste en diversas aplicaciones, asimismo es posible mejorar el sistema en diversas áreas, éstos temas son presentados en el presente capítulo.

6.1 Mejoras en la herramienta de recopilación de información

La herramienta de recopilación de información elegida para el sistema MOMSpider, tiene la característica de poder recuperar solamente documentos con formato de texto plano, en nuestro caso páginas HTML, para incrementar la funcionalidad y utilidad del sistema de búsquedas es importante que la herramienta de recopilación de información, tenga capacidad de recuperar archivos en diferentes formatos, como por ejemplo imágenes, documentos con formatos de Office como son word, excel, o power point.

De esta manera se le puede ofrecer al usuario una variedad más amplia de acervos en los cuales puede localizar la información que requiere

Por otra parte, como se presentó en el capítulo 4, el proceso de recorrido de la estructura de información de los sitios de la UNAM, se programó para realizarse los fines de semana, periodos de tiempo en los que no existe una gran demanda de servicios de Internet por parte de la comunidad universitaria, sin embargo esta parte del sistema puede ser mejorada, si se verifica que sitios de Web de la UNAM cambian con más frecuencia para recuperar solamente esa parte, ya que recorrer la red frecuentemente es ineficiente cuando solo unos cuantos documentos cambian.

Finalmente para ahorrar recursos de red como ancho de banda cuando se debe transferir una cantidad considerable de archivos de los sitios remotos al servidor local, sería importante, definir un formato para transferir colecciones de paquetes de documentos o documentos compuestos, una herramienta que se puede utilizar para este objetivo es el programa tar del sistema operativo Unix.

6.2 Mejoras en el programa de búsquedas

Con objeto de ofrecer mayores opciones al usuario para la definición de las cadenas de búsqueda, se pueden agregar características adicionales al sistema en el módulo correspondiente al programa de búsquedas, algunas de ellas pueden ser las siguientes:

- Capacidad de entender el lenguaje natural del usuario, por ejemplo la cadena de búsquedas ¿qué carreras existen en la UNAM?
- Integración de frases comunes del lenguaje.
- Capacidad de reconocer nombres propios

6.3 Personalización del sistema de búsquedas

Las mejoras presentadas en este punto son tomadas en su mayoría de la revisión de los productos comerciales de Sistemas de búsquedas que existen en el mercado, que son adicionales a las características propias de sistema de búsquedas, pero que dan al usuario un mejor manejo de información.

Una de las propuestas es la integración de un sistema de traducción, en vista de que la comunidad universitaria necesita consultar fuentes de información en diferentes idiomas, lo que ayudará a que la comunidad no esté limitada a la información que pueda obtener de fuentes escritas en el idioma español, sino que pueda hacer uso de los documentos que se encuentran distribuidos en Internet en diferentes idiomas, ampliando así sus fuentes de información.

Otra propuesta es la integración de información organizada a través de un árbol de categorías similar al presentado por la herramienta de búsquedas Yahoo, con categorías relativas a información importante de la UNAM, tal como

- Información acerca de la universidad
- Noticias e información en general
- Administración
- Educación e investigación
- Información a los alumnos
- Actividades y recreación
- Cómputo y comunicaciones

Para que de esta manera se pueda dar al usuario la facilidad de realizar búsquedas a través de la navegación en información organizada por temas

Como consecuencia del tema anterior, surge la propuesta de la integración de tecnología de empuje en el sistema, la razón para la integración de esta característica, es que actualmente los usuarios no se enteran de sucesos recientes si no visitan los sitios donde estos suceden, y por medio de esta tecnología pueden obtener la información más reciente de su interés, sin necesidad de visitar los sitios

Para lograr lo anterior, se puede hacer un programa integrado al sistema de búsquedas, por medio del cual los usuarios se inscriban a diversas categorías de su interés, en donde la entrega de la información nueva o más reciente se realice a través de correo electrónico, directamente a la cuenta del usuario.

6.4 Diversos usos del sistema, por medio de la sustitución del acervo

El presente proyecto surgió como respuesta a la problemática que representan las búsquedas de información en Internet para el usuario, por tal motivo, el acervo sobre el cual se realizan las búsquedas consiste de páginas de HTML con sus respectivas direcciones de Internet.

Sin embargo, una característica importante en el sistema es su flexibilidad, ya que se logró conformar un esquema base de trabajo, en el que con algunas modificaciones, es posible sustituir el acervo de información, con lo que el sistema resulta útil para realizar búsquedas de información de diversa índole.

Algo de gran utilidad sería la aplicación del sistema a las historias académicas de los alumnos de la UNAM, en donde el acervo de información podría encontrarse almacenado en CD-ROM, o en otro medio de solo lectura, y con modificaciones a los módulos que constituyen el sistema, pero basándonos en el concepto que presenta PVM en conjunción con el sistema de búsquedas, se puede obtener un sistema productivo y económicamente viable

Capítulo 7
Conclusiones

Al término del presente trabajo, se ha logrado la implementación, instalación y puesta en marcha de un sistema de búsquedas, constituido por la integración de varias herramientas independientes de software interactuando entre sí

Como se mencionó en el capítulo 3, existen en la actualidad, varios productos comerciales para realizar búsquedas, que presentan características adicionales al trabajo presentado como proyecto de tesis, sin embargo, su adquisición e implantación resulta costosa.

Como podemos verificar en el capítulo 5, su precio se encuentra arriba de $30,000, sin tomar en consideración el equipo necesario para recuperar, procesar y poner disponible la información. En la implantación del servicio de búsquedas expuesto en el presente trabajo, los costos son muy bajos, en primer lugar, gracias al uso conjunto de los recursos de hardware con que ya contaba la dependencia, que nos proporcionaron una capacidad de procesamiento suficiente para un buen rendimiento del sistema de búsquedas, y en segundo lugar, debido al uso de herramientas de software libre.

La velocidad de respuesta al usuario, lograda en el sistema de búsquedas, en general es aceptable, tomando en cuenta la velocidad de la red de comunicaciones en la que se encuentran las computadoras que forman parte de la máquina virtual, problemas que ya han sido expuestos en el capítulo 5, por lo que se sugiere que, de ser posible, en posteriores implantaciones del sistema presentado en este trabajo, se utilice un segmento de red de alta velocidad dedicado únicamente a la maquina virtual.

Por otro lado, algo importante que hay que mencionar, es que, uno de los factores más importantes que determinan el éxito de un sistema de búsquedas, se encuentra directamente relacionado con las entradas que el sistema recibe, las cuales dependen completamente de los usuarios, y del conocimiento que éstos tengan acerca de la forma de estructurar la cadena de búsquedas en un sistema en particular, por lo que, para poder hacer un uso eficiente de la herramienta de búsquedas desarrollada en el presente trabajo, antes, es necesario conocer las opciones que podemos utilizar, para que de esta manera podamos obtener resultados útiles y relevantes

Apéndice I
Sitios de web en el dominio WWW de la UNAM

www.unam.mx	www.math.unam.mx	ariel igeofcu.unam.mx
serpiente dgsca.unam.mx	www..unam.mx	graficas.matcuer unam mx
www dgae.unam.mx	hardy.fciencias.unam.mx	fax.ssn unam.mx
galois.dgae.unam.mx	uxmcc1.iimas.unam.mx	copan.cifn.unam mx
dgenp.unam.mx	pumas iingen unam.mx	www.pumas.iingen.unam.mx
www facmed.unam.mx	dgenp.dgenp.unam mx	khalo.fi-p unam.mx
hunabku.pquim.unam mx	www cuautitlan2 unam.mx	dgep.posgrado.unam.mx
info.juridicas.unam.mx	www pdcb.unam.mx	www.medinfo unam mx
cuib.laborales.unam.mx	pdcb.biomedicas.unam.mx	medinfo.fmedic.unam.mx
www estadistica.unam.mx	gondor.fi-c.unam.mx	www igeograf.unam.mx
www.dgbiblio.unam.mx	cdm1.fi-c.unam.mx	www.cad.fi-p.unam.mx
www.museovirtual.unam.mx	frida.fi-p.unam.mx	indy2.igeograf.unam.mx
www fciencias.unam.mx	uxmym1.iimas.unam.mx	larubi.cnb unam.mx
server.contad.unam.mx	bq.unam.mx	www acatlan.unam mx
www.derecho.unam.mx	www.iquimica.unam mx	geologia.igeolcu.unam.mx
www.veterin.unam.mx	lince.dgsca.unam mx	indiana.acatlan.unam.mx
www.ibt.unam.mx	catalisis.fmedic.unam.mx	www.ssn.unam.mx
www nuclecu.unam.mx	www.biomedicas unam.mx	jaguarundi.aragon unam.mx
fenix.ifisicacu.unam.mx	www cecafi.unam.mx	www.multi.com uy
arrakis.iimatercu unam.mx	www.fi-b unam.mx	sorjuana.dgsca unam mx
www matem.unam.mx	cosmeg.fi-a unam.mx	cifn.unam.mx
www cifn.unam.mx	cleopatra fi-a.unam.mx	bibliounam unam mx
tzetzal.dcaa.unam.mx	iris.ifisicacu unam mx	franco astroscu.unam.mx
info1.juridicas unam mx	laguna fmedic.unam.mx	irixcicc cuautitlan2.unam.mx
sunsite unam.mx	iqunam.iquimica.unam mx	apolo acatlan unam.mx
www.dtd unam.mx	lince.dcaa unam.mx	viper.acatlan.unam mx
www.nic.unam.mx	www.astroscu.unam mx	bachiller.dgsca unam.mx
www.noc.unam.mx	hussongs.astrosen unam.mx	osuno.fciencias.unam.mx
miztli.cchadm.unam.mx	dragon.dgsca.unam.mx	sua.duad unam.mx
uiip.posgrado.unam.mx	soledad.astroscu.unam.mx	www zaragoza unam mx
castor estadistica.unam.mx	www.astrosmo.unam.mx	pacific.fi-p.unam.mx
deneb labvis.unam.mx	genesis astrosmo unam mx	drbaz fmedic.unam.mx

hp.fciencias.unam.mx	www.super unam mx	ifcsun2.ifisol.unam.mx
themis.derecho.unam.mx	www mcc.unam.mx	www.iimtemix unam.mx
cuauhtli.veterin.unam.mx	genesis astroscu.unam.mx	www.crim unam mx
pbr322.ceingebi unam.mx	caifan.astroscu.unam.mx	www.filosoficas.unam.mx
luthien.nuclecu.unam.mx	mezcal.super.unam.mx	docencia.dgsca.unam.mx
calli.matem.unam mx	verona.fi-p.unam.mx	exodus.dcaa.unam mx
uxmal cifn.unam.mx	www.iingen.unam.mx	pompeya.cise-sua unam.mx
sunsite.dcaa.unam.mx	venus.ifisicacu.unam.mx	unamsi0.dgire.unam.mx
cuk.redes.unam.mx	www.ifisican.unam.mx	tehueque.unacc.unam.mx
akash.nic.unam.mx	www.ifisicaen.unam.mx	aristoteles.dgnsa.unam.mx
antares.noc unam.mx	chicoce.ifisicam.unam.mx	dgprov.proveed.unam.mx
calli.fciencias.unam.mx	gigante.ifisicaen.unam.mx	abogado.rectoria.unam mx
ludwig.dgsca.unam.mx	www mathmoo.unam.mx	www.ddu.unam.mx
www.matcuer.unam.mx	cad.fi-p.unam.mx	delfin.zaragoza.unam.mx
iocd unam.mx	khalo.depfi.unam.mx	xiuhcoatl.iimtemix unam.mx
eros.pquim unam mx	athena.fciencias.unam.mx	main.crim.unam.mx
www.fca.unam.mx	tlaloc.dgapa.unam mx	www.ibiologia unam.mx
guevera.matcuer.unam.mx	informatica.aragon.unam.mx	bucanero.redes.unam.mx
organica1.pquim.unam.mx	ce-atl.posgrado.unam.mx	minerva.filosoficas.unam.mx
www.juridicas.unam.mx	cad.depfi.unam.mx	pompeya cuaed.unam mx
uxdea1.iimas.unam.mx	dctrl.fidieec.unam.mx	cakes.dgasc.unam mx
xenon pquim.unam.mx	tlacaelel.igeofcu.unam.mx	ibunam ibiologia.unam.mx
www.cchazc.unam.mx	pacific.depfi.unam.mx	bucanero.tac.unam.mx
www.cch-vallejo.unam.mx	sociolan.politicas.unam.mx	infocuib laborales.unam mx
www.ccchs.unam.mx	troyadii.fi-p.unam.mx	cuib.unam.mx
www.ifisiol.unam.mx	granta fciencias unam.mx	fenix.cichcu unam.mx
biblioweb.dgsca.unam.mx	ds5000.super.unam.mx	graf.fciencias.unam.mx
marcopolo dgsca.unam.mx	sauron.dgsca.unam.mx	gopher acatlan.unam.mx
hermes.mascarones.unam.mx	gauss.matem.unam.mx	wwwfacmed.unam.mx
www.iztacala.unam.mx	indy aragon.unam mx	cd-atl.posgrado.unam.mx
www.cuautitlan.unam mx	pyros.igeofcu.unam.mx	hunabbu pqui.unam.mx
www.mineria unam.mx	www.igeofcu unam mx	dgep.posgrado.unam.mx.unam.mx
www.universum.unam mx	nundehui.igeofcu.unam mx	dgep.unam.mx
www.red-mat.unam.mx	biblional.bibliog.unam.mx	audiovi2.fciencias.unam.mx
odin fi-b.unam.mx	www.fi-p.unam mx	audiovisual fciencias.unam.mx
www.astrosen.unam.mx	pumas.fi-a.unam.mx	dgasc01.dgasc.unam.mx

condor.dgsca.unam.mx	anthrax.fi-a.unam.mx	www.dcn davis.ca us
coor.ccchs.unam.mx	aleph cinstrum.unam.mx	wheel dcn.davis ca.us
ifcsun2.ifisiol.unam.mx	pix.nuclecu.unam.mx	ola.icmyl.unam.mx
campus.iztacala.unam.mx	www.neclecu unam.mx	www.mexacceldb.unam.mx
tolsa mineria.unam.mx	dgnep.unam.mx	www.radiounam unam.mx
barajas.fciencias.unam.mx	uxestad1 iimas.unam.mx	uno.enap unam.mx
bufadora.astrosen.unam.mx	sigma.iimas unam.mx	leaac.dgsca.unam mx
galois.dgaesc.unam.mx	charro.igeofcu.unam.mx	buho.economia.unam.mx
biblios.iztacala unam.mx	mezcal.dgsca.unam.mx	terascan.igeograf.unam.mx
www.fi-a unam.mx	ds5000.dgsca.unam.mx	atlantis.igeograf.unam.mx
uxcomp2.iimas.unam.mx	miro.ifisicacu.unam.mx	sepiente.dgsca.unam mx
posgrado.psicol.unam.mx	www cenapred.unam.mx	zafiro.dgsca.unam mx
www.labvis.unam.mx	ensayes-1.cenapred.unam.mx	noc.noc.unam.mx
soc.ifisiol.unam.mx	kokiro.iingen.unam.mx	exodus.dgsca.unam mx
vd4.ifisiol.unam.mx	dctrl.fi-b.unam.mx	
smf1.fciencias.unam mx	astroscu.unam.mx	
cozumel.fi-a.unam.mx	www.javis.unam.mx	
uxmcc2.iimas.unam.mx	sismo1.ssn.unam.mx	

Fuente de información: Estadísticas generadas por el programa de recopilación de información "scooter" del producto Altavista Search IntraNet Extensions

Apéndice II
Directivas de MOMSpider

Directivas globales

SystemAvoid ruta

Especifica que el archivo de sitios no permitidos para este proceso puede ser encontrado en la ruta dada.

SystemSites ruta

Especifica que el archivo de sitios para este proceso puede ser encontrado en la ruta dada.

AvoidFile ruta

Especifica que el archivo de sitios no permitidos del usuario para este proceso puede ser encontrada en la ruta.

SiteFile ruta

Especifica que el archivo de sitios del usuario puede ser encontrada en la ruta.

SitesCheck N

Especifica el número de días entre verificaciones del archivo ./robots.txt de un sitio. El default es usualmente 15 días.

ReplyTo dirección de correo electrónico

Especifica la dirección de correo electrónico de la persona que está corriendo el programa

MaxDepth N

Especifica la profundidad máxima permitida para el proceso de recorrido de MOMSpider.

Directivas de tareas

<site

Indica el inicio de una instrucción de tarea para el recorrido por sitio

<tree

Indica el inicio de una instrucción de tarea para el recorrido por árbol.

<owner

Indica el inicio de una instrucción de tarea para el recorrido por dueño.

Name nombre_del_sitio

Especifica el nombre de la estructura de información. Es utilizado para identificar la estructura de información en los mensajes generados. El nombre es requerido para todas las tareas y debe consistir de una sola palabra

TopURL URL

Especifica el URL raíz de la estructura de información a ser recorrida.

IndexURL URL

Especifica el URL del archivo índice HTML que será producido por esta tarea.

IndexFile ruta html

Especifica la ruta del archivo real para el índice HTML.

IndexTitle cadena

Especifica la cadena que se va a usar como título del índice HTML y como tema en el mensaje de correo

ChangeWindow N

Especifica la ventana en N días anteriores a la fecha actual dentro de la cual la fecha de última modificación del URL evaluado se va a considerar interesante y debe ser destacada en el índice HTML.

ExpireWindow N

Especifica la ventana en N días después de la fecha actual dentro de la cual la fecha de expiración de los URL's recorridos es considerada interesante y debe ser destacada en el índice HTML.

Exclude prefijoURL

Especifica que el prefijo dado debe ser añadido a la tabla de nodos hojas, es decir que los URLs que contengan éstos prefijos solo serán evaluados pero no recorridos.

Apéndice III
Archivos de configuración de MOMSpider utilizados en el sistema

Archivo de Instrucciones

```
# MOMspider-0.1a Archivo de Instrucciones

SystemAvoid /home/servicio/MOMSpider-1.01/.momspider-avoid
SystemSites /home/servicio/MOMSpider-1 01/.momspider.sites
SitesCheck  7
ReplyTo edithv@servidor.unam mx
MaxDepth 20
<Site
   Name         UNAM
   TopURL        http://www.unam.mx
   IndexFile     /home/servicio/httpd/htdocs/MOM/UNAM.html
   IndexTitle    Indice de MOMSpider para los sitios de la UNAM
   EmailAddress  edith@servidor.unam.mx
>
```

Archivo de Sitios no permitidos

```
# MOMspider-0.1a Avoid File:  Lists URL prefixes to avoid or leaf.
# New URL prefixes can be added when the program is not running.
# The file format is:  EntryType  URLprefix  [ExpireDate]
# where EntryType  = "Avoid" or "Leaf"
#       URLprefix  = the full URL prefix for which this entry applies
#       ExpireDate =  [*] for never expire
#                  or [date] (see wwwdates.pl for valid date formats)
# This file is automatically generated, so don't bother changing the format.

Avoid http://www.ncsa.uiuc.edu:8001/ [*]
Avoid http://www.w3.org:8001/ [*]
Avoid http://www.contrib.andrew.cmu.edu:8001/sokoban/ [*]
Leaf  http://www.ncsa.uiuc.edu/SDG/Software/Mosaic/Docs/whats-new.html [*]
Leaf  http://www.ncsa.uiuc.edu/SDG/Software/Mosaic/MetaIndex.html [*]
Leaf  http://www.w3.org/hypertext/DataSources/ByAccess.html [*]
Leaf  http://www.w3.org/hypertext/DataSources/bySubject/Overview.html [*]
Avoid http://www.whitehouse.gov/ [*]
Leaf  http://www.ics.uci.edu/Admin/ [*]
Avoid http://www.ics.uci.edu/Test/momtest.html [Sat, 02 Apr 1994 00:00:00 GMT]
Avoid http //betelgeuse.com/name.html [Fri, 31 Dec 1999 09:30:00 GMT]
Avoid http://simplon.ics.uci.edu:8001/ [Thu, 29 Sep 1994 23:59.59 GMT]
```

Archivo de Sitios

```
# MOMspider-0.1a Sites File:  Lists IPaddress:port locations we've checked
# for a /RobotsNotWanted.txt file and followed its directions.
# New sites can be added when the program is not running.
# The file format is:  EntryType  IPaddress:Port  [ExpireDate]
# where EntryType  = "Site"
#       IPaddress  = the full hostname or IP address for the site
#       Port       = the numeric TCP port for the site (write 80 for default)
#       ExpireDate =  [*] for never expire (i.e. never check this site)
#                  or [date] (see wwwdates.pl for valid date formats)
#                     Entries are automatically cleared after 7 days
# This file is automatically generated, so don't bother changing the format.

Site www.unam.mx:80 [*]
Site serpiente.dgsca.unam.mx:80 [*]
```

Apéndice IV
Opciones del programa waisindex

Las siguientes son las opciones que presenta el programa waisindex al momento de indexar los documentos.

-d nombre_índice

Este es el nombre de archivo base para los archivos de índices Por lo tanto si se especifica /usr/local/base, los índices se van a llamar /usr/local/bases.dct, etc. El índice se debe almacenar en el sistema de archivos local de la maquina corriendo el waisindex

-a

Agrega este índice a un índice existente.

-r

Indexado recursivo en subdirectorios

-mem

Cantidad de memoria principal que se va a utilizar durante el indexado.

-export

Esto provoca que el archivo fuente de descripción resultante incluya el nombre de la máquina, y el puerto tcp, para ser usado por los clientes De otra forma el archivo no contiene información de conexión, y se espera que sea usado solo para búsquedas locales

-nocat

Inhibe la creación de un catalogo. Esta opción es funcional para bases de datos con un gran numero de documentos, porque el catálogo contiene 3 líneas por documento

-T tipo

Pone el tipo del documento al especificado por "tipo"

-t formato

Este es el formato de los archivos que son manejados por waisindex. Utilizar `-t campos' para usar el indexado por campos. El archivo `basedatos.fmt' debe contener la descripción del formato.

-stop

Esta opción se refiere a la lista de las palabras que serán excluidas del indexado y por lo tanto de las búsquedas. Si se da -stop la lista de palabras previamente construida no es utilizada. En otro caso el archivo `basedatos.stop' debe contener las palabras separadas por saltos de líneas

Nombre_archivo

Estos son los archivos que serán indexados acorde a los argumentos mencionados anteriormente. Para asegurar que los archivos se agregarán correctamente a la lista es recomendable utilizar rutas completas.

-stdin
Lee los nombres de los archivos a indexar de la entrada estándar en vez de la línea de comandos.

Apéndice V
Formato de descripción del programa waisindex

Una cosa que es importante saber cuando se escribe un formato de descripción es que waisindex revisa los archivos fuente línea por línea. Todas las expresiones regulares tienen que igualar una línea única.

Expresiones regulares

Todas las expresiones regulares en el formato de descripción deben ser incluidas entre 2 `/'. Por lo tanto un `/' en la expresión regular debe de protegerse con `\' Otras secuencias de escape interpretadas son·

\n Nueva línea
\r El carácter retorno de carro
\x Control x donde x va desde `A' a `Z'

Operador	
x	El carácter `x'
"x"	Una `x' aún si x es un operador
\x	Una `x' aun si x es un operador. Recordar que el parser se come un `\'. Así que debemos usar`/\\*/' para igualar un `*' en el texto.
[xy]	El carácter `x' o `y'
[a-c]	Los caracteres `a', `b' o `c'
[^x]	Cualquier carácter excepto `x'
.	Cualquier carácter excepto nueva línea
^x	Una `x' al principio de la línea
x$	Una `x' al final de la línea
x?	Una `x' opcional
x*	0,1,2,3 ... instancias de `x'
x+	1,2,3,4 ... instancias de `x'
x|y	Una `x' o una `y'
(x)	Una `x'
x{m,n}	m a través de n ocurrencias de `x'

Estructura del formato de descripciones

Un formato de descripción consiste en 3 partes. La primera parte (usualmente una sola línea) describe como se deben separar los registros en los documentos. La segunda parte, llamada la sección de distribución define como debe computarse la cabecera de línea para cada documento. La ultima y usualmente la más larga parte define que campos deben ser generados y como debe de hacerse el indexado

Un formato de descripción empieza con la directiva `<registro_fin>'

1 Sección de separador de documentos

Record-sep. regexp
Aquí regexp puede ser `/^\n/' para igualar una línea conteniendo un carácter de nueva línea. La línea que contiene el separador de registros nunca es indexada.

2. Sección de Layout

La sección de layout debe estar entre `layout.' y `end.'. Puede contener múltiples llaves `headline:' y una directiva `date:' Cada llave `headline ' define una región en el documento la cual debe ser copiada a una sección en la cabecera de línea. La región, definida por la expresión regular que la inicia y la termina puede aparecer muchas veces en el documento El orden de las directivas define el orden de las secciones en la cabecera de línea.

headline: start end width [skip]

La línea de anterior indica al indexador que copie el texto que se encuentra entre "start" y "end", opcionalmente saltando el texto igualado por "skip" a los siguientes "width" caracteres de la cabecera de línea.

Las directivas `date:' definen como puede ser extraída una fecha que va a ser asociada con el documento. Debemos notar que esta fecha no es utilizable para las búsquedas Es parte de la cabecera de línea y es desplegada al usuario por medio de algún cliente.

date: start scanfarg d-m-y d-m-y d-m-y skip

El "start" y "skip" trabajan como se mencionó anteriormente. El parámetro scanfarg se pasa a scanf para leer en año, mes y día El orden de los argumentos es indicado por d-m-y donde cada uno de ellos significa año, mes y día. El mes debe ser seguido por `string' para indicar que el mes va a ser representado por la abreviación estándar de 3 letras

A continuación se presenta un ejemplo:

```
layout:
headline: /^PY: / /^[A-Z][A-Z]:/ 5 /^PY: */
headline: /^AU: / /^[A-Z][A-Z]./ 21 /^AU: */
headline: /^TI· / /^[A-Z][A-Z]:/ 41 /^TI: */
date. /^ED: / /%d-%3s-%d/ day month string year /^ED: [^ ]/
end:
```

3. Sección de definición de campos y de tipos de índices

La parte de la especificación de campos del formato de descripciones esta compuesto de múltiples grupos `region:' `end:'

Cada uno mapea una región de texto a un conjunto de categorías de búsqueda por campos.

```
region: start [skip]
  fieldlist options indexspecs
end: end
```

Las expresiones start, skip y end definen la región del documento para la cual aplica la especificación del índice.
`fieldlist' es la lista de archivos (son los nombres de los índices) cada uno opcionalmente seguido por una cadena de descripción que se encuentra entre comillas La descripción se introduce en la base de datos de descripción.

Las opciones incluyen las directivas `numeric skip width', `date: d-m-y d-m-y d-m-y', y `stemming'

La primera opción le dice al indexador que permita solo valores numéricos y asegura que la búsqueda atómica de expresiones numéricas trabajen con las categorías. La segunda le dice al indexador que convierta la fecha en la región al formato `yymmdd' antes de indexarla numéricamente. La ultima opción le dice al indexador que contenga las palabras en la región antes de que ponerlas en el índice Cuando se hacen las búsquedas, el servidor también contiene términos a buscar antes de ver en el índice.

`indexspecs' es una lista de tipos de índices junto con una palabra clave indicando si la región debe ser mapeada a las categorías designadas (`LOCAL'), a la categoría por default (`GLOBAL') o a ambas (`BOTH'). La categoría por default es usada cuando no se especifica alguna en la cadena de búsquedas

Los tipos de índices soportados actualmente son `TEXT', `SOUNDEX' y `PHONIX'.

Consideremos el siguiente ejemplo:

```
region: /^AU: /
      au "nombres de autores" SOUNDEX LOCAL TEXT BOTH
end. /^[A-Z][A-Z]:/
```

Para el indexador esto significa·

Para todas las palabras que comienzan con 'AU: ' al inicio de la línea hasta otra línea que comience con 2 letras mayúsculas seguidas por dos puntos y un carácter en blanco, pon la palabra en el default y en la categoría `au' y el código soundex solo se pone en la categoría `au'.

Con lo anterior un nombre de autor puede ser encontrado en la base creada en la categoría por default o en la categoría `au' si se sabe exactamente la ortografía del nombre. Si no se sabe bien la ortografía del nombre, este puede ser encontrado usando la consulta· `au=(soundex misspelled-name)'

Apéndice VI
Descripción de archivos utilizados y generados por waisindex

El programa waisindex utiliza y genera varios archivos, al momento de ser invocado para la generación de los índices.

Por ejemplo si el comando es invocado de la siguiente manera:
waisindex -d test
Donde test es el nombre del índice a ser generado.

Utiliza los archivos:

test fmt	La definición del formato.
test.fde	Opcional, formato de descripciones, texto plano el cual es adicionado a la descripción de la base de datos.
test.syn	Opcional. Archivo de sinónimos, contiene múltiples líneas con sinónimos separados por espacios.
test.stop	Opcional Archivo que contiene palabras que deben ser ignoradas cuando se indexa, cada una en una línea.

Genera o modifica los archivos:

test.src	La descripción de la base de datos
test.fin	La lista de archivos. Una entrada por cada archivo en la base
test.hl	La lista de cabeceras de línea, una entrada por cada archivo en la base de datos
test.doc	Tabla de documentos. Una entrada por cada documento en la base, contiene punteros a la lista de nombres de archivos y a la lista de cabeceras de líneas.
test.cat	El archivo de catalogo. Una combinación de tablas de documentos con cabeceras de línea y una lista del nombre de los archivos entendible para los humanos.
test.dct	El diccionario global.
test.inv	EL archivo invertido para el campo por default
test stop	waisindex puede adicionar palabras al archivo de palabras que no se deban indexar, si ellas aparecen muchas veces.

Bibliografía

PVM: Parallel Virtual Machine, A Users' Guide and Tutorial for Networked Parallel Computing
Geist, A., Beguelin, A., Dongarra, J., Jiang, W., Manchek, R y Sunderam, V.
MIT Press, Cambridge,MA 1994

Computación Paralela en las Geociencias
Gómez Valdés, José
GEOS, Unión Geofísica Mexicana, Sep. 1996

Parallel Computing. Theory and Practice
Quinn, Michael J.
McGraw-Hill Inc 2nd Ed 1994

Handbook of Parallel Computation
Zomaya, Albert Y
McGraw-Hill Computer Engineering Series, 1996

IRIX Parallel Programming
Student Handbook
Silicon Graphics, Computer Systems.

Referencias en Internet

INFORMACION GENERAL

Dirección de Telecomunicaciones Digitales
http://www.dtd.unam.mx/index-home.html

WWW de la UNAM
http://www.unam.mx

Estadísicas
http://www.unam.mx/estadisticas

WWW

The World Wide Web Consortium
http://www.w3.org

WWW Intro -- Advantages of the WWW - Netscape
http://www.slis.ua.edu/wwwintro/wwwadv.htm

Why Use the World Wide Web?
http://cscsun1.larc.nasa.gov/~beowulf/db/why_use_web.html

Evaluation of Selected Internet Search Tools
http://www.library.nwu.edu/resources/internet/search/evaluate.html

Beyond Surfing Tools and Techniques for Searching the Web
http://magi.com/~mmelick/it96jan.htm

Understanding WWW Search Tools
http://www.indiana.edu/~librcsd/search/

Understanding and comparing web search tools
http://www.hamline.edu/library/bush/handouts/comparisons.html

HERRAMIENTA DE RECOPILACION DE INFORMACION

The Web Robots Pages
http //info webcrawler.com/mak/projects/robots/robots.html

MOMSpider
http://www.ics.uci.edu/pub/websoft/MOMspider/

MOMSpider Paper
http://www.ics.uci.edu/pub/websoft/MOMspider/WWW94/paper.html

COMPUTO PARALELO Y DISTRIBUIDO

Netlib
http //www.netlib.org/index.html

Cómputo Paralelo y distribuido
http //sundia.cem.itesm.mx/jesus/cursos/compd97/presenta.html

Supercomputadora Origin 2000
http://www.super.unam.mx/supercomputo/origin.html

MAF-Distributed Computing
http //www.maf.wisc.edu/distributed/condor/

PVM: Parallel Virtual Machine
http //www.epm.ornl.gov/pvm/

LAM/MPI Parallel Computing
http://www.osc.edu/lam.html

Condor Overview
http://www.maf.wisc.edu/distributed/condor/oview.html

The Condor System
http //www.bo.infn.it/ccl/condor/product.html

Condor Project Homepage
http://www.cs.wisc.edu/condor/

Scientific Home Page
http //www.sca.com/index.html

Yale Linda Group
http://www.cs.yale.edu/HTML/YALE/CS/Linda/linda.html

Piranha
http://www.cs.yale.edu/HTML/YALE/CS/Linda/piranha.html

NHSE ReviewTM 1996 Volume First Issue Cluster Management Software
http://nhse.cs.rice.edu/NHSEreview/CMS/

DataproClustering Technologies for Windows NT Servers
http://www.ncr.com/product/integrated/analyst_reports/1390-1.htm#TAG1

freeWAIS-sf

freeWAIS-sf Manual
http://amaunet.cs.uni-dortmund.de/projects/ir/freeWAIS-sf/fwsf_toc.html

freeWAIS-sf-faq
ftp://mirror-site/mirror-dir/freeWAIS-sf-faq
http://www.cis.ohio-state.edu/hypertext/faq/usenet/wais-faq/freeWAIS-sf/faq.html

WWW95 Paper
http://www.igd.fhg.de/www/www95/papers/47/fwsf/fwsf.html

freeWAIS-sf-2.0
ftp://mirror-site/mirror-dir/freeWAIS-sf-2.0/freeWAIS-sf-2.0.tar.gz