



UNIVERSIDAD NACIONAL AUTONOMA
DE MEXICO

FACULTAD DE INGENIERIA

SISTEMA DE CONSULTA DE INFORMACION EN EL
WEB DE INTERNET BAJO AMBIENTE DE BASES
DE DATOS

TESIS PROFESIONAL
QUE PARA OBTENER EL TITULO DE:
INGENIERO EN COMPUTACION
P R E S E N T A :
CLAUDIA ANGELICA CORDERO HIDALGO

DIRECTOR DE TESIS: ING. LAURA SANDOVAL MONTAÑO



CIUDAD UNIVERSITARIA,

1997

TESIS CON
FALLA DE ORIGEN



Universidad Nacional
Autónoma de México



UNAM – Dirección General de Bibliotecas
Tesis Digitales
Restricciones de uso

DERECHOS RESERVADOS ©
PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL

Todo el material contenido en esta tesis esta protegido por la Ley Federal del Derecho de Autor (LFDA) de los Estados Unidos Mexicanos (México).

El uso de imágenes, fragmentos de videos, y demás material que sea objeto de protección de los derechos de autor, será exclusivamente para fines educativos e informativos y deberá citar la fuente donde la obtuvo mencionando el autor o autores. Cualquier uso distinto como el lucro, reproducción, edición o modificación, será perseguido y sancionado por el respectivo titular de los Derechos de Autor.

Agradezco

A la *Universidad* por la valiosa formación que me brindó.

A la *Facultad de Ingeniería* por los conocimientos brindados para mi desarrollo profesional.

A la *Unidad de Servicios de Cómputo Académico antes (CECAFI)* por la gran oportunidad de ser parte de su equipo.

A la *Ing. Laura Sandoval Montaña* por su guía y gran apoyo en la elaboración de este proyecto.

A todos y cada uno de los *integrantes de UNICA* por su aportación al proyecto.

Doy gracias a Dios por permitirme llegar a la culminación de esta etapa de mi vida.

Dedico este trabajo a

Mis padres Juana María y Emilio con todo mi amor por su apoyo, su fe en mí, y por ser mi fuente de inspiración para luchar siempre por ser mejor persona.

A mis hermanos Josefina, Víctor, Rosa Ma., Edmundo y Antonio por su ejemplo, confianza, apoyo pero sobre todo por su amor, los amo.

A mis sobrinos Mar, Ari, Erika, Efrén, Katy, Elena y Chucho por su cariño y ternura.

A mis amigos que debo agradecer a la vida el poder decir que tengo muchos y grandes amigos:

A Martha, Wendy, Carmen y Betzabé, por su apoyo y cariño , gracias por estar siempre presentes.

A Laura por su guía, su apoyo, su confianza, pero sobre todo gracias por tu valiosa amistad.

A Oscar por tu apoyo incondicional y por el invaluable regalo de tu amistad.

A Alejandro, Edgar y Jorge, por ser siempre mis amigos.

A todo los chicos de UNICA por su cariño.

Claudia

	Página
Índice	
Introducción	i
Capítulo I Internet	
I.1.-¿Qué es Internet ?	1
I.2.- Estructura física de Internet	3
1.2.1.- TCP/IP bajo el modelo OSI	4
1.2.2.- Características del direccionamiento en Internet	7
1.2.3.- Dominios en Internet	9
I.3.- Protocolos que maneja Internet	12
Capítulo II El World Wide Web (WWW)	
II.1.- Antecedentes	16
II.1.1.- Sesión remota	16
II.1.2.- Transferencia de archivos	18
II.1.3.- Archie	20
II.1.4.- Correo electrónico	21
II.1.5.- Gopher	23
II.1.6.- Verónica	25
II.1.7.- WAIS	26
II.2.- Surgimiento del Word Wide Web	26
II.3.- Características y funcionamiento	28
II.3.1.- Mosaic	29
II.3.2.- Netscape	33

Capítulo III El Hipertexto

III.1.- ¿Qué es hipertexto?	40
III.1.1.- Funcionamiento de un modelo de Hipertexto	41
III.2.- Relación con lenguajes y protocolos	44
III.2.1.- Partes de un documento HTML (niveles 0 y 1)	45
III.3.- Creación de un Home Page	52

Capítulo IV El hipertexto bajo ambiente de bases de datos

IV.1.- Servidores de bases de datos	54
IV.2.- Servidores de hipertexto	55
IV.3.- Relación del hipertexto y bases de datos.	59

Capítulo V Desarrollo del sistema de consulta de planes de estudio de las carreras que ofrece la Facultad de Ingeniería en el Web de Internet.

V.1.- Análisis de sistema	63
V.1.1.- Objetivo General	63
V.1.2.- Antecedentes	63
V.1.3.- Planteamiento del problema	64
V.1.4.- Propuesta de solución	65
V.2.- Subsistema de información	69
V.2.1.- Análisis de requerimientos	69
V.2.2.- Diseño	70
V.2.3.- Implantación	83
V.2.4.- Mantenimiento	85

V.3.- Subsistema de presentación	90
V.3.1.- Análisis de requerimientos	90
V.3.2.- Diseño	91
V.3.3.- Implantación	97
V.3.4.- Mantenimiento	108
V.4.- Subsistema de enlace	109
V.4.1.- Análisis de requerimientos	109
V.4.2.- Diseño	112
V.4.3.- Implantación	117
V.4.4.- Mantenimiento	125
Conclusiones	126
Apéndice	
Glosario	
Bibliografía	

Introducción

Introducción

Internet es sin lugar a dudas la plataforma de comunicaciones con más futuro y más crecimiento; desde sus inicios ha causado gran expectación y ha provocado que todos quieran tener acceso a alguno de los servicios que proporciona. Pero, cuáles son estos servicios y por qué algunos son más utilizados que otros, estas son algunas de las preguntas planteadas para el desarrollo del presente trabajo de tesis.

Aunado a esto, el saber que podemos estar comunicados a cualquier parte del mundo y obtener información de cualquier cosa que se nos ocurra de forma confiable, requiere de un gran compromiso por parte de aquellos que proporcionan dicha información; ya que deben clasificarla y actualizarla vertiginosamente; esta labor no es nada sencilla, si no existe una forma directa de actualizar dicha información, es decir si no se cuenta con una estructura de almacenamiento con las características de bases de datos.

Por ello el "boom" de los manejadores de bases de datos o bien de las herramientas de desarrollo de aplicaciones al ofrecer una integración de sus sistemas de información al ambiente del Web para realizar operaciones de consulta y/o actualización de dicha información.

En el presente trabajo se abordarán temas relacionados con este fenómeno dentro de la informática, como lo son: el desarrollo de Internet y sus diferentes servicios, el hipertexto, su relación con las bases de datos; así como la puesta en marcha de un sistema que integra los puntos ya mencionados y cuyo objetivo es:

Crear un sistema de información útil a la comunidad de la Facultad de Ingeniería y que tenga proyección a través de la plataforma Internet.

Descripción del contenido de los capítulos

Capítulo I

En el capítulo uno, se hace una introducción a lo que es el mundo de Internet; su creación, desarrollo y sus alcances, así como sus características de operación desde la estructura de las redes que la conforman hasta la interconexión entre ellas, sus protocolos de comunicación y la transferencia de información.

Capítulo II

En este capítulo hacemos un recorrido por todos los servicios que se han implementado en Internet desde su puesta en marcha hasta la actualidad, por mencionar algunos: archie, gopher, wais, mosaic y netscape, estos dos últimos incluidos en el World Wide Web, del cual se mencionarán sus características y funcionamiento, su relación con el hipertexto y el home page y por qué este servicio es el más utilizado por los usuarios de Internet.

Capítulo III

El capítulo tres es un estudio sobre las características del hipertexto como un lenguaje de comunicaciones, como generador de home pages que son las pantallas de presentación por las que navegamos en el Web y cómo es que éstas se crean.

Capítulo IV

Se hace una descripción de los servidores de bases de datos y los servidores de hipertexto, cómo se crean y cómo se pueden relacionar, cómo actualizar la información de un home page de acuerdo a la información contenida en una base de datos.

Capítulo V

Se plantea el desarrollo del sistema de consultas de los planes de estudio de las carreras que ofrece la facultad y su integración al home page de la facultad, a través de una base de datos creada en Oracle y su inclusión a través de hipertexto.

Capítulo I

Capítulo I Internet

Introducción

El crecimiento en las comunicaciones, el uso de las computadoras, así como el aumento en la demanda de información actual en todos los ámbitos de la vida del ser humano han provocado que éste encuentre métodos cada vez más eficientes para trasladar esta información en cantidades y tiempos impresionantes.

En cuanto a las comunicaciones se refiere: la fibra óptica, las comunicaciones vía satélite, entre otros, han permitido enlaces a los lugares más recónditos del planeta, sumado a esto el surgimiento de las redes de computadoras que poco a poco han sabido aprovechar estos avances para sumarlos a las ya poderosas aplicaciones de los equipos de cómputo que permiten al hombre estar enterado de lo que ocurre no sólo en su lugar de estancia sino hasta el otro extremo del mundo.

Las primeras redes en surgir fueron las locales, la base y sustentación de lo que ahora es la gran INTERNET, basadas la mayoría de ellas en arquitecturas cliente - servidor son capaces de brindar recursos a todos los equipos conectados en su topología a través de PROTOCOLOS que son otra herramienta que se ha desarrollado enormemente, sin la ayuda de los protocolos de comunicación no habría la posibilidad de intercambio de información que es el fin principal de las redes.

I.1.- ¿ Qué es Internet ?

El origen de Internet lo podemos ubicar a finales de la década de los 60's, como parte de un proyecto del Departamento de la Defensa de Estados Unidos DARPANET (Defense Advanced Research Project Network) cuyo propósito principal era la investigación y el desarrollo de protocolos de comunicación para redes de área amplia (WAN) para ligar redes de transmisión de paquetes de diferentes tipos capaces de resistir las condiciones de operación más difíciles y continuar funcionando aún con la pérdida de una parte de la red como por ejemplo en caso de guerra y para requerir el mínimo de información de las computadoras que forman parte de la red.

Lo que se intentaba en un principio era conectar 4 lugares (El Instituto de Investigaciones de Stanford, la Universidad de Utah y las Universidades de California en Los Angeles y Santa Bárbara), el resultado de las investigaciones hechas arrojaron el nacimiento del protocolo TCP/IP (Transmission Control Protocol/Internet Protocol) con el cual para enviar un mensaje en la red, una

computadora sólo tiene que poner la información en un sobre, llamado paquete de protocolo Internet (IP:Internet Protocol) y le asigna el domicilio destino en forma correcta; las computadoras que se comunican -no la red- tienen la responsabilidad de asegurar que la comunicación se lleve a cabo, este sistema de comunicaciones es bastante sólido y bajo él, se integran todas las redes que conforman lo que es Internet. Durante el desarrollo de este protocolo la popularidad de la red creció tanto que para principios de los 70's contaba ya con 50 nodos o sitios conectados y para los 80's eran ya 200 sitios los que estaban conectados derivando con ello que la red se dividiera en dos subredes, la MILNET (red militar) y la ARPANET surgiendo el término INTERNET para referenciar a la red completa; a partir de esta separación se permitió el acceso a esta red a todo aquel que lo requiriera sin importar de qué país proviniera la solicitud siempre y cuando fuera para fines académicos o de investigación (y por supuesto, que pagaran sus costos de conexión), los usuarios (que cada vez fueron más) se encontraron que la información que fluía en la red era por demás útil y que si cada quien aportaba algo se enriquecería aún más el acervo existente.

Desde entonces ha crecido enormemente la demanda de Internet, conectando a miles de redes de institutos, universidades, empresas comerciales y de servicios alrededor del mundo, todos ellos bajo la filosofía de proporcionar el servicio de información, convirtiéndose así en un verdadero escaparate mundial.

Con respecto a quién es usuario y quién no, aunque parezca increíble en Internet no existe una autoridad central que controle ni su funcionamiento ni el acceso, sin embargo existen grupos y organizaciones que se dedican a organizar de alguna manera el tráfico en ella.

Por ejemplo el grupo considerado de mayor autoridad sobre el desarrollo de la red es la Internet Society (ISOC) creada en 1990 y formada por un grupo de voluntarios, cuyo propósito principal es el de promover el intercambio de información global a través de la Internet, este grupo puede tomar decisiones sobre lo que se hace en la red, está formado por miembros de Internet Architecture Board (IAB) que son quienes fijan los estándares de comunicaciones entre las diferentes plataformas de manera que puedan interactuar máquinas de diferentes fabricantes sin problemas, este grupo es responsable de cómo se asignan direcciones y otros recursos de la red, aunque realmente no son ellos los que realizan estas asignaciones sino que fijan las reglas de asignación.

En Internet todos tienen opinión acerca de cómo deben hacerse las cosas, y lo pueden expresar a través de las reuniones del Grupo de Trabajo de Ingeniería de Internet (IETF: Internet Engineering Task Force). El IETF es otra organización

voluntaria que se reúne para discutir problemas operacionales y problemas técnicos a corto plazo.

Y ¿quién paga Internet? Dado que no existe la empresa Internet, ¿cómo se paga lo que cuesta? Bueno, esto es algo sencillo de contestar, cada red incorporada a Internet paga por lo que cuestan sus conexiones a otra red y por sus comunicaciones, de esta forma Internet no es tan gratis como todos podríamos imaginar.

1.2 Estructura física de Internet

Internet ha sido por mucho tiempo una red internacional, (red de redes) como se le conoce comunmente, pero dado que el origen de ésta fue en Estados Unidos sólo se había extendido hacia los países con los cuales éste mantenía buenas relaciones y hacia las bases militares que tiene fuera de su territorio. Ahora, con las situaciones militares menos tensas, Internet se ha esparcido por todos lados, actualmente se le puede encontrar en más de 60 países y siguen aumentando ya que ahora todos los países principalmente los del llamado Tercer Mundo, lo ven como un medio para elevar sus niveles educativos y tecnológicos.

El surgimiento de Internet y su apertura así como su estandarización se vieron opacados en un principio por la Organización de la Estandarización Internacional (ISO:International Organization for Standardization) que dedicaba su tiempo en el diseño del último estándar de comunicaciones entre computadoras, sin embargo los desarrolladores de Internet difundieron el software de IP que en realidad era el único capaz de comunicar computadoras de diferentes arquitecturas.

Al mismo tiempo que Internet se consolidaba, las redes locales Ethernet se estaban desarrollando, las cuales llegaron a su clímax cerca de 1983, cuando aparecieron las primeras estaciones de trabajo de escritorio las cuales en su mayoría tenían sistema operativo UNIX en la versión de la Universidad de Berkeley la cual ya contenía el software de IP, y con esto ya todos podían conectarse a ARPANET, pronto muchas compañías empezaron a construir sus propias redes privadas utilizando los protocolos IP, por lo que los usuarios de una red podían comunicarse unos con otros.

1.2.1 TCP/IP bajo el modelo OSI

El sistema moderno de redes está constituido sobre el concepto de "niveles o capas" basado en el modelo OSI (Open Systems Interconnect).

Nivel de Aplicación	Capa 7
Nivel de Presentación	Capa 6
Nivel de Sesión	Capa 5
Nivel de Transporte	Capa 4
Nivel de Red	Capa 3
Nivel de Enlace de Datos	Capa 2
Nivel Físico	Capa 1

Capa 1

El nivel o capa física se refiere a los medios a través de los cuales se hace el envío de la información, estos medios pueden ser subterráneos, aéreos, a través de cables, fibra óptica, vía satélite, etc.

Capa 2

Verifica de acuerdo al protocolo que se utilice si el medio físico está apto para la transmisión. Esta capa se divide en dos partes: el Control de Enlace Lógico y el Control de Acceso al Medio.

Capa 3

La capa de red se encarga de 1) Control de enrutamiento de datos: para ello se cuenta con un equipo que guarda todas las posibles trayectorias para acceder a un punto de la red. 2) Direccionamiento de la red: cada nodo es a su vez otra red local y si queremos llegar a una terminal en especial necesitamos manejar su nombre y dirección. 3) Segmentación y creación de bloques de paquetes de información: divide la información y manda la señalización de inicio y fin. 4) Conexiones de red múltiple: manejo de los protocolos de comunicación para redes de diferentes características.

Capa 4

La capa de transporte determina la calidad del servicio, se encarga de la administración de los mensajes.

Capa 5

La capa de sesión, fija las prioridades de tráfico de datos, controla el flujo implícito para la capa de presentación, delimita el punto entre la información y el manejo de los datos. Establece y libera sesiones (negociación de conexiones y control de sesiones).

Capa 6

La capa de presentación verifica el contexto de la presentación, la transformación o conversión de datos.

Capa 7

Esta capa es la que maneja las aplicaciones.

TCP/IP es la solución a un gran número de problemas dentro de las redes ya que posee las siguientes características para garantizar una comunicación y transferencia de datos acorde a los requerimientos de calidad propuestos por el usuario.

- 1.- Es un estándar maduro que trabaja desde mediados de los años 70's.
- 2.- Es independiente de los fabricantes y está disponible para gran número de ellos; como ejemplos están : TriCom, Sun, DEC, HP, IBM.
- 3.- Se le puede conseguir para cualquier tamaño de máquinas.
- 4.- Está muy bien integrado a UNIX y altamente implantado.
- 5.- Soporta tecnología de enrutamiento dinámico.
- 6.- Soporta arquitectura cliente-servidor .
- 7.- Las aplicaciones OSI son capaces de correr en TCP/IP

TCP/IP es un conjunto de protocolos que transfieren la información como una secuencia de *datagramas* . Un *datagrama* es una colección de datos que se manda como un mensaje único. Cada *datagrama* es mandado a través de la red de manera individual, existen *datagramas* que abren conexiones, por ejemplo, una

conversación; sin embargo, estas conversaciones son divididas en *datagramas* y éstos son tratados por la red de manera separada, es así por ejemplo: un mensaje dividido en 14 *datagramas* puede ser recibido en un orden diferente, es decir. el *datagrama* 14 puede llegar antes del 13.

Un *datagrama* es la unidad de datos con la cual trabajan los protocolos.

Mencionar *datagrama* o paquete es casi indistinto dentro de la transmisión de datos, aunque técnicamente la palabra correcta a utilizar es *datagrama*, un paquete es una cosa física que pasa a través de los cables de la red Ethernet por ejemplo. en la mayoría de los casos un paquete simplemente contiene un *datagrama*.

IP es un protocolo que opera en el nivel de Internet proporcionando los mecanismos necesarios para el transporte de unidades de datos denominados *datagramas* a este nivel :

Nivel 1	Nivel 2	Nivel 3	Nivel 4
0-1	Frames	Paquetes	Mensajes

Este protocolo fue diseñado para operar sin conexión por lo que es posible que los *datagramas* se pierdan y se dupliquen o lleguen fuera de secuencia entre dos estaciones finales, por lo que para el control de estos problemas se emplea el Protocolo de Control de Transmisión.

IP implanta 2 funciones básicas: direccionamiento y fragmentación de la información.

La forma en la que el protocolo IP transmite la información de una a otra máquina conectada a red, es transmitiendo *datagramas* de un módulo de Internet a otro hasta que alcance su dirección destino, los módulos de Internet se definen por la forma en la que los enrutadores están interconectados.

En el direccionamiento se debe hacer la diferencia entre el nombre, la dirección y la ruta para la transmisión de la información. El nombre indica lo que se busca, la dirección indica dónde se encuentra y la ruta nos dice cómo llegar a ella y obtenerla.

La fragmentación de un *datagrama* es necesaria cuando éste origina en la red local paquetes largos en tamaño que se van a transmitir a una red que maneja los paquetes de tamaño más pequeño, la fragmentación no es otra cosa que la división

de la información en un número arbitrario de piezas que posteriormente serán reensambladas.

IP a su vez nos proporciona sistemas de recuperación de errores y cuenta con mecanismos de control de flujo, por lo que ejecuta descubrimientos de rutas en el intercambio de información que se emplea en la conexión de paquetes, esto conocido como conmutación.

De la cual utiliza los siguientes tipos:

a) Circuitos: Proporciona simplemente una ruta para las sesiones entre componentes para la conmutación de datos, este tipo se clasifica en tres arquitecturas:

- 1) Concentración (más líneas de entrada que de salida)
- 2) Expansión (más líneas de salida que de entrada)
- 3) Conexiones iguales.

b) Conmutación de mensajes: Requiere de un procesador especializado responsable de aceptar tráfico de terminales y computadoras conectadas mediante el examen de la dirección en el encabezado del mensaje y la conmutación del tráfico ya sea al conmutador de ruta siguiente o a la estación receptora.

c) Conmutación de paquetes: Los datos son divididos en paquetes y enrutados como entidades independientes dentro de la red, entre los factores que intervienen en la elección del algoritmo para la comunicación se encuentran los siguientes:

- 1) Capacidad de enlace
- 2) Número de paquetes a ser transmitidos
- 3) Carga en la red
- 4) Tipo de tráfico vs. Tipo de enlace
- 5) Número de enlaces intermedios entre transmisor y receptor.
- 6) Habilidad para enlazar los nodos intermedios y los nodos destino.

1.2.2 Características del direccionamiento en Internet

1.-Cada dispositivo en Internet es asignado a una dirección única.

2.-Esos dispositivos pueden ser servidores de comunicación, puertos, puentes, etc.

3.-El direccionamiento Internet utiliza un campo de direcciones de 32 bits el cual divide las direcciones dentro de la red en una parte de direccionamiento de red y en una parte de direccionamiento local.

4.-Estos 32 bits son divididos a su vez en cuatro octetos separados por un punto.

5.-Conceptualmente cada dirección es un par (net#.host#) en donde net# identifica una red y host# identifica un *host* en la red.

6.- Se dividen las direcciones en tres clases, de acuerdo al par que identificará tanto a la red como al *host*.

La clase A ocupará el primer octeto para identificar la red y los siguientes 3 octetos para identificar al *host*. Por lo cual podemos tener 16,777,214 direcciones IP, es decir *hosts* en nuestra red.

La clase B ocupará los dos primeros octetos para identificar la red y los 2 siguientes para identificar el *host*. Por lo que tenemos 65,534 *hosts* en una red tipo B.

La clase C ocupará los 3 primeros octetos para identificar la red y el último octeto para identificar el *host*. Por lo cual podemos tener 254 direcciones IP, en nuestra red.

Reglas de direccionamiento:

1a. No se permite ningún direccionamiento con los 3 octetos de mayor orden en 1.

2a. El octeto utilizado para definir la porción del *host* de una dirección Internet no puede tener todos los bits en 1, de acuerdo al estándar cualquier dirección Internet con la porción del *host* integrada por valores de 1 está reservada a la dirección de *broadcast* (mensaje a todas las máquinas, por lo general de advertencia).

3a. Los bits utilizados para definir la red o la porción del *host* de una dirección Internet no puede tener todos los bits en 0 ya que algunas aplicaciones utilizan este valor para dirección de *broadcast*.

1.2.3 Dominios en Internet

Un dominio es una agrupación de computadoras a las cuales se hace referencia a través de un nombre, el nombre del dominio. Puede decirse que un dominio es el nombre de una red, y que éste puede contener a otras redes.

Para hacer referencia a cualquier computadora conectada a Internet basta con saber su dirección IP, que como ya se mencionó es un número compuesto de 4 octetos. Sin embargo, el manejar números puede llegar a ser muy complicado, para un usuario que tiene que trabajar con múltiples máquinas todo el tiempo, aprenderse el número de cada máquina requiere muy buena memoria, es por ello que se crea los llamados Dominios que no son otra cosa que la agrupación de un conjunto de máquinas que tienen en común parte del número de la dirección IP y a las cuales se les asocia un nombre, así por ejemplo, una red tipo B con direcciones 132.248.54.x, es una red que pertenece al dominio fi-a.unam.mx, es decir, que la dirección 132.248.54, corresponde al dominio fi-a, mientras que el número 132.248 corresponde al dominio unam.mx, es decir todas las máquinas cuya dirección IP comience con 132.248 están dentro del dominio de la red unam, que a su vez está en el dominio de todas las redes que están en México (mx), de tal manera que:

La máquina cuya dirección IP es 132.248.54.10 tiene asociado el nombre en dominios : cancen.fi-a.unam.mx, con lo cual asumimos que toda máquina en Internet tiene un nombre asociado en términos de dominio.

El funcionamiento de lo antes mencionado en Internet se realiza de la siguiente forma:

Las diferentes partes de Internet están conectadas por un conjunto de computadoras llamadas enrutadores, que interconectan las redes. Estas redes pueden ser Ethernet, token ring o en ocasiones líneas telefónicas.

Las Ethernet y las líneas telefónicas son el medio a través del cual la información va de un lado a otro. Los enrutadores son los encargados de dirigir la información. No toda subestación o todo enrutador cuenta con una conexión a cada uno de los otros enrutadores de la red, así por ejemplo cuando quiere mandar un paquete de información a un enrutador del cual no tiene su dirección, lo envía al más cercano y éste a su vez al más cercano, hasta que se encuentra con un enrutador que tiene la dirección de la máquina destino. De acuerdo con lo anterior, basta con poner el paquete de información en la red y de marcar su dirección; ya que como se vió, el protocolo IP garantiza que esa dirección sea única por lo cual no hará confusión para encontrar la máquina destino.

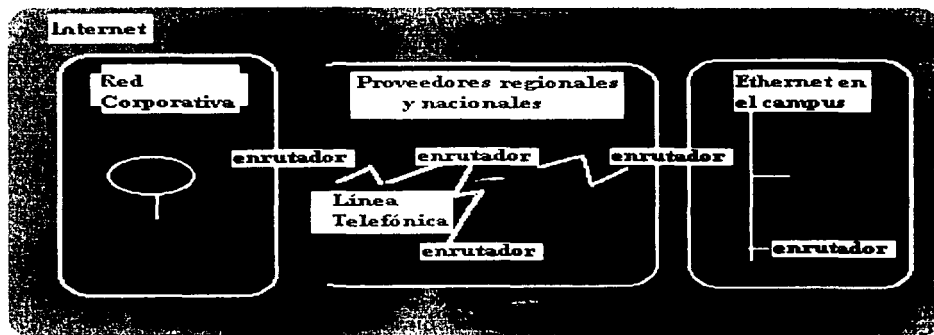


Fig. 1 Arquitectura de la red Internet.

Un enrutador (*Gateway or Router*), es un sistema que conecta una red con otra u otras redes; un enrutador, es una máquina normal que tienen más de una interfase de red, esto es, que puede tener conexiones con distintas redes al mismo tiempo y que es capaz de direccionar los *datagramas* que le llegan a la red correspondiente.

El enrutamiento se hace a través de las direcciones IP asociadas a los *hosts*: cada máquina tiene un tabla de direcciones de red, en cada una de estas tablas debe existir por lo menos la dirección de un enrutador.

Cuando una computadora quiere enviar un *datagrama* verifica primero si la dirección IP a la cual la enviará se encuentra en su red local, de ser así la información es transmitida directamente, de otra forma, la máquina tiene que esperar a que le respondan si esa máquina se encuentra o no dentro de la red local, al no encontrar respuesta positiva la información es mandada al enrutador, si este enrutador puede alcanzar la dirección IP a la cual va dirigida la información la redirecciona hacia ella, de no ser así, manda un mensaje a la máquina origen de la información informándole que no es el mejor enrutador, pero brinda la posibilidad de salir hacia otro enrutador que tal vez dé el direccionamiento hacia la máquina destino, esto debido a que la lista de direcciones de un enrutador contiene las direcciones IP de los enrutadores de las redes a las que se puede conectar.

Como ya se mencionó TCP/IP es un conjunto de protocolos de los que se ha descrito ya a IP y ahora se hará lo propio con TCP.

El protocolo TCP crea la apariencia de que existe una conexión permanente entre dos aplicaciones, garantizando de esta forma que lo que se transmite de un lado llegue al otro. Realmente no se cuenta con un enlace directo entre emisor y receptor (otras personas pueden usar los mismos enrutadores y la red de cableado en los lapsos que ocurren entre el envío de cada paquete).

El protocolo TCP toma la información que se desea enviar y la divide en segmentos. Además, enumera cada segmento para que el receptor pueda verificar la información y ponerla en el orden adecuado. Para que el protocolo TCP pueda enviar esta secuencia de números a través de la red, cuenta con su propio sobre que le permite "escribir" en él la información a transmitir. Este sobre es puesto, a su vez, dentro del sobre del protocolo IP y posteriormente es transmitido a la red. Una vez que se pone algo en un sobre IP, la red lo puede transmitir.

Del lado del destinatario, una parte del software de TCP reúne los sobres, extrae la información de ellos y la pone en el orden adecuado. Si algún sobre se pierde en la transmisión, el receptor solicita su retransmisión al emisor. Una vez que el protocolo TCP tiene toda la información en el orden adecuado, la pasa a la aplicación del programa que esté utilizando sus servicios.

Los paquetes no sólo se pueden perder, además de esto pueden ser modificados por el mal funcionamiento durante la transmisión a través de líneas telefónicas. El TCP también resuelve este tipo de problemas. Así como coloca la información en un sobre, el protocolo calcula algo llamado "número de verificación". El número de verificación es un número que permite que el receptor TCP detecte errores en el paquete transmitido. Cuando un paquete llega a su destino, el receptor calcula el número de verificación y lo compara con el enviado por el transmisor. Si no coinciden, significa que ocurrió un error en la transmisión. El receptor deshecha el paquete y solicita la retransmisión.

TCP no es tan ideal como parece, si toda la información que se requiere transmitir cabe en un solo paquete y no es necesario garantizar su entrega, el protocolo TCP puede ser desperdiciado.

Para ello existe otro protocolo estándar que evita el exceso de trabajo. Este protocolo es el Protocolo de Datagramas de Usuario (UDP). Es utilizado en algunas aplicaciones en el lugar del TCP; esto es, en lugar de tomar la información y ponerla en un sobre TCP para después ponerla en un sobre IP, esta aplicación pone la información en un sobre UDP y después en un sobre IP.

El protocolo UDP es mucho más sencillo porque no se preocupa de que los paquetes se pierdan, ni porque la información llegue en orden o cualquier situación de ese tipo. El UDP se usa comúnmente en programas que envían mensajes cortos y que sólo reenvían la información si no reciben una respuesta en un tiempo determinado. Y que no se preocupan por poner *datagramas* en secuencia.

Este protocolo proporciona un procedimiento para programas de aplicación que mandan mensajes a otros programas con los mecanismos mínimos de un protocolo. Es un protocolo orientado a la transacción de datos, sin importar la duplicación o la pérdida de información.

Otro protocolo es ICMP (Internet Control Message Protocol) y es utilizado para captar mensajes de error y algunos otros que tiene que ver con las llamadas al software del protocolo en sí mismo, por ejemplo cuando estamos tratando de conectarnos a un *host* y recibimos un mensaje que dice que el *host* no puede ser alcanzado es una labor propia de este protocolo que es la de buscar información dentro de la red para tenernos informados del estado de las conexiones.

Sin embargo, éstos no son los únicos protocolos con los cuales podemos acceder a Internet: por otro lado tenemos SLIP, PPP, CSLIP que se explican a continuación.

I.3 Protocolos que maneja Internet

Además de los ya mencionados TCP e IP, tenemos protocolos de acceso a Internet de manera dedicada, el acceso dedicado ofrece la conexión más flexible. Cada computadora es un miembro de Internet con todos los derechos, capaz de realizar cualquier función de la red. Los proveedores rentan una línea telefónica dedicada, a la velocidad que desee, e instala un enrutador especial en sus instalaciones. Este enrutador es responsable de tomar las comunicaciones de su instalación que están dirigidas a otros lugares y de enviarlas a su destino.

SLIP (Serial Line Internet Protocol) y PPP (Point-to-Point Protocol)

SLIP permite utilizar TCP/IP sobre una línea serial, esto es una línea telefónica con un módem a través del cual se hace la conexión a un servidor de SLIP conectado directamente a Internet. Fue desarrollado en 1984 para soportar transmisiones de red con TCP/IP a baja velocidad de transmisión serial por el Unix de Berkeley.

PPP fue diseñado para realzar las características de SLIP, es un conjunto de protocolos que permite la conexión serial con múltiples proveedores de red.

permite una transmisión de *datagramas* a través de una red punto a punto y permite la utilización de múltiples protocolos entre los cuales está por supuesto TCP/IP.

SLIP y PPP, son versiones del software de Internet que corren en líneas telefónicas normales, empleando módems estándar de alta velocidad. Con SLIP y PPP ni siquiera se tiene que usar la línea telefónica "dedicada": puede conectarse a la red vía línea telefónica cuando quiera tener acceso, dejando la línea libre para otro uso cuando no necesite conectarse. La ventaja real tanto de SLIP como de PPP es que permiten tener conexión completa a Internet, y no se utiliza otro equipo como "punto de acceso" a la red, uno mismo está dentro de la red.

SLIP y PPP son muy apropiados para conectar una computadora casera a una red local más grande que a su vez esté conectada a Internet, por ejemplo se puede usar SLIP para conectar una computadora casera a la red de una compañía o universidad de esta manera, la computadora casera tendrá acceso completo a Internet como si estuviera en la *Ethernet* de la compañía.

SLIP es una opción de costo moderado y provee muy buen servicio.

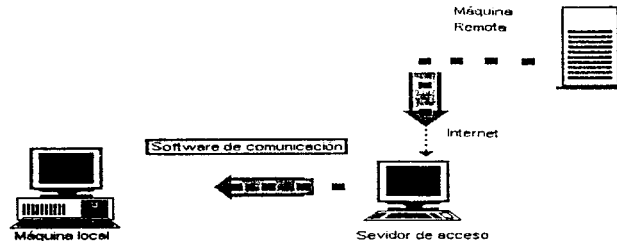


Figura que muestra la conexión de una máquina casera a un servidor de acceso en una ethernet que da salida a Internet.

Acceso ISDN

ISDN "Red Digital de Servicios Integrados" (Integrated Services Digital Network).

Lo que en realidad significa utilizar una línea telefónica entre una computadora casera u oficina y la central de la compañía de teléfonos u "oficina central", esto

podría sonar a una nueva tecnología, pero no lo es. Aunque está apenas empezando a utilizarse en Estados Unidos.

El acceso a ISDN puede ser a través de línea conmutada (acceso intermitente, según se necesite), o dedicado (una conexión permanente a Internet). La ventaja de ISDN es que provee un acceso de muy alta velocidad a un costo relativamente bajo. Un canal de ISDN incluye 2 canales digitales de 56 o 64 KB. Con velocidades de acceso como éstas, los servicios de multimedia se ejecutan a altas velocidades.

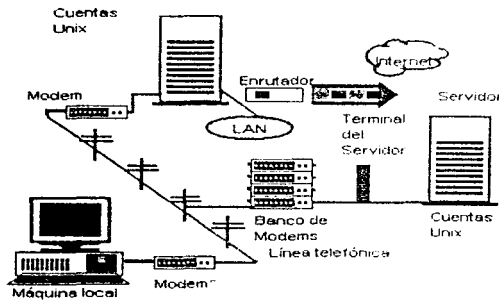


Figura que muestra el acceso ya sea ISDN o bien línea conmutada

Acceso vía línea conmutada

Para ello se necesita una cuenta de tiempo compartido en una máquina de acceso dedicado, después desde una máquina casera se puede hacer la conexión a este sistema remoto, el acceso de tiempo compartido es casi tan bueno como tener su propia conexión y es más fácil de implantar, la computadora casera realmente no es parte de la red sino simplemente está teniendo acceso a un servicio de cómputo conectado permanentemente a la red. En este caso como el acceso se comparte el costo es más reducido.

Acceso UUCP (Unix to Unix Copy)

Existe una subclase de acceso vía línea conmutada. Todos los sistemas Unix manejan una serie de servicios llamada UUCP, que transfiere datos por líneas telefónicas estándar. UUCP se utiliza principalmente para recoger correo electrónico de Internet y artículos de texto; lo que hace realmente UUCP no es

tener una conexión permanente a la red sino que realiza estas conexiones periódicamente vía una línea conmutada a una computadora en Internet y transfiere archivos.

Acceso vía otras redes

Existe algunas redes conectadas a Internet que permiten el acceso a ésta aunque los servicios que prestan son realmente pobres. Correo electrónico y transferencia de archivos.

Capítulo II

Capítulo II El World Wide Web

II.1 Antecedentes

El World Wide Web es un concepto que ha surgido para nombrar al conjunto de redes locales interconectadas a través de un servicio gráfico de información. En su traducción más aproximada podríamos decir que es "Telaraña a lo ancho del mundo", que realmente describe muy bien las conexiones entre todas las máquinas dentro de lo que es Internet.

Existen muchos servicios que anteceden a lo que ahora se conoce como *el Web* entre ellos se encuentran servicios de transferencia de archivos, de búsqueda de información, servicio de correo electrónico, información catalogada, etc.

Estos servicios surgen de la necesidad de brindar ayuda a los usuarios de la red para poder localizar de manera más fácil la información y los recursos compartidos de otras redes conectadas a Internet.

Los servicios de Internet son muchos y se encuentran en todas partes, de hecho podemos considerar que algunas de las utilerías del protocolo TCP/IP como lo son Telnet y FTP son servicios de Internet.

Para poder determinar la importancia del Web hablaremos a continuación de las características de algunos de los servicios que le anteceden.

II.1.1 Sesión remota

La esencia de lo que es una red, es el compartir recursos, en este caso hablando específicamente de Internet el recurso más importante es la información; sin embargo, para poder brindar este servicio se necesita que los servidores que comparten su información permitan el acceso a los usuarios que quieren utilizar su información, es ahí cuando surge el concepto de sesión remota; una sesión es una instancia de un usuario en el servidor, es decir, un servidor puede estar siendo utilizado por múltiples usuarios al mismo tiempo y cada usuario entonces tiene una o más sesiones en el servidor.

El hecho es que un usuario no necesita estar físicamente sentado frente a la máquina que contiene la información que le interesa para poder acceder a ella, sino que a través de la red puede solicitar tener una sesión en el servidor desde su máquina local.

Para lograr esto basta con tener una máquina conectada a Internet y hacer uso del protocolo asociado al comando TELNET (Terminal emulation) "Emulación de Terminal", para ello el usuario debe tener una clave de acceso en el servidor al cual está tratando de conectarse, al lograrlo es como si el teclado de la máquina local estuviera conectado al servidor remoto.

Este protocolo consta de dos partes de software que colaboran entre sí: el cliente, que corre en la computadora que solicita el servicio y el servidor que provee el servicio. La red, sin importar el protocolo que utilice es el medio que permite la comunicación entre ambos.

El cliente, que es el programa que debe actuar en la máquina local, al momento de escribir el comando **telnet**, hace lo siguiente:

- Crea una conexión de red TCP con un servidor.
- Acepta lo que se escriba en el teclado proveniente del usuario.
- Da formato a la entrada de datos del usuario y la manda al servidor.
- Acepta la respuesta del servidor.
- Da formato a las respuestas del servidor para desplegarlas en la pantalla del usuario.

El software del servidor actúa en la máquina que provee el servicio, si el servidor no está funcionando el servicio no está disponible. El servidor realiza lo siguiente:

- Informa al software de red que está listo para aceptar conexiones.
- Espera una solicitud de sesión remota.
- Atiende la solicitud.
- Envía los resultados de regreso al cliente.
- Regresa a un proceso de espera.

La forma de establecer una sesión remota es a través del siguiente comando:

```
telnet dirección_IP  
telnet nombre_en _dominio
```

ejemplo:

```
telnet 132.248.54.10  
telnet cancun.fi-a.unam.mx
```

Cabe hacer notar que el establecer una sesión en el servidor remoto implica hacer uso de la información y software que los privilegios de la clave con la que

establecemos la sesión nos permita. Es decir, si no tenemos privilegios no podemos acceder a toda la información del servidor.

Para establecer una sesión remota podemos utilizar también el comando RLOGIN (Remote Login) "Acceso Remoto", que tiene las mismas características que Telnet con la diferencia que para poder ejecutarlo debemos asegurarnos que existe una clave de acceso igual tanto en el servidor remoto, como en el local y conocer el password de ambos, ya que nos lo pide para confirmar que efectivamente somos los dueños de la clave de acceso.

El Rlogin es una variante del Telnet muy utilizada sobre todo en redes locales que tiene varios servidores y que comparten su base de datos de usuarios, es decir, realmente sólo un servidor es el encargado de validar el acceso de los usuarios.

La forma de crear una sesión remota con Rlogin es de la siguiente forma:

rlogin dirección_IP
rlogin nombre_en_dominio

ejemplo:

```
rlogin 132.248.54.10  
rlogin cankun.fi-a.unam.mx
```

Al hacer esto garantizo que la clave con la que estoy trabajando en una sesión de mi servidor local es la misma clave con la que intento tener una sesión en el servidor remoto y sólo preguntará por el password.

II.1.2 Transferencia de archivos (FTP)

Comúnmente se puede encontrar información en Internet. Si encontramos un servidor con información interesante, lo más lógico será querer tener esa información en nuestra máquina local, por lo que debemos hacer una copia de esa información y traerla, la forma de lograrlo es a través de otra herramienta que proporciona el conjunto de protocolos TCP/IP y que se conoce como FTP (File Transfer Protocol) "Protocolo de Transferencia de Archivos".

El principal objetivo de FTP es precisamente el mover archivos de un servidor a otro no importando dónde se encuentren o bien qué sistema operativo estén utilizando, siempre y cuando se comuniquen a través del protocolo FTP.

FTP permite una conexión a un servidor remoto bajo la consigna de FTP que es precisamente la transferencia de archivos, no obstante cumple con características de

una sesión remota, por lo cual se requiere una clave de acceso, esta clave por lo general es una clave estándar para cualquier usuario, es decir, el servidor que va a compartir su información debe tener una clave que permita el acceso de los usuarios a esa información.

Normalmente se han utilizado dos claves estándar dentro del manejo de este protocolo y son las claves "guest" y "anonymous", esta última es la más reconocida ya que incluso a los servidores que permiten el uso de FTP se les conoce como servidores FTP anónimos.

No obstante para tener una sesión de transferencia de archivos podemos utilizar también nuestra clave de acceso al servidor FTP, si es que tenemos clave en él; la diferencia es que teniendo una clave de acceso en el servidor adquirimos los permisos que esa clave tiene, siendo que éstos pueden ser mayores o menores que los de las claves guest y anonymous.

Para establecer una sesión remota a través de FTP lo hacemos con el siguiente comando:

```
ftp dirección_IP  
ftp nombre_en_dominio
```

ejemplo:

```
ftp 132.248.54.13  
ftp cozumel.fi-a.unam.mx
```

Una vez que ya se estableció la sesión, nos encontramos trabajando en el servidor remoto y podemos visualizar todos los archivos que forman parte de la información disponible para ser transferida.

La forma de transferir información es a través de algunos comandos propios de la utilería FTP.

Por ejemplo si queremos traer información a nuestra máquina local lo hacemos con el comando **get** de la siguiente forma:

```
ftp> get nombre del archivo en el servidor remoto
```

Un servidor de FTP puede permitirnos el transferir archivos de nuestra máquina hacia él, esto es, nosotros podemos compartir información a través de un servidor FTP remoto; lo hacemos de la siguiente forma:

ftp> put nombre del archivo en la máquina local

En general, se puede decir que la mayoría de los servicios que brinda Internet para la transferencia de archivos utilizan FTP anónimo

II.1.3 Archie

Históricamente, uno de los grandes problemas de Internet ha sido encontrar lo que sabemos que existe, por algún tiempo los servidores de FTP anónimo fueron utilizados sólo por aquellos conocedores de la red, que entre ellos se comunicaban las direcciones de los servidores en los cuales existía información interesante, pero ahora con el gran crecimiento de la red es imposible saber la direcciones de todos los servidores.

Esta es la razón del surgimiento de Archie. Es un sistema que permite explorar índices disponibles en los servidores públicos especiales. Hasta 1994 existían aproximadamente 1200 servidores y 2.5 millones de archivos catalogados por Archie; como usuario se le puede pedir que encuentre nombres de archivos que correspondan a ciertos criterios de búsqueda o que muestre archivos que contengan ciertas palabras. Archie devuelve los nombres de archivo que concuerdan con el criterio de búsqueda y el nombre de los servidores que contienen esos archivos. Después podemos decidir cuál de esos archivos nos conviene más y lo podemos trasladar a nuestra computadora con FTP anónimo.

Archie crea su catálogo preguntando por toda la red quiénes están ejecutando programas servidores de FTP anónimo para anotarlos. Esto se hace cada mes, se conecta a los servidores FTP anónimo y lista su directorio guardando el contenido en su base de datos, así cuando nosotros preguntamos por una palabra en específico, Archie explora en los directorios catalogados y nos envía los nombres de archivo que concuerdan con la cadena proporcionada, junto con los nombres de los servidores, el servicio de Archie lo brindan ciertas máquinas que tienen instalado el software correspondiente, es por ello que para poder utilizarlo debemos tenerlo corriendo en un servidor local o bien conocer la dirección de un servidor remoto que tenga una cuenta especial con la cual podamos acceder a Archie. Existen múltiples servidores de Archie en todo el mundo y lo más importante es que existe comunicación entre ellos permitiendo así que no todos los servidores tengan toda la información acerca de los servidores de FTP anónimo, sino que si un servidor no encuentra información referente a la cadena de búsqueda solicitada puede conectarse a otro servidor de Archie y pedirle que revise en su catálogo.

Las consultas realizadas a cualquier servidor Archie nos reportan los resultados de su búsqueda a través de otro servicio de Internet llamado correo electrónico del cual

hablaremos más tarde, lo importante de ello es el hecho de que nosotros podemos tener el resultado de la consulta en un archivo para poder acceder a él cuando nosotros queramos, el contenido de este archivo ya se había mencionado; contiene la dirección IP del servidor en el cual encontró un archivo con la cadena de búsqueda solicitada, así como la localización dentro de la estructura de directorios del servidor FTP anónimo del archivo correspondiente. Cabe aclarar que el resultado de la búsqueda puede reportar un número muy grande de posibilidades o bien no encontrar ningún servidor.

Una vez localizados tanto el servidor como el archivo que nos interesa, bastará con crear una sesión de FTP para traer el archivo correspondiente a nuestra máquina local.

II.1.4 Correo electrónico

El correo electrónico fue realmente el primer servicio de Internet así catalogado, es una aplicación que venía integrada ya con el Unix de Berkeley a raíz de la creación de TCP/IP y que consiste en la transferencia de mensajes dirigidos entre usuarios de una red, es decir un usuario dirige un mensaje en específico a otro usuario y lo hace como el correo tradicional, siendo la dirección a la cual va dirigida la carta, la dirección electrónica (e-mail o apartado postal) del otro usuario, al tener una clave en un servidor Unix tenemos asociada una dirección electrónica y dependerá de la forma en la que está conectado nuestro servidor con respecto a Internet si es que puedo o no mandar cartas a cualquier persona dentro de Internet o sólo a aquellas personas que tengan clave en mi red local.

La forma en la que se mandan los mensajes depende totalmente del protocolo TCP/IP ya que son mensajes totalmente dirigidos a una clave en un servidor específico.

La forma en la que se asigna el e-mail o apartado postal de cada usuario es de la siguiente forma:

clave_de_acceso@nombre del servidor en el que tenemos clave
ejemplo:

chc000@cancun.fi-a.unam.mx

donde chc000 es la clave de acceso del usuario y cancion.fi-a.unam.mx es el nombre del servidor, o bien:

chc000@[132.248.54.10]

donde chc000, de igual forma, es la clave de acceso del usuario, mientras que el número dentro de los corchetes es la dirección IP asociada al servidor.

Dependiendo del sistema operativo que esté ejecutando el servidor en el cual tenemos nuestra clave de acceso, va a ser el software correspondiente al correo electrónico, así podemos decir que existen múltiples aplicaciones para ejecutar el correo electrónico, entre las más utilizadas están :

mail
mailx
elm
eudora
pine
etc.

El estándar hasta el auge de Internet es el mail, la forma de mandar un mensaje a través de esta utilería es la siguiente:

mail *e-mail de la persona a la que va dirigido el mensaje*

después de esto nos pregunta por un *subject* (asunto o tema del mensaje) después nos permite editar el mensaje en línea, es decir, crear el mensaje escribiendo desde el teclado mientras se refleja en pantalla; una vez terminado el mensaje, con las teclas CTRL-D se manda.

En particular con esta utilería, si no recibimos ningún mensaje significa que el mensaje fue enviado, en caso contrario el mensaje es devuelto y guardado en un archivo de error llamado "*dead.letter*", en particular el correo electrónico utiliza como protocolo de comunicación UDP (*User Datagram Protocol*) es decir, un protocolo sin conexión, puesto que no es necesario establecer una comunicación directa con la máquina que recibirá el mensaje y aún más no es necesario que exista una sesión de la persona a cuyo buzón va dirigido el mail, con que exista el buzón es suficiente para que éste llegue.

En general, todo el software para correo electrónico realiza las mismas acciones, la diferencia es la interfaz que se le presenta al usuario, cada vez son más amigables y fáciles de utilizar.

II.1.5 Gopher

Gopher es un servicio (software), que permite encontrar información a través de la red en forma catalogada, esto es, existe una base de datos con la información clasificada de tal manera que el usuario cuando accede a este servicio puede conectarse directamente al servidor en el cual está la información que está viendo en el catálogo, la forma en que accede gopher a los servidores donde se encuentra la información es a través de FTP anónimo.

Al igual que Archie, existen múltiples Gophers en toda la red y entre ellos existe comunicación; por lo que si en determinado momento no encontramos información en el catálogo en el que estamos, podemos conectarnos a otro y buscar en él. la ventaja de Gopher consiste precisamente en el hecho de que la información está clasificada y podemos darnos mayor idea de hacia dónde dirigimos para realizar nuestra búsqueda.

Gopher realiza las acciones que hacen Archie y FTP, es decir, hace conexiones de FTP anónimo para acceder a la información mostrada en el catálogo y no obstante puede mostrarla de manera que podemos valorar si realmente es información que nos interesa. Una ventaja más sobre Archie y FTP, que para ver el contenido de algún archivo que parece interesante tenemos que ejecutar comandos del sistema operativo y más aún muchos de los archivos que se encuentran en los servidores de FTP anónimo se encuentran comprimidos por lo que tenemos que descomprimirlos y después listar su contenido, mientras que en Gopher la podemos ver y también la podemos mandar en un mail a nuestro buzón para poder manipularla, y todo lo podemos hacer desde la interfaz que nos brinda Gopher.

La interfaz que Gopher brinda es tipo texto, cuenta con menús de comandos dependiendo de la parte del catálogo en el cual se encuentra; en terminales gráficas podemos utilizar el *ratón* para desplazarnos entre las opciones del catálogo o bien existen teclas de control que nos permiten ejecutar una acción en específico.

¿ Cómo acceder a Gopher ? Para poder utilizar Gopher necesitamos conocer la dirección IP de algún servidor que cuente con este servicio, como en la mayor parte de los servicios en Internet todos podemos utilizarlo si conocemos su dirección IP y conocemos alguna clave de acceso.

Para ello hacemos uso del Telnet o Rlogin.

```
telnet dirección_IP_del_servidor_gopher  
telnet nombre_en_Dominio_del_servidor_gopher
```

ejemplo:

```
telnet condor.dgsca.unam.mx
```

La mayoría de los servidores de Gopher en el mundo tienen una clave de acceso llamada 'gopher' sin *password*, esto para garantizar que todos puedan utilizar este servicio sin necesidad de tener una cuenta especial en el servidor, además de que esta clave cuenta con los privilegios necesarios para ejecutar el software de Gopher y claro acceder a todos los servidores de información de aparecen catalogados en él, así como la ejecución del correo electrónico cuando se requiere enviar información al usuario en la máquina local.

El Gopher es un servicio sustentado en una base de datos que a diferencia de Archie sí tiene que actualizarse manualmente, es decir que los creadores de su Gopher tienen que preguntar a aquellos que brindan servicio de información a través de la red si quieren tener una entrada en su catálogo y así crear su vínculo correspondiente.

Por otro lado la información dentro del Gopher se restringe sólo a aquella que aparezca catalogada, sin que esto represente una limitante mayor ya que cuenta con la conexión hacia otros Gophers en los que probablemente se encuentre información referente a lo que estamos buscando. La conexión se hace automática, aunque en ocasiones al tratar de acceder a otros Gopher nos indica la clave y *password* que debemos utilizar para poder hacer uso de sus servicios.

Con Gopher al igual que con FTP traer hacia nuestra cuenta de usuario la información deseada, se realiza a través del correo electrónico, es decir una vez que localizamos la información deseada Gopher nos puede mandar en un mail el contenido del archivo que está desplegando en pantalla.

Las búsquedas a través de Gopher a pesar de ser más sencillas que con Archie y FTP suele ser complicada ya que sólo navegamos a través de menús preestablecidos, es por ello que surge otra herramienta de búsqueda a través de catálogos de Gopher.

```
Gopher en: condor.dgsca.unam.mx

--> 1. Acerca de estos servicios de informacion/
    2. Informacion sobre la UNAM/
    3. Revistas y Boletines Electronicos/
    4. Noticias de Mexico y el Mundo/
    5. Consulta a Bases de Datos de la UNAM/
    6. Consulta a Catalogos en linea de Bibliotecas/
    7. Otros Servicios de Informacion en la UNAM/
    8. Otros Servicios de Informacion en Mexico y el Mundo/
    9. Presencia de Otras Instituciones en este Gopher/
   10. Herramientas de Busqueda en Internet/
   11. Directorio de Servicios por Tema/
   12. Centro de Informacion de la RedUNAM/
   13. Centro de Operacion de la RedUNAM/

Presione [2] para Ayuda, [3] Salir                               Pagina: 1/1
```

Figura que muestra el menú principal del gopher de d.g.s.c.a.

II.1.6 Verónica

Así como existe un servicio de búsqueda para FTP (Archie) , en Gopher existió también este servicio, Verónica.

Verónica como ya se mencionó es un servicio que nos permite realizar búsquedas por temas y palabras especiales sobre el servicio de Gopher, es decir, realiza la búsqueda sobre los catálogos y nos da como resultado la posición en el catálogo correspondiente, ya sea en el Gopher local o bien en algún Gopher remoto.

Verónica extrae datos exactamente igual que Archie: visita servidores de Gopher alrededor del mundo y consulta sus menús, recuerda lo que hay en cada uno de ellos y construye un índice combinado para presentarlo en un menú Gopher.

Lo más importante de Verónica es que se utiliza como cualquier índice del menú que nos presenta Gopher y solicita palabras para configurar la búsqueda e integrar los resultados en un menú especial. Los elementos de menú son elementos provenientes de todos los Gophers alrededor del mundo los cuales contienen en su nombre las palabras que se han especificado.

II.1.7 WAIS

“Servicio de Información de Gran Cobertura” (Wide Area Information Service) es otro de los servicios de Internet. WAIS permite realizar búsquedas a través de archivos que contengan ciertos grupos de palabras.

WAIS es un software diseñado para trabajar con conjuntos de datos o bases de datos.

WAIS es un sistema distribuido de exploración de texto. Está basado en un estándar llamado Z39.50 que describe la manera en que una computadora solicita a otra que realice una búsqueda en lugar de ella.

Para hacer que un documento esté disponible a través de un servidor WAIS alguien crea un índice para ese servidor con el fin de que sea utilizado en la búsqueda. Para información textual, cada palabra en el documento por lo regular esta indizada.

Cuando se solicita una búsqueda desde un cliente WAIS, éste se enlaza con el servidor que controla las bibliotecas que se han sugerido. El cliente WAIS solicita a cada servidor, en su momento, que explore su índice en busca de un conjunto de palabras. Entonces, el servidor envía una lista de documentos que pueden ser los apropiados y una “calificación” que le indica qué tan apropiado supone que es cada uno. Las calificaciones se otorgan de manera que el documento que mejor coincida con el criterio de búsqueda obtendrá la mayor puntuación por ejemplo 1000 y los demás van descendiendo proporcionalmente de acuerdo a las coincidencias con respecto a las palabras de búsqueda.

Sin embargo, esta búsqueda no es del todo certera o bien la más real ya que las calificaciones se otorgan a cada documento de acuerdo a la incidencia de las palabras de búsqueda, por ejemplo; se hace una búsqueda de las frase ‘matemáticas aplicadas a la computación’, en este caso puede ser que nos regrese un documento en el cual encontró la palabra ‘aplicadas’ más veces de lo que encontró la palabra ‘computación’ y sin embargo ese documento recibe la mayor puntuación, es por ello que no se puede considerar que es una forma cien por ciento eficaz, pero sí de gran ayuda.

II.2 Surgimiento del World Wide Web

Por un gran número de razones, Internet ha surgido como una rica fuente de información accesible sólo a través de herramientas no lo suficientemente amigables. Los comandos básicos como Telnet, FTP, Archie, e-mail u otros, son poderosos pero poco intuitivos, y el incremento en Internet ha provocado el crecimiento en el

número de usuarios quienes no tienen la paciencia y el tiempo para aprender estas interfaces.

Es por ello que la popularidad de Gopher creció rápidamente, pero Gopher es una fuente de información restringida a la información desplegada en sus tablas de contenidos y mediante la cual los usuarios pueden ver y obtener información.

El World Wide Web comienza a existir a raíz de un proyecto para distribuir información científica a través de computadoras en una red por medio de un sistema conocido como hipertexto. La idea fue el permitir a los investigadores presentar sus investigaciones con texto, gráficas o algunos otros elementos de multimedia.

El World Wide Web es conocido también como WWW, es un concepto asociado al servicio más reciente en Internet. El Web esta basado en una tecnología flexible y sencilla que permite "navegar" por Internet, sus aplicaciones están basadas en sistemas hipermedios distribuidos y continúan desarrollándose

Los tres elementos principales que componen el Web son:

- El hipertexto.
- Internet misma, es decir, el conjunto de redes conectadas.
- Multimedia (imágenes, audio y video).

Hipertexto es una idea que fue introducida en la década de los setentas por Ted Nelson, de hipertexto se hablará en el siguiente capítulo, su idea es la siguiente; un documento en hipertexto es un documento capaz de proveer ligas visibles hacia otros documentos, un ambiente de computadoras con hipertexto se crea al seleccionar una liga en un documento y moverse directamente hacia otra máquina.

Con respecto a Internet, es importante aclarar que el Web es el concepto surgido de mantener ligas en un sistema de red, pudiendo ser ésta una internet (es decir un red de área local) o intranet como más recientemente se ha nombrado, mientras que el World Wide Web se refiere a este mismo concepto pero ya bajo Internet (la red de redes), cabe hacer la aclaración de que el World Wide Web no es Internet, es sólo el foco de interés para todos aquellos usuarios de esta red.

Al igual que cualquier servicio de Internet, para utilizarlo se debe contar con el programa cliente o con el servidor, en el caso de tener el cliente lo que realmente trabaja es un software que en la terminología del Web es conocido como "examinador" (browser) o bien "navegador", entre los examinadores más conocidos y más utilizados se encuentran Mosaic y Netscape, éste último realmente vino a

sustituir a Mosaic y es el "navegador" utilizado aproximadamente por el 80 % de los usuarios de Internet.

Los examinadores pueden acceder archivos a través de FTP, NNTP (The Internet Network News Protocol) e incluso hasta buscar documentos en una base de datos.

Pero, realmente ¿Qué es el WWW? Es un intento por organizar toda la información en Internet, a manera de un conjunto de documentos en hipertexto, y la forma de recorrer la red es a través de vínculos.

Los vínculos son palabras que dan paso a la conexión con otros servidores o bien que permiten abrir otros documentos.

El Web proporciona una interfaz uniforme para diferentes tipos de servicios, el modelo de hipertexto del Web le permite seguir un vínculo y realizar una búsqueda no importa qué tipo de recurso se esté utilizando. Además de que el Web elimina la barrera entre los datos del usuario y los "datos públicos". Si se configura un servidor WWW y un editor de hipertexto adecuado, se pueden integrar notas personales (aunque se pueden tener vínculos con documentos públicos).

II.3 Características y funcionamiento

La forma en la que el Web trabaja se puede explicar a partir del protocolo de transferencia que utiliza HTTP (Hypertext Transfer Protocol), el cual realiza cuatro fases claves:

- Conexión
- Petición
- Respuesta
- Desconexión

En la etapa de conexión la aplicación cliente (por ejemplo Mosaic o Netscape) pretende conectarse con el servidor, esto aparece en la barra de estado de la mayoría de los "navegadores" *'Connecting to HTTP server'* si el cliente no puede ejecutar la conexión se manda un mensaje explicatorio.

Una vez que se estableció la conexión con el servidor HTTP el cliente manda una petición al servidor. La petición especifica cuál protocolo será utilizado y qué objeto es el que será utilizado, el protocolo puede ser desde luego el mismo HTTP, más FTP y NNTP (Network News Transfer Protocol), Gopher o WAIS, esto es conocido como método de acceso y se realiza a través de los URL (Uniform

ResourceLocator) que consisten de una cadena de caracteres que identifica de manera única algún recurso; el formato básico para los URL es el siguiente:

método de acceso: //host:puerto/ruta de acceso al recurso

método de acceso : Puede ser alguno de los ya mencionados HTTP, FTP, Telnet, NNTP, Gopher.

host: es la computadora en la cual reside el recurso.

port: Es un número particular que identifica el servicio que se está pidiendo con respecto al servidor, éste se provee si el servicio está instalado en un puerto diferente al puerto estándar para ese servicio.

ruta de acceso al recurso: es la identificación de la localización

Un ejemplo sería:

`http://www.cecafi.unam.mx/licenciatura/fixh.html`

en la cual se hace referencia a un servidor de HTTP y además también especifica el método de acceso al archivo, el nombre del servidor viene después de las dos antidiagonales, después del nombre del servidor se puede especificar el nombre del archivo en HTML (HiperText Markup Language) "Lenguaje de marcado de hipertexto" que se va a desplegar separando con antidiagonales la ruta de directorios que debe seguirse para encontrar el archivo.

Así en el ejemplo anterior tenemos que se va acceder al archivo /licenciatura/fixh.html a través de un servidor HTTP llamado www.cecafi.unam.mx.

Como se mencionó, un examinador puede desplegar archivos a través de otros recursos que Internet proporciona por ejemplo FTP, así un archivo disponible a través de un servidor FTP y encontrado por un examinador tendría un nombre a través del URL llamado :

`ftp://cozumel.fi-a.unam.mx/usuarios/unicafi/claus/htdoc.html`

donde nuevamente FTP indica el tipo de servidor como de acceso al documento /usuarios/unicafi/claus/htdoc.html que se encuentra en el servidor de FTP anónimo cozumel.fi-a.unam.mx.

II.3.1 Mosaic

Mosaic fue diseñado como un examinador del World Wide Web, presenta una interfaz multimedia para Internet. Realiza más que la presentación de hipertexto

convinculos hacia otros documentos, en realidad es una herramienta de hipermedia, lo cual quiere decir que puede manejar audio, imágenes y video (imágenes en movimiento) y reúne casi todas las herramientas antes mencionadas en un solo paquete. en ocasiones las herramientas especializadas realizan una mejor tarea, pero si se desea instalar sólo un software de navegación los examinadores de Internet son la mejor solución.

A continuación se muestra la pantalla principal y de bienvenida de Mosaic, a través de la cual se explicarán sus funciones.

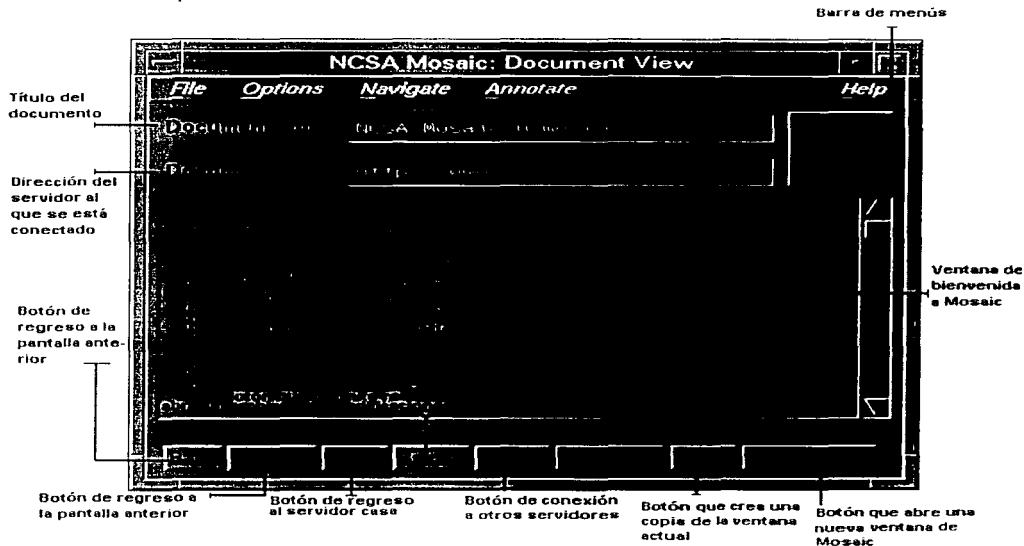


Figura que muestra la pantalla principal de Mosaic

Barra de menús: Aquí se muestran todos los menús, estos son desplegables, cuando se mueve el ratón a uno de estos menús y se mantiene oprimido el botón izquierdo, se obtiene la lista de opciones, se mueve el cursor hacia el comando a elegir para ejecutar esa acción sobre Mosaic.

Document Title (Título del documento): Título a través del cual se permite realizar búsquedas a través de todos los servidores http para localizar un tema en especial.

Document URL (Dirección del documento al que se está conectando): En realidad aquí se proporciona el URL, es decir, la cadena que identifica de manera única el recurso al cual se quiere acceder.

En la parte baja de la pantalla se localiza un conjunto de botones. Son comandos de un solo paso, sin menús; son las instrucciones más comunes.

Back: Botón de regreso a la pantalla anterior, sea ésta una página diferente o parte de la misma.

Forward: Botón para avanzar a la siguiente página.

Home: Botón para ir a la página casa, es decir al primer documento abierto, siendo que éste puede ser un documento en un servidor remoto o en nuestro servidor de HTTP.

Reload: Botón que nos permite visualizar los nuevos cambios hechos sobre alguna página o bien nos permite “refrescar” la pantalla.

Open: Botón para conectarnos a otros servidores, se realiza a través de un URL.

Save as: Opción que nos permite guardar en nuestra máquina local los archivos visualizados en las páginas, con diferentes formatos, guardar sólo el texto, o bien con el formato HTML, o poscript.

Clone: Nos permite visualizar otra pantalla igual a la que tenemos abierta; es muy útil realizar esto sobre todo cuando se requiere hacer búsquedas por distintas rutas de búsqueda sin perder un punto en común.

New window: Nos proporciona una ventana más de Mosaic, ésta es una ventana independiente de la primer ventana que se ejecutó.

Stop: El botón que aparece en la parte superior derecha de la pantalla que muestra un mundo girando es una herramienta que nos permite detener la acción de transferencia que se esté realizando y que por demorarse demasiado deseamos abortar.

En el filo derecho de la pantalla se puede ver la barra de desplazamiento, que se utiliza para desplazarse por documentos extensos que ocupan más de una pantalla. Al hacer clic sobre la parte más baja de la barra de desplazamiento el documento avanza hacia abajo; al hacer clic sobre la parte superior realiza la acción contraria. Para los casos en los que el documento es demasiado ancho, Mosaic cuenta con una

barra de desplazamiento horizontal, colocada en la parte baja de la pantalla de igual funcionamiento que la barra vertical.

Como todos los examinadores Web, Mosaic empieza encontrando una página de presentación en algún lugar de Internet. A menos que se configure nuestro cliente esta página de presentación será la de Mosaic que viene del Centro Nacional de Supercómputo en la Universidad de Illinois (donde nació Mosaic). Es posible crear nuestra propia página de presentación o que tome alguna de las muchas que existen en diferentes lugares de la red. La página de presentación está conformada por texto e imágenes. El texto subrayado es un vínculo hacia otro documento: si se da un clic en texto subrayado con el botón izquierdo del ratón, Mosaic obtiene y despliega un documento nuevo, incluyendo las gráficas que contiene.

En Mosaic las palabras vínculo aparecen subrayadas o bien en otro color demostrando precisamente que son vínculos, al volver de una página a la cual se accedió a través de un vínculo se notará que el vínculo aparece en un color distinto al que tenía originalmente, esto es una forma de indicarnos que ya hemos abierto ese vínculo, pero no significa que no podamos volver a visitar la página que abrimos con él.

Mosaic nos ofrece otros tipos de herramientas como lo son la de "bajar" información de Internet, esto quiere decir guardar la información que se está visualizando, en archivos que podemos ubicar en nuestra máquina local.

También podemos hacer búsqueda de palabras sobre un documento abierto, esto es, encontramos un archivo de nuestro interés pero es muy largo y sólo nos interesa una parte, desde el menú de file podemos utilizar la herramienta "search in current" con la cual podemos saber si el documento contiene alguna palabra en específico ya que nos pregunta por una cadena de entrada que va a ser la palabra a buscar.

El hecho de "navegar" puede hacerse tedioso, si por ejemplo, encontramos un lugar al cual dirigirnos siempre pues sabemos que ahí encontramos todo lo que buscamos y para poder acceder a ese lugar tenemos que hacerlo a través de una serie muy larga de vínculos. Para evitar esto Mosaic cuenta con una lista de acceso directo, "hotlist", esto es una lista donde se encuentran las direcciones (URL) de los recursos a los que más frecuentemente accedemos, de tal forma que cuando queremos ir a ese recurso basta con listar nuestra "hotlist" e indicar con el ratón a qué dirección queremos conectarnos, esta lista se puede editar, es decir, podemos agregar o quitar direcciones.

Mosaic permite hacer anotaciones sobre los documentos que se están visualizando, así cada vez que se intente abrir un documentos y se encuentra con alguna referencia de anotación las despliega como un pie de página, de tal manera que podemos ver los documentos de forma personalizada.

Existen algunas páginas que fueron diseñadas para ayudar en la búsqueda de información a través del Web, estas páginas cuentan con herramientas de búsqueda precisamente sobre todas las páginas creadas y que son accesibles vía un examinador, sin embargo, esto es una limitante de Mosaic, ya que no cuenta con esta herramienta integrada, es quizá una de las principales diferencias que han hecho del examinador Netscape el software de Web más utilizado y del cual hablaremos a continuación.

II.3.2 Netscape

Netscape, como ya se mencionó, es el software más utilizado por los usuarios del Web y en general por todos aquellos que tienen una conexión a Internet.

A continuación se mencionan algunas de las características que le han hecho tan famoso:

Cache de disco: Esto es utiliza cierto espacio en disco para guardar temporalmente las últimas páginas visitadas en la última sesión establecida de tal forma que al entrar nuevamente a Netscape y querer regresar a ellas no se tenga la necesidad de rehacer la conexión, es decir, funciona como una memoria cache permanente, con esto, también permite disminuir el tráfico en la red, aunque en el espacio en disco reside también su principal desventaja, aunque claro el espacio en disco es menor que la memoria, por lo que a pesar de todo sigue siendo muy costoso.

Acceso a ligas de documentos que ya han sido revisados: Cuando un documento está siendo traído desde una máquina remota, éste no se visualiza hasta que esté completo, y no se pueden utilizar los vínculos ni ver las imágenes, por lo menos esto es lo que hacían los examinadores comunes; sin embargo, Netscape no hace esto, si una página es demasiado grande, se visualiza poco a poco, pero aún y con eso ya se pueden utilizar sus vínculos e ir viendo poco a poco sus imágenes, esto es importantísimo sobre todo considerando a aquellos usuarios que se conectan vía módem utilizando la línea telefónica, no tiene que esperar ya tanto para poder ver en sus monitores las imágenes y ligas y seguir "navegando". Con respecto a las imágenes Netscape permite que se vea la imagen conforme se está "cargando" en la pantalla en lugar de esperar hasta que se haya transferido toda para presentarla, esto es muy importante para el usuario, ya que el hecho de ver qué imágenes están

“cargándose” da la impresión de que la máquina continúa trabajando, a diferencia de que si no ve nada pensaría que hay que reiniciar la máquina o por lo menos salir del software y volver a intentar la conexión.

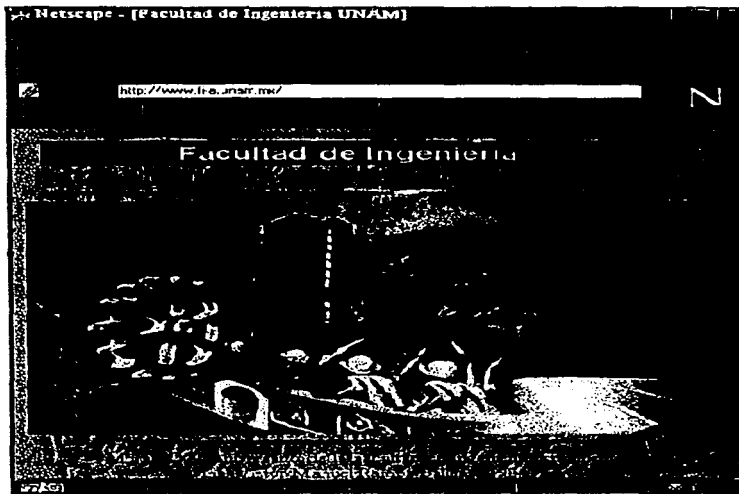
Soporte para etiquetas Netscape no estándar: Netscape fue diseñado para desplegar páginas HTML estándar, pero puede desplegar algunas páginas hechas en HTML extendido o mejorado, esto ha causado furor entre los desarrolladores de páginas ya que desde la versión de Netscape 2.01, las imágenes se pueden centrar, pueden ser sensibles, se pueden ejecutar animaciones, podemos ejecutar videos con audio y sonido, tener distintos tipos de letra, tamaños, tablas, etc., en el formato del texto desplegado, ahora existe un lenguaje llamado JAVA que permite todas estas posibilidades y aún de ejecutar aplicaciones desde una página.

Características e interfaces a través de plataformas o interfaces gráficas: Existen realmente muy pocas diferencias entre las versiones para X Window, Microsoft Windows y Macintosh, mientras que las versiones de otros navegadores para cada una de estas plataformas son sustancialmente muy diferentes, esto es importante ya que los usuarios pueden utilizar el navegador en distintas máquinas y lugares sin que exista diferencia entre las acciones que tiene que realizar para ejecutar una tarea o sin que visualmente sea diferente.

Capacidades para leer y “postear” hacia grupos de interés (Newsgroups): Los examinadores del Web tienen la posibilidad de acceder a grupos de interés, mientras que Netscape ofrece un gran conjunto de grupos de interés, se pueden leer sus artículos, mandar artículos (postear), enviar mails, etc.

Capacidades de seguridad incluidas: La seguridad de los documentos del World Wide Web es una característica que ha tenido especial interés ya que se pueden transferir datos privados y bastante delicados como pueden ser números de tarjetas de crédito para transferencias financieras y Netscape es el primer examinador en crear, distinguir y aplicar seguridad a las páginas. Esta seguridad la pueden ofrecer los servidores de Web, aún así Netscape te avisa cuándo se está haciendo una transferencia de información que puede ser delicada.

A partir de la pantalla de Netscape explicaremos sus funciones.



Barra de títulos: En la parte superior Netscape muestra el título que tiene la página que estamos observando.

Barra de Menús: Aquí tenemos una serie de menús disponibles al pasar con el puntero del ratón sobre alguno de ellos se despliega una lista de comandos que se pueden ejecutar eligiéndolos con el puntero del ratón y dando un clic sobre el botón izquierdo de éste.

Barra de herramientas: Son un serie de botones que nos permiten ejecutar las acciones más comunes en Netscape con sólo dar un clic izquierdo con el ratón sobre ellos, además es muy fácil identificar la acción que realizan ya que tienen asociada una imagen que sugiere la acción a desempeñar.

Back: Botón que permite regresar a desplegar la última página visitada.

Forward: Permite avanzar hacia la página siguiente, esta opción se activa una vez visitadas varias páginas.

Home: Botón que me permite regresar hasta la página que carga Netscape al iniciar una nueva sesión; esta página es conocida como HOME PAGE.

Edit: Esta es una herramienta muy novedosa con respecto a los demás examinadores del Web, permite editar “en línea” las páginas que se están visualizando, claro estas páginas a menos que sean las nuestras y que tengan los archivos de información e imágenes en nuestra cuenta local podrán ser modificadas y posteriormente visualizadas por todos, de otra forma primero tendremos que copiarlos a nuestra cuenta para más tarde modificarlos pero estas modificaciones sólo las veremos nosotros, esto es una muestra del grado de “personalización” que nos permite Netscape”.

Reload: Permite “refrescar” la pantalla, es decir, re-dibujar lo contenido en la página, ésta es una herramienta muy utilizada por aquellos creadores de páginas ya que con un “*reload*” pueden ver las nuevas modificaciones hechas a sus páginas sin necesidad de salir y volver a entrar de Netscape o de utilizar algún método de acceso a su página.

Imágenes: Permite ver una imagen en una página de manera individual, esto porque en algunas ocasiones las imágenes mostradas en algún documento no aparecen del tamaño real ya que no cabrían, al dar esta opción podemos verla en sus dimensiones originales.

Open: Permite acceder a un recurso dentro de Internet para ello debemos proporcionarle el URL del mismo.

Print: Permite mandar a imprimir el documento visualizado a la impresora definida para la máquina local.

Find: Herramienta que me permite localizar alguna palabra dentro del documento visualizado.

Stop: Esta herramienta me permite detener alguna acción relacionada con la transferencia de archivos, como una conexión o bien el guardar algún archivo, ya que estas operaciones pueden llegar a ser demasiado largas, podemos abortarlas al dar un clic sobre este botón.

Location: Dirección en URL del recurso que estamos accediendo.

Barra de novedades: Ésta es una de las principales innovaciones de Netscape; nos brinda estos botones para mostrar lo nuevo a través de Netscape, los grupos de

interés. los nuevos lugares a visitar, y la herramienta más utilizada por los usuarios de este examinador "Net Search", que no es otra cosa más que la selección de algunas de la páginas que han implantado una herramienta de búsqueda a través de los documentos en hipertexto a través de la red, la búsqueda se puede hacer por cualquier palabra que sugiera algo referente al tema del cual deseamos obtener información y los resultados de la búsqueda no es otra cosa que un documento en hipertexto con las ligas hacia los lugares de la red en donde existe la información que estamos buscando, entre los lugares más utilizados para estas búsquedas se encuentran páginas como las de Yahoo, Lycos, Excite, Infoseek, etc., que incluso cuenta con una clasificación de temas para realizar sus búsquedas de manera más eficiente.

Barra de estado: es una barra a través de la cual Netscape nos informa de las operaciones que se están realizando así como el estado de éstas, es decir, si ya se terminaron, si están estancadas; un ejemplo podría ser la transferencia de un archivo, aquí podemos ver la velocidad de transferencia del mismo así como el tiempo transcurrido.

Barras de desplazamiento: En el filo derecho e inferior de la pantalla aparecen dos barras de desplazamiento la horizontal y la vertical, las cuales permiten el desplazamiento por documentos extensos.

Guardando documentos: Con Netscape la transferencia de archivos hacia nuestra cuenta local se puede realizar de una manera muy sencilla desde el menú de FILE. con la opción de SAVE AS, la cual nos da la opción de "bajar" el archivo que estamos visualizando, y con formatos diferentes, es decir, me permite que tenga el archivo con formato HTML, con todas las etiquetas que le dan formato propiamente, o como sólo texto, en donde sólo se guarda la información sin ningún formato, en ambas opciones anteriores si un documento contiene una imagen ésta no es almacenada, sólo se mantiene la referencia que se hace dentro del texto, sin embargo con la opción de salvar en formato poscript sí se almacena la imagen.

Por otro lado otra ventaja de Netscape con respecto a Mosaic es precisamente la opción de poder guardar la imágenes de manera separada con respecto al documento en donde se encuentra inmersa, esto es, nos colocamos sobre la imagen y con el botón derecho del ratón dando un clic sostenido aparecerá un menú colgante en donde se encuentra la opción SAVE THIS IMAGE, de tal forma que al seleccionarla con un clic derecho del ratón nos pregunta por la ruta y el nombre con el cual guardaremos la imagen, los formatos de imágenes más comunes son .GIF y .JPG, que son los formatos que permiten por sus algoritmos de almacenamiento

(vectorial), que las imágenes se “carguen” más rápidamente y que permite la movilidad sobre la páginas aún y cuando no terminen de desplegarse éstas.

Búsquedas en documentos: Se pueden realizar búsqueda de palabras claves sobre un documento, es decir, una vez que se localiza un documento de interés, y éste es un documento muy extenso podemos buscar sólo lo que nos es de mayor utilidad, con la herramienta FIND.

Accesos más comunes: Existen lugares a los cuales accedemos de manera muy frecuente porque en ellos encontramos información de nuestro interés, estos accesos los podemos realizar ya sea a través de los ya conocidos vínculos o ligas, a través de una búsqueda con el Net Search o bien podemos almacenar las direcciones de estos lugares para posteriormente a través de la herramienta OPEN le digamos el URL del recurso requerido, o también, podemos tan sólo dar un clic sobre esta dirección y trasladarnos automáticamente a esta página, esto último es el principio de lo que Netscape llama BOOKMARKS, es decir, marcado de páginas, lo que hace es guardar en un archivo los URL de las páginas que le indiquemos a través del menú Bookmarks con la opción de ADD BOOKMARK, en este mismo menú en la parte de abajo nos muestra todos los Bookmarks anteriores y los podemos editar (agregar, borrar).

Opciones de configuración: Netscape es el primer navegador que brindan esta posibilidad, la de configurar la presentación de la información, podemos hacer que las fuentes (la letra), aparezca con un tipo en particular, por ejemplo: Script y con un tamaño de 10, o bien que aparezca en un color diferente, es decir que no tenemos que conformarnos con el diseño que tengan propiamente las páginas que visualizamos, sino que podemos verlas con los colores y formatos de fuente.

Podemos también permitir que utilice las ventajas ya mencionadas de Caché de disco o de memoria de acuerdo a las posibilidades de nuestra máquina con el fin de hacer más rápidas las transferencias de información.

Además, claro, de poder indicarle que al iniciar una sesión pueda “cargar” una página en particular, en alguna parte de la red, o bien nuestra propia página.

Opciones de directorio: Podemos configurar las rutas que nos ofrece por *default* para guardar los archivos tanto de imágenes como de texto, de tal forma que yo le puedo indicar cuál es mi directorio de trabajo, de tal manera, que cuando vaya a guardar alguna información no me preocupe por indicarle la ruta.

Podemos tener abiertas múltiples ventanas de Netscape, esto para cuando realizamos búsquedas a través de vínculos por distintas rutas no perdamos el punto de partida inicial.

Capítulo III

Capítulo III El hipertexto

III.1.- ¿Qué es hipertexto?

Como ya se mencionó en capítulos anteriores el concepto de hipertexto es aparentemente muy simple, se utiliza la capacidad de almacenamiento y búsqueda de una computadora para ligar documentos entre sí y poder acceder a cualquiera de ellos no importando en cuál de ellos estemos situados.

Un hipertexto es un conjunto de documentos, cada uno de los cuales tiene ligas visibles a cada uno de los demás documentos que forman el conjunto. Una liga usualmente es una palabra en un color diferente que indica precisamente que es eso, un vínculo hacia otro documento.

En el surgimiento del hipertexto existen dos personas que han sido fundamentales: Vannevar Bush que fuera el asesor científico del presidente Roosevelt y quien tenía el propósito de crear un sistema capaz de conectar toda la información científica de la pos-guerra ya que estaba creciendo tanto que no sabían ya quién investigaba algo que pudiera interesarles; la idea era colocar toda la información de investigaciones incluyendo diagramas, imágenes y notas orales, aunque no lo pudo realizar para su tiempo fue una gran idea; por otro lado Ted Nelson hasta la década de los sesentas recuperó esta idea con el fin de crear un mecanismo capaz de imitar la habilidad del cerebro humano para almacenar y recuperar información al realizar asociación de ideas, aunque éste aún no se ha podido realizar ya que no existe una máquina capaz de realizar estas operaciones, un paso ya se ha dado a través del World Wide Web.

El hipertexto en su forma básica es un sistema manejador de bases de datos que permite conectar diferentes entidades de información por medio de ligas asociativas, dando como resultado que un sistema de hipertexto es, en sus formas más complejas, un ambiente de procesamiento de información para trabajo de grupos, comunicaciones y adquisición de conocimiento.

La unidad fundamental de información en un documento de hipertexto es llamado nodo. Idealmente, un nodo es una porción del documento que cubre un concepto. Un nodo puede caber en una pantalla de computadora, o bien puede ser tan pequeño como un palabra o tan extenso como un libro. Un nodo puede contener una combinación de texto, gráficas y/u otras formas de información.

Los nodos se conectan entre sí a través de referencias cruzadas denominadas ligas. Para obtener más información acerca de un tema en especial, el lector de

hipertexto simplemente selecciona una liga, lo cual trae a la pantalla el contenido del nodo al cual hace referencia dicha liga. El número de ligas es variable dependiendo del contenido de cada nodo. Algunos están relacionados con muchos otros, por lo cual contienen muchas ligas, mientras que otros pueden servir sólo como documentos o destinos finales.

Bush había mencionado que sería necesario almacenar más que texto en cada uno de los nodos. Él estaba interesado en almacenar mensajes orales. En la actualidad, los sistemas de hipertexto incorporan medios de información adicionales al texto, como son gráficas, sonido, audio y vídeo, estos sistemas son conocidos como *hipermedios*.

En general, un documento tradicional en forma impresa o electrónica es un documento secuencial, es decir, lleva una secuencia lineal que determina el orden en el cual aparecerá o, será leído, mientras que el hipertexto no es secuencial, no existe un orden fijo de cómo se leerá la información. Dependiendo del nodo que estemos accediendo, las posibilidades de abrir otros documentos y el orden no están establecidos ya que podemos seleccionar cualquiera de las ligas para continuar. Es decir el lector decide qué trayectoria seguir.

El autor del hipertexto muestra una serie de alternativas y debe contemplar siempre aquella posibilidad de regresar al documento del cual se está partiendo.

En desarrollo de sistemas de hipertexto ha sido vasto desde su concepción en 1945 por Bush, hasta su implantación en los años 60 con Nelson y otros investigadores, con lo cual se quiere decir que realmente no es nada nuevo, simplemente que ha cobrado importancia por la magnitud que ha tomado, el sueño de Nelson de tener una gran biblioteca que guardarla la información literaria no importando la máquina en la que estuviera y a la cual se accediera de manera transparente al usuario ya no es un sueño, el concepto "web" que surgiera en uno de los tantos proyectos de hipertexto toma formas gigantescas en el World Wide Web y es ya una realidad.

III.1.1 Funcionamiento de un modelo de hipertexto

En general un modelo de hipertexto tiene definidos los siguientes niveles:

- 1.- Nivel de base de datos
- 2.- Nivel de hipertexto
- 3.- Nivel de presentación

El nivel de base de datos está íntimamente ligado con los conceptos ya mencionados como lo son: el almacenamiento y la recuperación de información, funciones realmente independientes de lo que es el hipertexto.

En sí, para un sistema de hipertexto es necesario el almacenamiento de grandes cantidades de información en múltiples medios de almacenamiento como lo pueden ser los discos magnéticos, ópticos, etc., y debe ser capaz de recuperar esta información ya sea completa o en una porción. Cada nodo es un objeto sin ningún significado particular.

El nivel de hipertexto es la parte media del modelo, el vínculo entre el nivel de base de datos y la interfaz con el usuario, es aquí donde se establecen las relaciones entre los nodos a través de las ligas.

El nivel de interfaz de usuario es donde se presenta la información con el formato en hipertexto y debe incluir los comandos que el usuario puede ejecutar para mostrar nodos y ligas. Incluyendo la forma en que estas opciones se muestran al usuario ya que si se tienen definidas desde el nivel de hipertexto diferentes tipos de ligas le corresponde al nivel de interfaz de usuario definir de qué manera se presentan al usuario para que se distinga la diferencia.

Hablando un poco más acerca de los nodos, que como ya mencionamos son la unidad fundamental del hipertexto, existen dos grupos de ellos; los nodos basados en ventanas y los nodos basados en marcos o "*frames*".

Los nodos basados en marcos definen en términos generales que el tamaño del nodo debe estar de acuerdo a la cantidad de información visible en una pantalla de computadora, sin que en realidad sea un estándar ya que existen sistemas de hipertexto que definen el tamaño a una carta, con esto se puede ver que para en el caso de tener más información que la del tamaño definido entonces tendría que fraccionarse en varios nodos, haciendo realmente tediosa la labor de presentar la información.

Por otro lado los nodos basados en ventanas no tienen este tipo de problemas, aunque es cierto que la información visible está sujeta al tamaño de la ventana, ésta tiene un mecanismo de desplazamiento, "*scroll*" que permite desplazarse sobre todo el documento así sea muy extenso.

Sin embargo, ¿qué es lo óptimo? Tener muchos nodos pequeños o un nodo muy grande: esto sin duda es decisión del autor, sin embargo, por "navegabilidad", resulta más rápido acceder a nodos pequeños y procesar su información que en

nodos grandes. Esto es algo que agradecen mucho los usuarios del sistema hipermedio más utilizado "World Wide Web".

Las ligas son la otra unidad fundamental del hipertexto, las ligas casi siempre parten de un punto determinado dentro del nodo para mostrar al usuario un objeto (recurso) explícito, con esto se tiene que el hipertexto juega una doble tarea al ser información desplegable y al ser un punto de partida hacia otro nodo.

La mayoría de las ligas son explícitas, es decir, que alguien las colocó ahí para conectar un nodo con otro. Sin embargo existen las ligas implícitas que son ligas no definidas como tales, pero que existen de acuerdo a propiedades de la información.

Existen también las llamas superligas, es decir que una liga tenga conexión con más de un nodo de tal forma que al elegirla nos mostrará un menú de los documentos a los cuales podemos dirigirnos y podemos elegir el que nos convenga más.

Fundamentalmente existen dos formas para definir ligas en un hiperdocumento:

1.- Comandos incorporados al texto: Este sistema consiste en definir las ligas a través de comandos especiales para el efecto, que se intercalan dentro del texto del documento. Una vez creado, el documento se somete a un procesamiento que interpreta los comandos y presenta al lector el hiperdocumento en su formato final, de tal forma que cuando esté activa una liga, el sistema sabe a qué nodo dirigirse para presentarlo. Existen dos tipos de comandos: los procedurales y los declarativos. Los procedurales especifican qué debe hacer una liga: la siguiente palabra es una liga al nodo X. Los declarativos especifican qué es un liga: la siguiente línea es un título de capítulo y definen una serie de reglas y convenciones para ligar un título de capítulo a diferentes párrafos.

2.- Manipulación directa: Este método consiste en establecer las ligas directamente sobre el texto, el autor selecciona un texto con el cursor o con un ratón, luego crea sobre él una liga y define el documento destino.

Este segundo es el más utilizado debido a su facilidad de creación así como de menor posibilidad de error por parte de los usuarios.

III.2.- Relación con lenguajes y protocolos

El hipertexto tiene una estrecha relación con los lenguajes de programación o bien con los procesadores de texto ya que se basa en el formato del texto, o de los objetos que se despliegan en los examinadores del web, todos los examinadores del Web deben reconocer una versión de HyperText Markup Language (HTML) Lenguaje de Marcado de Hipertexto llamado nivel 0, existen especificaciones adicionales de HTML incluidas en el nivel 1 que permite manipular formatos de texto y de imágenes, el nivel 2 de HTML define tipos de formas que no todos los examinadores pueden reconocer, el nivel 3 incluye los niveles anteriores así como la opción para manipular figuras, tablas, ecuaciones matemáticas, hojas de estilo y algunas otras herramientas de control de distribución.

HTML es similar a un lenguaje de programación ya que requiere que se exprese de manera específica la estructura en favor de que lo entienda la computadora, y en particular para que lo entienda el examinador de hipertexto, aunque no es tan complicado como un lenguaje de programación, escribir en HTML se requiere de reglas para dar formato o marcar las partes del documento. El propósito de HTML es el contrario a un procesador de texto WYSIWYG (What You See Is What you Get) que se concentran en la apariencia del documento, mientras que HTML se concentra en lo que es la estructura.

HTML está definido por el Estándar General del Lenguaje de Marcado SGML "Standar General Markup Language" que es en realidad un estándar internacional (ISO 8879) para procesamiento de texto e información. SMGL en sí mismo es un metalenguaje (un lenguaje que define lenguajes). El acierto de SMGL es el de proporcionar los formatos de ayuda en línea para distribución de información electrónica, búsquedas y respuestas en el sentido en que no son tan importantes los detalles de apariencia del documento, es como tener un lugar para almacenar la información y en realidad quien tiene que preocuparse por su apariencia es el "despliegue".

Por ejemplo una lista ordenada en HTML está definida por etiquetas, no por una serie de números, de tal forma que el examinador puede desplegar esta lista utilizando números arábigos o bien números romanos.

III.2.1 Partes de un documento HTML (niveles 0 y 1)

Un documento en HTML luce como a continuación se muestra:

```
<HTML>
<HEAD>
<TITLE>Facultad de Ingenier&iacute;a<TITLE>
</HEAD>
<BODY>
<H1>Algunas de las divisiones que la integran</H1>
<HR>
<UL>
<LI><A HREF="http://www.cecafi.unam.mx/die.html"> Divisi&oacute;n de
Ingenier&iacute;a El&eacute;ctrica</A>
<LI><A HREF="http://www.cecafi.unam.mz/dcb.html">Divisi&oacute;n de
Ciencias B&aacute;sicas</A>
</UL>
<HR>
<ADDRESS>
Para mayores informes comun&iacute;cate con:
<A HREF="http://www.cecafi.unam.mx/~claus/htdoc.html"> Claudia
Cordero</A>
<A HREF="mailto:claus@cancun.fi-a.unam.mx">
</ADDRESS>
</BODY>
</HTML>
```

Un documento HTML está compuesto de una serie de elementos que definen su estructura, a continuación se darán las definiciones y detalles de cada uno de ellos. Las etiquetas de estos elementos no son sensitivas a las mayúsculas o minúsculas, un examinador puede interpretar de igual forma la etiqueta "title" que la etiqueta "TITLE".

Existen principalmente tres tipos de elementos: encabezado (head), cuerpo (body) y gráficos (graphics); a continuación se detallan cada uno de ellos.

1.- Elementos de encabezado : Los elementos de encabezado son utilizados para marcar propiedades en cualquier parte del documento, como por ejemplo en el título, ligas que indican la relación entre un elemento y otro, así como el URL del documento. Esta información no es desplegada como parte del documento en sí, pero es parte de la información del documento que utilizan los examinadores en

varios sentidos. (Para localizar el documento en una búsqueda, para acceder a otro servidor a través de su URL, etc.) . Ejemplo:

```
<TITLE>Facultad de Ingenier&iacute;a</TITLE>
```

2.- Elementos de cuerpo (Body): Los elementos de cuerpo son utilizados para definir propiedades en el cuerpo del documento. A diferencia de los elementos de encabezado, los elementos de cuerpo sí son reconocidos por el examinador en el momento de leer el documento, algunos elementos de cuerpo son los siguientes:

a. Headings: Son utilizados para marcar divisiones en un documento. marcan el tamaño del texto que se está escribiendo. Existen seis niveles.

```
<H1></H1>  
<H2></H2>  
<H3></H3>  
<H4></H4>  
<H5></H5>  
<H6></H6>
```

b. Anchors: Se utilizan para marcar el principio y el fin de ligas de hipertexto. Estas ligas se refieren a otros documentos, otros recursos en la red (se determinan a través de su URL) o como otra parte del mismo documento.

Ejemplo:

```
<A HREF="http://www.cecafi.unam.mx/~claus/htdoc.html">Claudia  
Cordero</A>  
<A HREF="mailto:claus@cancun.fi-a.unam.mx"></A>
```

c. Marcas de Párrafo: Define un salto de párrafo, es decir marca un salto dejando espacio entre un párrafo y otro.

Ejemplo:

La marca de párrafo usualmente puede ser expresada por un examinador en cualquier parte, usualmente, sin embargo, un examinador deja una línea en blanco entre párrafos. <p>

d. Saltos de línea: Indica el inicio de una nueva línea, sin una marca de salto de línea, todo el texto a menos de que exista una línea en blanco (indicadas en

HTML) aparecerá en un solo párrafo. Cualquier espacio en blanco, así como tabuladores o retornos de línea (RETURN) son ignorados.

Ejemplo:

Facultad de Ingeniería;
Circuito Interior

Ciudad Universitaria

aparecerá:

Facultad de Ingeniería
Circuito Interior
Ciudad Universitaria

e. Horizontal rule: Indica que el examinador en caso de encontrar una línea de texto larga tratará de ajustarla al ancho de la ventana de despliegue.

f. Etiquetas de dirección: <ADDRESS> y </ADDRESS> son utilizadas para indicar quién es el autor del documento, con una liga al *home page* del mismo, así como el de su dirección electrónica.

Ejemplo:

<ADDRESS>

Para mayores informes comunicate con:

 Claudia Cordero

</ADDRESS>

g.- Blockquote style: <BLOCKQUOTE></BLOCKQUOTE> se utilizan para indicar una cita textual referenciada en algún otro documento.

Ejemplo:

<BLOCKQUOTE>

Antes de querer dominar a la naturaleza, hay que aprender a obedecerla.

Umberto Eco </BLOCKQUOTE>

h.- List: Utilizadas para generar listas de definiciones (glosarios), listas ordenadas, menús; las partes de una lista se denotan con la etiqueta (List Item).

Ejemplo:

 (Unsorted List)
 Definition List
 Unordered List
 Ordered List
 Menus

i.- Graphics. Algunos *browsers* pueden desplegar gráficos en línea, es decir de manera simultánea a los textos que se despliegan. La manera de indicar que se despliegue una imagen dentro de un texto determinado, o bien de manera independiente es a través de la etiqueta:

```
<IMG SRC="nombre_imagen.gif">Texto_al_pie_de_la_imagen
```

IMG: Indicativo de imagen

SRC: Ruta y nombre del archivo de imagen, la mayoría de las imágenes que nos encontramos dentro de los documentos a lo largo de la red son imágenes .gif o bien .jpg, esto debido al algoritmo de almacenamiento y despliegue de estas imágenes, que permite presentar el texto del documento en general aún y aunque la imagen no sea visible, esto debido a que siempre es más lento el desplegar una imagen que texto normal.

Texto_al_pie_de_la_imagen: Etiqueta que aparece al final de la imagen.

Ejemplo:

```
<IMG SRC = "facultad.gif">Escudo Facultad
```

J.- Entities. Para poder escribir caracteres especiales HTML se auxilia de expresiones, por ejemplo: si quisieramos escribir un ">" (mayor que), puesto que es un carácter que está asociado a todos los fines de instrucciones en HTML para poder escribirlo debemos decirle al *browser* que no está cerrando ninguna instrucción, y para desplegarlo le decimos que lo haga a través de la expresión ">" (grather than").

Otro ejemplo sería el de escribir acentos, para ello debemos determinar cuál será en este caso la vocal a acentuar y agregar la siguiente expresión:

&vocalacute;

Ejemplo:

á ---- á
é ---- é
í ---- í, etc.

Mientras que si lo que queremos es escribir la “ñ” o “Ñ”;

ñ ---- ñ
Ñ --- Ñ

O bien podemos hacer referencia al carácter que queremos escribir a través de su correspondiente código ASCII, de la siguiente manera:

&#número_ASCII

Ejemplo:

& = (ampersand)

Dentro de las características de HTML existen herramientas avanzadas entre las que se encuentran la generación de formas, que entre otras cosas permiten la interacción con el usuario permitiéndole la entrada de información, o bien están los mapas sensitivos, que son imágenes sensibles al contacto con el ratón y al oprimir éste en alguna sección de la imagen puede invocar a otros documentos, es decir, una imagen puede ser una liga.

Common Gateway Interface(CGI) Es la llave que nos permite conectar al Web con otro software y con bases de datos y realmete es un traductor dígase archivo script, programa ejecutable o de intercambio de información a través de una interfaz con otro software.

Los *scripts* son archivos de comandos principalmete UNIX generados bajo cualquiera de los *shells* que éste maneja ya sean Korn Shell, Bourne, así como otros lenguajes como Perl, etc.

Mientras que los programas *Gateway* son *scripts* o archivos ejecutables escritos para ser utilizados en un servidor de web como una interfaz traductora y pueden ser archivos ejecutables compilados en C o C++.

Estos programas interactúan con HTML a través de etiquetas que van desde el SCRIPT, hasta el ISINDEX que es un método de búsqueda dentro de cajas de texto de las formas que se pueden generar. Se pueden generar scripts que accedan

a una base datos, generen una consulta y sus resultados los reflejen en una forma dentro de una página (propósito principal del trabajo). Los CGI pueden sin embargo generar vulnerabilidad con respecto al servidor Web debido a que para poder utilizarlos deben estar incluidos en directorios de acceso común así como también debemos de tomar en cuenta que son archivos que pueden acceder de manera directa a los comandos del sistema operativo que se maneje dentro del servidor Web y pueden provocar filtrados en cuanto a la seguridad de la información de éste, así como de su configuración en determinado momento.

Otra característica con respecto a los CGI es que ocupan tiempo de procesador con respecto al servidor Web, cada vez que se accede a un CGI, éste intenta ejecutarse en el CPU del servidor, por lo cual éste puede llegar a alentarse demasiado, a menos que los programadores de estos *scripts* y en general de los programas lo hagan de manera que éstos se ejecuten en otra máquina para no cargar de tabajo al servidor de Web.

Pasos para utilizar un CGI y cómo actúa éste dentro de un servidor Web:

1.- Instalación del *Gateway*:

a.- El programador crea el programa *gateway*, este programa puede ser cualquier *script* (ya sea en cualquier, *shell* de Unix o bien en Perl) o puede ser un programa ejecutable hecho en C o C++.

b.- El programador o el administrador del Web hace disponible el *gateway* para su ejecución desde el servidor, esto lo hace colocándolo bajo un subdirectorío llamado */cgi-bin/* dentro del servidor de Web o bien en el directorío especificado por la configuración del servidor.

c.- El programador de la aplicación *gateway* crea una liga hacia el programa en una página Web. Para ello utiliza el URL correspondiente a la localización del CGI asociado a la etiqueta HREF para incluir una liga.
Ejemplo:

```
<A HREF= "http://www.server.com/cgi-bin/prueba.pl">Prueba de un script de Perl</A>
```

2.- Utilizando el Programa *Gateway*

a.- El usuario selecciona la liga que activa el programa *gateway*. Esto a través de la etiqueta de HTML puesta con la asociación al archivo (el anchor).

b.- El usuario hace una petición a través del URL al servidor de Web, debido a que el *script* se encuentra en el servidor bajo el subdirectorio */cgi-bin/*.

c.- La salida de resultados que genera el programa *gateway* es pasado al examinador del usuario y es desplegado de acuerdo a la forma de despliegue establecida en la salida.

La principal diferencia entre acceder a un CGI o un archivo regular de HTML es que el programa CGI se ejecuta y pasa sus resultados al examinador del usuario, mientras que el archivo regular HTML sólo despliega su contenido a través del examinador del usuario.

La tendencia en general hacia la creación de las *Home page* está enfocada hacia la interacción con el usuario, a través de medios como las animaciones, el audio, las imágenes sensitivas, etc. Todo esto se ha generado a través de lenguajes especiales que han permitido conjuntar lo que es el hipertexto con estos tipos de "medios", en particular un lenguaje que ha causado sensación por lo que ha logrado con respecto al diseño de *home pages* es Java, que es un lenguaje de programación orientado a objetos perfeccionado por Sun Microsystems que proporciona animación e interacción con el World Wide Web. HotJava es el nombre del software generado por Sun.

Java se conecta con el HTML y Web a través de una etiqueta del HTML llamada APP, que le permite al creador o programador incluir programas especiales de Java llamados *applets* en las páginas Web. Estos *applets* básicamente son programas de software que el examinador del usuario baja (automáticamente, como parte de la página de observación) y ejecuta. Como le permite la entrada y salida de gráficas a través del *applet* en la página, Java puede manipular estas opciones permitiendo así una interacción más profunda con el Web.

Una ventaja más con respecto a las aplicaciones generadas con Java es que es hasta el momento en que el examinador despliega la información cuando se "compila" dicha aplicación y por lo tanto libera al servidor de Web de dicha acción. Es por ello que no todos los examinadores pueden desplegar *applets* de Java ya que no todos contienen el compilador.

III.3.- Creación de una Home Page

Para la generación de una *Home page* como se les llama a los documentos en HTML, que hacen referencia a la información personal o propia de alguna persona, empresa u organización que ha decidido ponerla a disposición de los usuarios de Internet, se requiere primeramente de la información, una vez teniendo ésta a través de las posibilidades que nos brinda HTML se genera el documento con las características que se deseen y finalmente se incluye en un servidor HTTP, que es un servidor que permite poner a disposición de los usuarios de la red (en este caso del Web) en particular, las *Home pages* que lo soliciten y cumplan con sus normas de creación, de información, etc.

En la actualidad existen ya muchas herramientas, díganse aplicaciones que permiten la generación del documento en HTML de manera más amigable que el hecerlo nosotros mismos desde un editor de texto y colocando las etiquetas adecuadas para obtener lo que descamos, éstos van desde simples editores que lo que nos ahorran es la molestia de colocar las etiquetas, ya que con un clic del ratón sobre alguno de sus botones de etiquetas se escriben sobre el documento, hasta verdaderas herramientas gráficas; que permiten generar imágenes sensitivas de una manera sorprendentemente rápida, incluir imágenes con movimiento, audio, acceso a bases de datos de manera natural o bien la generación de un banco particular de información, etc.

Lo que nos permite esto es una gran flexibilidad y múltiples posibilidades de generar nuestra *Home page* sin importar qué tantos son los recursos con los que contamos, los editores de HTML ya están la mayoría en ambientes *windows* y hasta hace poco tiempo no mostraban el documento tal y como de vería incrustado dentro de una liga dentro del *home page* principal de nuestro servidor HTTP, sin embargo ahora, esto ha cambiado y la mayoría de los editores a tratado de implantar la característica de la mayoría de las aplicaciones *windows* que tiene que ver con edición en general y que son WYSIWYG (What You See Is What You Get), "lo que tú ves en tu pantalla es lo que tú obtienes al final", de tal forma que al usuario le permiten ver una emulación de lo que se verá en el Web. De hecho, ahora las nuevas versiones de los *browsers* o navegadores ya incluyen un editor en línea que permiten modificar un *home page* de manera directa sobre el servidor, siempre y cuando se esté trabajando sobre él directamente y seamos los dueños de la *home page*, de no ser así podemos hacer una copia de los archivos HTML a nuestra máquina local, hacer los cambios que se consideren pertinentes y de esta forma personalizar las *home page* de otros de tal forma que se pueden acceder a la *home page* oficial, es decir, a la original o bien, a la nueva que se ha creado con los cambios que se le indicaron.

No existe limitación real para generar una *home page*; las limitantes las dan los recursos con los que se cuenta para generarlo, pero decimos que aún y cuando no contemos con una aplicación de diseño gráfico o con un editor de HTML podemos crear nuestra *home page* desde cualquier editor utilizando las etiquetas de HTML.

Dentro de las consideraciones generales para tener una *home page* podemos citar las siguientes:

- 1.- Incluir siempre un título de tal forma que los usuarios del Web puedan encontrar a través de herramientas de búsqueda propios de los examinadores, *browsers* o navegadores referencia hacia ésta.
- 2.- Incluir siempre una dirección a la cual referirse en caso de que la información mostrada sea de interés para algún usuario.
- 3.- Mantener ligas hacia nodos de trascendencia general (para no aislarse).
- 4.- Incluir imágenes en los formatos antes mencionados (.jpg,.gif) para no hacer lento el acceso a la página (los usuarios agradecen mucho la velocidad de acceso).
- 5.- Si se incluyen imágenes muy grandes no colocarlas en la página de inicio, sino de manera separada en un documento al cual se acceda a través de una liga.
- 6.- Generar páginas con código para usuarios que no tienen un examinador gráfico, o bien para aquéllos que no tengan un examinador capaz de desplegar las aplicaciones que incluimos.

Capítulo IV

Capítulo IV El hipertexto bajo ambiente de bases de datos**IV.1.- Servidores de bases de datos**

Para definir lo que es un servidor de base de datos debemos definir antes ¿Qué es una base de datos?.

Una base de datos es una colección de información contenida en tablas (estructura de almacenamiento; compuesta de columnas y renglones donde las columnas son los atributos generales de la entidad que representan cada uno de sus renglones, dígame, empleado, ciudad, pedido, materia, alumno, etc.), además de contener en sí misma algunos otros "objetos" que son elementos adicionales para el manejo de la información y la seguridad de la misma como lo son: las vistas, las consultas, los índices, los usuarios mismos de las base de datos, etc.

En general para que una base datos pueda mantener la estructura antes mencionada debe de existir una plataforma para soportarla, así como una herramienta para administrarla, estos son en ese orden, un servidor de base de datos y un sistema manejador de base de datos (DBMS) y en particular para el tipo de base de datos que definimos (relacional) sería un RDBMS (Sistema Manejador de Bases de Datos Relacionales).

Para poder soportar un manejador de estas características debe preverse la plataforma en la que se encontrará el sistema manejador de base de datos y después el lugar en el cual residirá dicho software, así como la base de datos misma.

Para ello deben considerarse los niveles de seguridad que mantendrá dicha base de datos, a nivel de plataforma estamos hablando del sistema operativo bajo el cual se instalará el software manejador de base de datos y desde aquí se puede imprimir seguridad en cuanto al acceso mismo, por el tipo de sistema operativo que manejemos: multiusuario o monousuario.

Una vez definido esto pasamos al nivel del RDBMS que de acuerdo al que utilicemos nos generará cuentas para acceder a él y a la base de datos dependiendo de los derechos que debo tener dentro de dicha Base de Datos.

Como un usuario determinado que accede al servidor a nivel de sistema operativo y posteriormente a una sesión del RDBMS las acciones que tome sobre la información dependerán de los derechos que como ese usuario determinado se me asignaron y las acciones que se pueden tomar van desde sólo ver la información de

las tablas ya creadas, hasta la construcción de una nueva base de datos y por lo tanto administración de la misma, para lo cual nos auxiliamos del RDBMS, el cual entre las acciones que debe tomar se encuentran la validación de usuarios, el mantenimiento de la consistencia de la información, por consistencia entendemos que no debe existir duplicidad de la misma y debe tener sus correspondientes relaciones dentro de la estructura misma de la base de datos, por otro lado debe controlar la concurrencia hacia la información (que más de un usuario intente acceder a algún recurso al mismo tiempo, esto no sucede ya que el que accede primero bloquea el recurso y hasta que lo libera el siguiente usuario podrá accederlo), etc. .

Un servidor de base de datos debe permitir el acceso remoto a él de tal manera que estando en cualquier parte de la red podamos acceder al recurso, en este caso la base de datos y trabajar con ella como si la tuviéramos de manera local, (concepto fundamental del acceso a través de cualquier herramienta de Internet).

IV.2.- Servidores de hipertexto

Un servidor de Web como se les conoce también es una computadora que ejecuta un software especial diseñado específicamente para dar respuesta a peticiones de documentos hechos por clientes del Web (esto es por los examinadores).

La computadora por sí misma puede ser cualquiera desde una PC, un mainframe, etc. Pero típicamente un servidor de Web es una máquina configurada especialmente para almacenar y hacer disponibles páginas Web en Internet.

Los documentos Web (archivos en HTML, así como gráficos y archivos multimedia) son almacenados en el disco duro del servidor y el *software* del mismo básicamente lo que permite es que otros usuarios transfieran estos archivos para desplegarlos a través de su propio *software* examinador.

Lo primero que se necesita para tener un servidor de Web es tener una computadora, una vez que ésta se tiene se debe adquirir el *software*, no es necesario que la máquina sea un servidor dedicado sólo a ser servidor de Web, sin embargo se debe considerar el hacerlo así por cuestiones de seguridad con respecto a su configuración de "Webserver" como tal.

El *software* para servidor de Web está disponible para la mayoría de las plataformas de sistemas operativos: UNIX, VAX, Macintosh, Windows NT, Windows y OS/2, existen dos asociaciones que han desarrollado software para el

servidor de Web y son CERN (European Particle Physics Laboratory) y NCSA (National Center for Supercomputing Applications) , la diferencia entre ellos es que para algunas de las plataformas se encuentran de manera libre en la red de manera que se pueden obtener gratuitamente, mientras que para otras plataformas no.

El *software* de ambas organizaciones para el caso de plataformas UNIX son llamados HTTPD (Daemon for HyperText Transmission Protocol) los cuales como su nombre lo indica son *daemons* (programas residentes para plataformas UNIX , para habilitar servicios), ambos pueden ser ejecutados como un servicio en máquina simple (*stand alone*) o un Servicio de Internet (que es otro *daemon*. llamado *inetd*) , si es configurado como *stand alone* el servidor que se inicia al momento de que el sistema arranca.

Si es configurado como *inetd*, el servidor no inicia hasta que no se se hace una petición. Esta configuración se recomienda sólo cuando el tráfico en la red es muy ligero ya que el hecho de que el servicio arranque cada vez que se genera una petición toma demasiado tiempo y consume también demasiado tiempo de procesador.

Ambos HTTPD soportan el manejo de CGIs y la creación de un directorio de acceso de HTTP para cada usuario y está marcado como `http://hostname/~username/` que hace referencia al subdirectorío HTML del usuario, así como cada uno de ellos tiene soporte para precompilar archivos al ejecutarlos.

En la actualidad existen múltiples versiones de estos *daemons* por parte de cada uno de los desarrolladores con las características que los requerimientos de ejecución de software implica el desplegar o mantener una página Web.

Suponiendo que tenemos una *workstation* UNIX y en ella el software para servidor Web, deben hacerse las consideraciones pertinentes para acceder al servidor como tal. a través de la red; debemos, entre otras cosas, preguntarnos ¿Cómo es que tenemos acceso a Internet? , si es a través de una línea telefónica se deben hacer las consideraciones para determinar si las 24 horas se estará utilizando la línea para que los usuarios puedan acceder al servidor Web, debe considerarse la velocidad de dicha línea telefónica para determinar si es la adecuada para la transferencia de información.

En caso de tener acceso a través de redes Ethernet, o bajo cualquier otra topología hay que hacer las consideraciones pertinentes para el acceso a Internet como lo son

los enrutadores, las conexiones apropiadas y por supuesto el *software* de comunicación.

La administración de un servidor Web requiere de la mayor atención por parte de la persona encargada, no basta el hecho de dejar ejecutándose el *daemon*, ni de tener instalado un examinador, hay que verificar qué tan eficientes son los accesos a nuestro servidor, qué tan rápidamente pueden desplegar los usuarios nuestras páginas Web. Incluir las páginas de las personas que lo requieran, mantener el espacio necesario en disco para el caché de disco que utilizan los examinadores.

El Web site o bien un servidor de Web o nodo dentro de Web es un espacio de interacción con los usuarios de Web ya sea a nivel Intranet (dentro de la red de la organización) o Internet, es un espacio dedicado al intercambio de información por lo cual hay que mantener vivos los mecanismos de comunicación entre los usuarios del Web site a través de sus cuentas de correos electrónicos o de mensajes dentro de las mismas páginas.

Otro aspecto dentro de la administración son las políticas de información que se debe o puede desplegar bajo una página Web, esto depende netamente de las políticas de la organización.

Dentro del diseño de un Web site el punto importante a tratar es el propósito de generarlo considerando lo siguiente:

- El tema o área a abarcar: Bajo qué área de conocimiento o en qué contexto de información convivirá tu Web site (Geología, Biología, información de presentación de empresa, o persona etc.)
- La audiencia: Definir a quién está orientada la información que se presentará en el Web (a estudiantes, empresarios, público en general, etc.) de ello depende el material a incluir.
- Definir el nivel de detalle a desplegar con respecto a la información, determinará la cantidad de material a utilizar y la forma en la que éste se presentará.
- Evaluar los beneficios que proporcionará: Tanto al usuario, como a nosotros el hecho de mantener un espacio como éste.

Además deben considerarse las políticas a nivel usuario para determinar quién tendrá derecho de ver o acceder a los archivos HTML a nivel de *workstation*

dentro del servidor Web y a su vez quién podrá colocar su información dentro de éste ya a nivel de servidor, es decir, a través de su *home page*.

Dentro de este mismo renglon cabría indicar también las convenciones para la creación de *home pages*, como pueden ser el tamaño máximo en bytes de la hoja de presentación de todos, los colores, logos, botones etc., que pueden llevar éstas.

Dentro del diseño de Web site existen metodologías que nos permiten explotar las características de nuestra información así por ejemplo tenemos Web sites de los tipos:

Top-Down

En este método se pretende iniciar con una página principal (front o top page) . también llamada "*home page*" y de ahí esparcirnos sobre páginas secundarias que contengan información específica y mínima.

Botton-Up

Con este método se generan las páginas secundarias de manera indistinta sin la preocupación de mantener una página principal, y la navegación entre ellas se da dependiendo de la relación de información que contengan unas con respecto a otras.

Incremental/In-time

Aprovecha los métodos anteriores, mantiene una página principal y a partir de ahí se generan las páginas secundarias, pero manteniendo relación entre ellas de manera que para saltar de una a otra no es necesario pasar por la página *top*.

Si se pretende que el servidor Web sea disponible para la comunidad Internet se debe desarrollar un plan de seguridad.

Lo primero es determinar cuáles son los datos que dentro de nuestra red deben permanecer seguros, garantizar que esta información no pertenezca a directorios públicos y como último verificar los directorios internos de cada usuario, evaluar la importancia que tiene la información de cada uno y el riesgo de permitir accesos a dichos directorios, ya que el otorgarlos inadecuadamente podría llevarnos a perder dicha información o a que alguien pueda dañarla.

IV.3.- Relación del hipertexto y bases de datos

La relación entre hipertexto y base de datos la podemos remontar al uso de bases de datos en la información disponible en Internet, desde el uso de las herramientas de búsqueda como Archie en el cual se generaba una base de datos con los nombres de los servidores FTP anónimos disponibles en la red o bien a través de Gopher que ya es, en sí, un catálogo que se actualiza rápidamente (manualmente) para la conexión a servidores FTP, pero ya con el despliegue de la información, o aún más con WAIS que maneja ya bases de datos indizadas para su búsqueda de información, es por ello que con la introducción del concepto de hipertexto para desplegar información y su utilización en herramientas visualizadoras o examinadoras como son Mosaic, Netscape, Lynx bajo Internet no puede dejarse de lado la relación con las bases de datos.

Las páginas Web o *home page* como las hemos llamado son páginas generadas bajo el concepto de hipertexto, cuya esencia son las ligas entre la información; sin embargo esta información es almacenada de manera general en archivos HTML, lo cual quiere decir que si la información llegara a actualizarse, se deben actualizar todos los archivos HTML que existen con esa información para mantener la consistencia y veracidad de dicha información.

Una consideración más es aquella en la cual decimos que mucha de la información que se puede o se despliega en Internet a través del Web es información con estructuras predeterminadas, es decir con formatos estándares y que varía de forma mínima constantemente, como lo pueden ser estadísticas, etc.

Visto de esta manera podemos considerar que la información almacenada en una base de datos y desplegada a través de una forma o diseño dentro de una página Web es una forma óptima de mantener actualizados datos y de mantener la veracidad dentro de las páginas sin necesidad de editar el archivo HTML cada vez que se actualice cierta información.

Es por ello que se han diseñado innumerables herramientas de acceso a base de datos a través de un examinador, para ello se requiere de *software* adicional y de la programación de éste para acceder a la correspondiente base de datos y para el correcto despliegue de la información contenida.

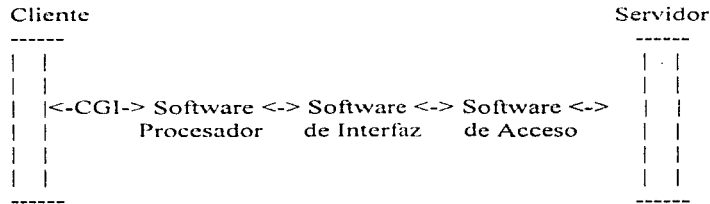
Entre el *software* que ya han pensado en ello se encuentra el lenguaje Perl que ha diseñado el "msql" (mini structured query language) lenguaje para acceso a bases de datos con sentencias SQL, lo podemos hacer a través de *scripts* normales haciendo llamados a la base de datos y ejecutando consultas SQL. Oracle tiene una

herramienta llamada Web Designer que permite un acceso casi natural a las bases de datos nativas de Oracle.

De hecho cualquier aplicación que logra conjuntar el despliegue de información de una base de datos a través del Web (cualquiera que sea el examinador que se utiliza) es en esencia un CGI.

Acceso a las bases de datos a través de un CGI

Detalles del Uso de un CGI y un Servidor HTTP que funciona a su vez como servidor de Base de datos



Notas:

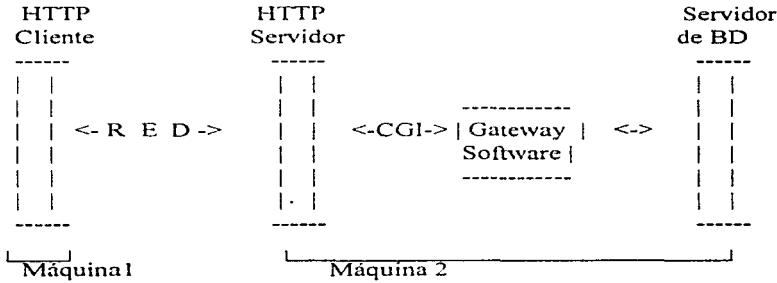
El *software* procesador (PS) es un programa llamado directamente por un documento HTML en el cliente del servidor HTTP. El PS procesa la entrada pasada a través de la entrada estándar o bien obtiene la información del ambiente de variables del cliente de HTTP. El método para pasar los datos está determinado por la llamada al CGI.

Una vez que los datos son leídos (puede ser una llamada a un enunciado SQL); el PS usa el *software* de interfaz para preparar el enunciado SQL para pedirlo a través del *software* de acceso. El PS usa el *software* de interfaz para procesar los resultados que son devueltos por la base de datos. El *software* de interfaz (IS) es una interfaz específica de base de datos necesaria para trasladar las consultas dentro de un formato que pueda ser reconocido por la base de datos que se está utilizando. El IS contiene las estructuras, variables y llamadas a funciones necesarias para comunicarse con la base de datos.

El *software* de acceso (AS) es usualmente un *software* que es distribuido con la base de datos. El AS permite el acceso a la base de datos a través de un programa API (Application Program Interface) que da el formato necesario a las consultas

que son mandadas por el AS, la base de datos regresa el resultado o los mensajes de error y el AS los pasa al PS a través del IS.

Utilizando una Base de datos que no tiene acceso directo a red

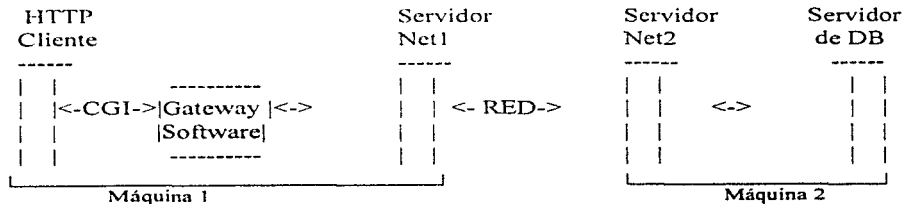


Notas:

El diagrama asume que el cliente y el servidor HTTP están en diferentes máquinas y el usuario está accediendo a la base de datos a través de la máquina cliente. El servidor HTTP y el servidor de base de datos deben ser la misma máquina.

La red en este caso es cualquier red que exista entre las dos máquinas. Esto puede ser una red Novell o bien una red TCP/IP entre máquinas Unix o bien una conexión Internet. El servidor HTTP debe ser configurado para permitir acceso de CGIs.

Utilizando un Software de red hecho por terceros y Accediendo a un Servidor de Base de Datos a través del World Wide Web



Notas:

Esta solución es únicamente para usuarios que accedan a un servidor de base de datos desde una máquina remota. El servidor de red debe aceptar consultas a la base de datos, transportar éstos a través de la red, mandar las peticiones al servidor de base de datos, regresar los resultados, regresarlos a través de la red y mandarlos a su vez al *software gateway* que los regrese al cliente HTTP.

El servidor de red en la máquina servidora de base de datos puede hacer llamadas a la base de datos a través de APIs provistos por la base de datos o a través de otro *software* de base de datos.

Esto puede ser innecesario ya que el *browser* del Web y el servidor de HTTP puede manejar el tráfico de la red por ellos mismos .

Capítulo V

V.- Desarrollo del sistema

V.1 Análisis del sistema

V.1.1 *Objetivo General*

Desarrollar un sistema de consulta de los planes de estudio de las carreras que ofrece la Facultad de Ingeniería y su integración a *la home page* de ésta a través de una base de datos creada en Oracle y su acceso por medio de hipertexto.

V.1.2 *Antecedentes*

El plan de estudios es la expresión formal y escrita de la organización de todos los requisitos que deben cubrir los alumnos para obtener un título, diploma o grado.

Los planes de estudio deben contener entre otros los siguientes apartados:

- a) Requisitos escolares previos o antecedentes necesarios para poder inscribir al estudiante al plan de estudios correspondiente.
- b) Estructura del plan de estudios.
- c) Valor en créditos de cada asignatura o módulo y del plan completo, en su caso.
- d) Tiempo de duración en periodos académicos del plan de estudios.
- e) Programas de cada asignatura o módulo.

Un plan de estudios es aprobado primeramente por el Consejo Técnico o Interno de la Facultad o Escuela correspondiente, posteriormente es sometido, a través del director de la institución académica, a la revisión del Consejo Universitario. Sin embargo de acuerdo a las últimas modificaciones con respecto a la aprobación de los planes y programas de estudios, esta última fase es cubierta ahora por los Consejos Académicos de Área respectivos.

Estos dictámenes son extractos de la legislación universitaria en la cual se manifiesta todo el proceso de los planes y programas de estudio, en donde no existe ningún apartado que indique el procedimiento de difusión de dichos planes.

Para obtener esta información, en el caso de la Facultad de Ingeniería, el alumno tiene que dirigirse al módulo de información de la facultad en los periodos de inscripción, periodo en el cual se le proporciona una hoja con el plan de estudios de su carrera, con la distribución por semestre de las asignaturas correspondientes, sus créditos y su seriación. Con respecto a los programas de asignatura el alumno tiene acceso a ellos si el profesor de la asignatura se lo presenta al inicio del semestre o

bien si lo pide en la coordinación o departamento de la materia correspondiente. Aunado a ello existe un libro en el cual se tienen los planes de estudio por cada carrera de la Facultad, este libro tiene varios tomos, en realidad de 1 a 2 tomos por cada carrera, ya que vienen los planes, las asignaturas, los temarios por asignatura, horas, créditos, seriación, bibliografía, e incluso sugerencias para los profesores con respecto a la forma de impartir la cátedra.

El libro antes mencionado se encuentra en la biblioteca de la Facultad a la disposición del alumno, sin embargo no existe la suficiente difusión de tal libro como para que los alumnos lo utilicen, siendo éste de gran utilidad ya que la información que proporciona puede ayudar en la preparación de los temas de las materias que los alumnos están cursando, o bien para aquellos que requieran presentar exámenes extraordinarios, ya que éstos se basan en los programas de las asignaturas.

Otro de los usos que se le puede dar a los planes de estudio es para información que requieran personas ajenas a nuestra Universidad, tal es el caso de los múltiples mensajes (correos electrónicos) recibidos en la Unidad de Servicios de Cómputo Académico por parte de varias Universidades del interior de la República solicitando información correspondiente a los contenidos de las materias impartidas por nuestra Facultad, o bien el caso de los alumnos que no concluyeron sus estudios y que les interesa reanudarlos solicitando temarios de algunas materias en particular y referencias para estudiar, incluso para los propios profesores ya que para cada asignatura se proporcionan sugerencia de técnicas de enseñanza para la impartición de los temas, así como técnicas de evaluación para los mismos.

V.1.3 Planteamiento del problema

Uno de los motivos que impulsan este proyecto es que la difusión de los planes de estudio sea de manera masiva. Otro es el hecho de poder tener en una lugar fijo y seguro toda esta información que por el momento es sólo una recopilación de hojas en los tomos correspondientes a cada carrera. Cabe mencionar que dichas hojas son copias de los programas de asignaturas y algunas de ellas no se notan claramente, además de que la presentación de la información es diferente dependiendo de la división a la cual pertenece la carrera que se está revisando, un ejemplo son las carreras de Ingeniería Civil e Ingeniería en Computación, por una parte para la carrera de ingeniero civil las asignaturas se muestran agrupadas de acuerdo a la división que imparte la materia, división de Ciencias Básicas, de Ingeniería Mecánica e Industrial, de Ingeniería Civil, etc. Mientras que para la carrera de ingeniero en computación las materias están agrupadas por el semestre en el cual el alumno debe inscribirse a ellas. Por lo cual se ve la necesidad de tener la información almacenada de manera estándar permitiendo que el acceso a ella sea

más directo y su modificación más sencilla, además de que se podría impulsar su difusión.

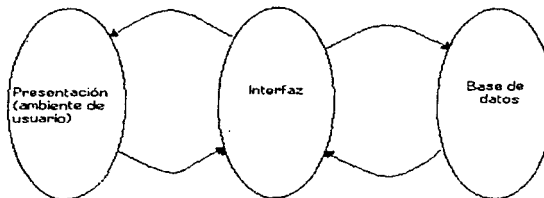
Con base en lo anterior se puede considerar el desarrollo del sistema que dé respuesta a estas necesidades. Si esta información se encontrara en una base de datos se tendría almacenada de manera estándar; su acceso sería más sencillo tanto para su difusión como para su actualización considerando que los cambios que se hacen sobre un plan de estudios es sobre la información ya existente.

Su difusión se propone sea a través de un ambiente de usuario que le permita accederlo desde cualquier parte, siempre y cuando cuente con conexión a Internet y con un navegador de Web.

V.1.4 Propuesta de solución

Podemos identificar tres elementos que integrarían el sistema a desarrollar:

- 1) Una Base de Datos
- 2) Ambiente de Usuario
- 3) Interfaz entre la base de datos y el ambiente de usuario



El esquema antes mencionado está basado en la arquitectura de las aplicaciones que acceden a las bases de datos en las cuales se distinguen 3 elementos básicamente:

El *backend*, el Programa de interfaz y el *frontend*.

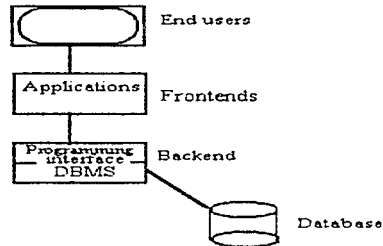
El *backend* es el Manejador de Base de Datos (DBMS), debe soportar todas las funciones básicas de un programa como tal, proporcionar la estructura adecuada para la correcta definición de los datos de la base de datos, mantener la seguridad y la integridad de los datos .

Los *frontends* son programas de aplicación que corren sobre los DBMS; existen dos tipos: aquellos escritos por los usuarios y los proporcionados por el propio DBMS.

En cuanto a los primeros son programas escritos en algún lenguaje de programación a través del cual se pueda manipular a su vez el lenguaje de acceso a la información que utiliza el DBMS. En la actualidad hablando de manejadores de bases de datos relacionales (RDBMS) este lenguaje es un estándar y se llama SQL (Structured Query Language) y como su nombre lo indica permite estructurar consultas a partir de la información contenida en una base de datos.

Con respecto a los programas proporcionados por el propio DBMS para la manipulación de la información éstos pueden ser varios y con diversas funciones, por mencionar algunos tenemos:

- Procesadores de lenguaje de consulta
- Generadores de reportes
- Hojas de cálculo
- Paquetes estadísticos
- Generadores de aplicaciones (incluyendo procesadores "4GL")

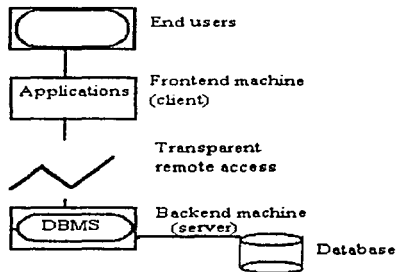


El esquema anterior nos muestra la arquitectura básica en la cual está sustentada la propuesta de solución: nuestros usuarios finales (End users) serán los alumnos, profesores y personas que requieran de la información contenida en la base de datos (Database), mientras que la aplicación que ellos usarán (*frontend*), será la página del Web que se construirá para tal efecto (*home page*); la forma de comunicarse ésta con la base de datos será el programa de interfaz (CGI Common Gateway Interface o Programming interface), y el *backend* es en sí el manejador de base de datos bajo el cual se creará ésta.

Aún más, considerando que el *frontend* y el *backend* no se encontrarán en la misma máquina ya que el *frontend* en este caso estará en la máquina que es el servidor de Web, y a él podrá accederse desde cualquier máquina que cuente con un navegador (*browser*) de Web, mientras que el *backend* estará en otra máquina junto a la base de datos; el esquema puede tomarse en un esquema cliente-servidor en un ambiente de procesamiento distribuido.

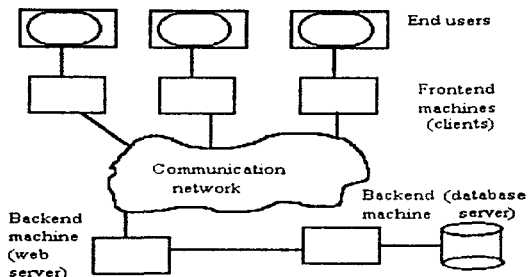
El término "procesamiento distribuido" significa que múltiples máquinas pueden estar conectadas a través de una red de tal forma que una simple tarea de procesamiento de datos es dividida entre algunas de las máquinas en la red. Cabe mencionar que ésta es sólo una de las múltiples variedades de procesamiento distribuido que se pueden implantar.

El esquema sería el siguiente:



Aquí observamos un acceso remoto mismo que se realizará a través del programa de interfaz .

O bien, una vez ya implantado el sistema, la forma en la que se verá en cualquier máquina que cuente con una conexión a Internet y con un *browser* de Web se puede representar así:

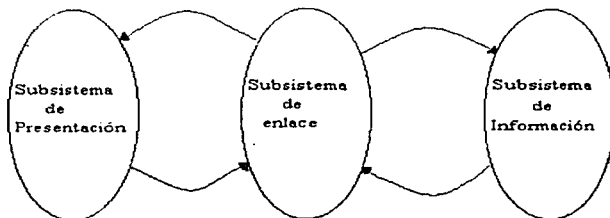


En este esquema la comunicación en la red es parte de la conexión a Internet que se debe tener para acceder al servidor de Web que contendrá la página de acceso a la base de datos .

Con base en lo ya expuesto se considera analizar de manera independiente cada una de las partes que constituirán el sistema, considerando a cada una un subsistema:

- Subsistema de información (Base de datos)
- Subsistema de enlace (Programa interfaz)
- Subsistema de presentación (Programa de usuario, *frontend*)

Por lo que el esquema mostrado al inicio de la propuesta de solución quedará:



V.2 Subsistema de información:

V.2.1 Análisis de requerimientos

Para la base de datos, se requiere de un manejador de base de datos que permita la administración de la misma, así como de un espacio en disco para su almacenamiento y, a través del manejador de base de datos, una cuenta de usuario que permita hacer las consultas necesarias a la base de datos. La información es lo más importante para la creación de la base de datos y para lo cual requerimos en particular de los siguientes datos:

- Planes de estudio actualizados
- Programas de asignatura (temarios)
- Antecedentes para cada asignatura
- División a la cual pertenece cada asignatura
- Créditos por asignatura
- Horas por asignatura
- Bibliografía recomendada para cada asignatura por tema

Para obtener dicha información y poder estructurar la base de datos se recurrió a los planes de estudio impresos, de los cuales en el anexo 1 de este capítulo se encuentra el programa de asignatura de una de las materias impartidas para todas las carreras de la Facultad. Para obtener un modelo de lo que será la base de datos se empleará el modelo Entidad-Relación que permite definir bases de datos relacionales. Considerando que el manejador de bases de datos a utilizar (Oracle) es un manejador de bases de datos relacionales, hablaremos un poco de las características de dichas bases de datos y sus ventajas con respecto a otros modelos.

De acuerdo al desarrollo que ha tenido la informática en general y considerando que el principal motor de su evolución ha sido sin lugar a duda el uso de la información, el acceso a ésta de manera más fácil, segura y sin que pierda su consistencia ha dado pie al desarrollo de técnicas o métodos de almacenamiento y de manipulación de ella llegando hasta lo que hoy conocemos como bases de datos.

Aún más, las bases de datos por sí mismas han evolucionado en su forma de almacenar o bien relacionar la información contenida en ellas; de acuerdo a ello tenemos bases de datos jerárquicas, en modelo de red, relacionales y distribuidas.

En el presente trabajo se profundizará en el modelo relacional ya que se creará una base de datos basada en este modelo, de cualquier forma en el apéndice del mismo se podrá consultar más acerca de los otros modelos.

V.2.2 Diseño

Empecemos por algunas definiciones:

Modelo: Abstracción de un hecho real o representación de una situación.

Modelo de datos relacional: Es una colección finita de tablas (de dos dimensiones; formadas de columnas y renglones) que representan una situación.

Características del modelo relacional:

- **Simplicidad :** Las tablas son una forma familiar y explicables por sí mismas para representar datos. La mayoría de la gente ha utilizado datos en forma de tabla, no se requiere de un entrenamiento especial para entender o utilizar los datos que se representan en las tablas. En pocas palabras podemos decir que son amigables al usuario.
- **Precisión:** Las tablas correctamente diseñadas mantienen un rigor matemático. dicen lo que significan y significan lo que dicen. Pueden ser implantadas y procesadas por una gran variedad de configuraciones de *software* y *hardware*. En pocas palabras son amigables con la computadora.
- **Flexibilidad:** Las tablas no solamente muestran la estructura de los datos, sino pueden mostrar los datos también. Esto nos permite manejar el modelo antes de implantarlo. En otras palabras, las tablas son apropiadas no sólo para modelar datos sino para procesarlos también.

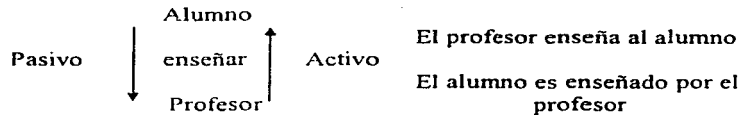
Como ya se había mencionado, se hará uso de una técnica de modelado llamada Entidad-Relación para obtener un prototipo de la base de datos que se implantará. Hablemos ahora del modelado, en él podemos encontrar algunos elementos que es necesario definir.

Situación: Conjunto de circunstancias bien definidas que pueden ser descritas utilizando un lenguaje natural suficientemente completo. Un lenguaje natural suficientemente completo incluye al menos los siguientes 3 constructores gramaticales:

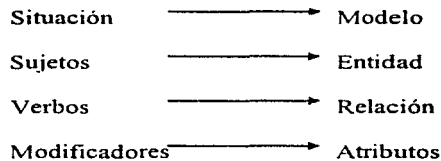
Sujetos: Un sujeto es el nombre de una persona, de un tipo de persona, de un animal, una planta, un lugar, una cosa, una substancia o una idea. Un sujeto propio es el nombre de una ocurrencia particular o instancia de un sujeto. Un pronombre es una palabra utilizada como sustituto de un sujeto que hace referencia a un sujeto que ha sido nombrado que se sobre entiende de acuerdo al contexto en el cual se utiliza.

Los sujetos pueden ser tangibles o intangibles y es imposible describir alguna situación sin al menos el uso de un sujeto.

Verbos: Un verbo es una palabra que describe un modo de ser una asociación, una acción o un evento. Los verbos describen el estado de los sujetos y relacionan a los sujetos dentro de las situaciones. Los verbos pueden ser activos o pasivos.



Modificadores: Un modificador es una palabra que califica a un sujeto o a un verbo de acuerdo a sus características, cantidad, extensión, etc. Los modificadores de los sujetos se llaman adjetivos y los modificadores de los verbos se llaman adverbios.



El modelo Entidad-Relación nos dice que podemos dividir el análisis en 3 fases.

- a) Modelado de Entidades
- b) Modelado de Relaciones
- c) Modelado de Atributos

Modelado de Entidades

El modelado de entidades se basa en los siguientes pasos:

- 1) Descubrir entidades: Concentrarse en sujetos, las entidades existen por sí mismas.

- 2) Definir el alcance de la entidad. Definir si esa entidad es de interés al sistema.
- 3) Definir una llave primaria. Una llave primaria es un atributo de la entidad, que permitirá identificar cada renglón dentro de la tabla que formará, de manera única; sus características son:

ND: No duplicada, garantiza la existencia única de ese elemento.

NN: No nula, garantiza que este dato siempre existe.

NC: No cambia, no permite cambiar este dato.

- 4) Documentar.
- 5) Incluirla en el diagrama entidad-relación.

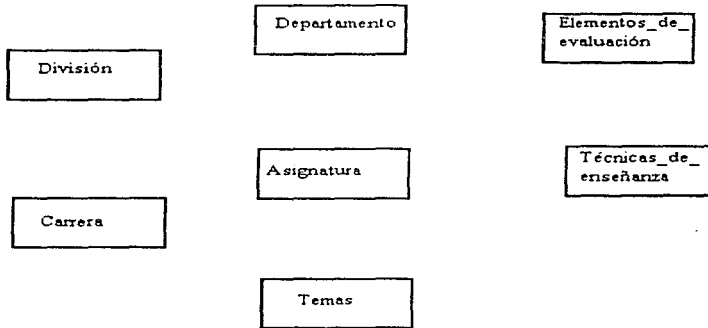
Las entidades identificadas pasarán a ser las tablas de la base de datos.

En el programa de asignatura se pueden distinguir los siguientes sujetos:

- 1.- Asignatura: Es el elemento principal de los programas de asignatura; cada una de ellas son cursadas por los alumnos de la Facultad y son impartidas por los profesores de la misma.
- 2.- División: La Facultad de Ingeniería para su organización realiza sus labores en divisiones de acuerdo a las distintas áreas de estudio de la Ingeniería. Estas divisiones se hacen cargo de la coordinación de las carreras que se imparten en la Facultad.
- 3.- Departamento: Las diversas divisiones de la Facultad están a su vez organizadas en departamentos; mismos que se encargan de la coordinación de la impartición de ciertas asignaturas, por lo cual las asignaturas son coordinadas por determinado departamento.
- 4.- Carrera: Clasificación de las diversas áreas de estudio de la ingeniería y de la cual se tiene un plan de estudio.
- 5.- Elementos de evaluación: Sugerencias planteadas para evaluar una asignatura.
- 6.- Técnicas de enseñanza: Sugerencias por parte de los departamentos coordinadores de las asignaturas hacia los profesores para la impartición de la cátedra.
- 7.- Temas: Clasificación del programas de estudio de cada asignatura a lo largo del período escolar.

Con base en lo anterior nos podemos dar cuenta de la importancia que cada uno de los elementos identificados tiene en la base de datos por lo cual podemos considerarlos indispensables para la obtención de información.

En el diagrama entidad-relación las entidades se representan como rectángulos.



Para cada una de las entidades se definirá un atributo que fungirá como llave primaria.

Asignatura: Clv_asigna
Departamento: Clv_depto
División: Clv_div
Carrera: Clv_carrera
Elementos_de_evaluación: Clv_evalua
Técnicas_de_enseñanza: Clv_enseña
Temas: Num_Tema

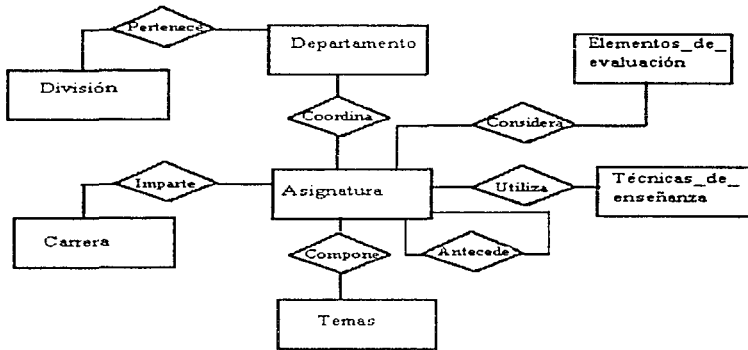
Modelado de relaciones

Una vez identificadas las entidades, las relaciones las obtenemos a partir de los siguientes pasos:

- 1) Descubrir relaciones, concentrarse en verbos.

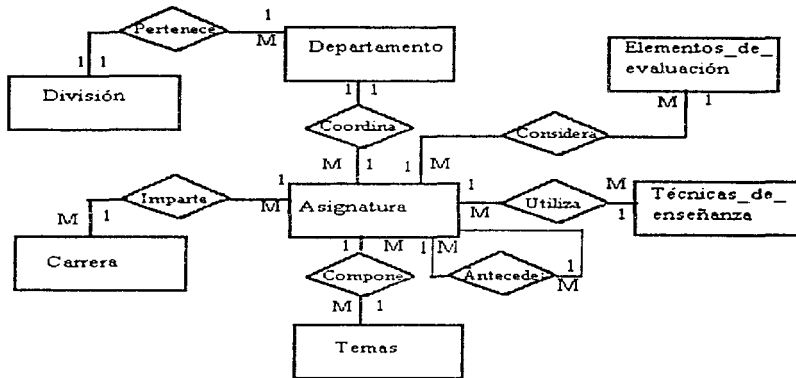
- 2) Definir el alcance de la relación, definir si la relación es importante para el sistema.
- 3) Definir el tipo de relación.
- 4) Documentar en el diagrama entidad-relación.
- 5) Documentar en tablas.

Para hacer referencia a las relaciones plantearemos el diagrama entidad-relación:



Las relaciones de acuerdo al número de elementos que se involucran de ambos lados de las entidades que participan (cardinalidad) se clasifican en 3.

- Relación 1 a 1
- Relación 1 a M
- Relación M a M



Para modelar las relaciones de acuerdo a su tipo se debe emplear la llave primaria de las entidades involucradas, esta llave al ser colocada en otra entidad pasa a llamarse llave foránea.

Relación 1 a 1:

La llave primaria de cualquiera de las 2 entidades involucradas debe migrar hacia la otra entidad, es indistinto cuál de las entidades migra su llave primaria, sin embargo de acuerdo a la utilización de los datos esto puede cambiar.

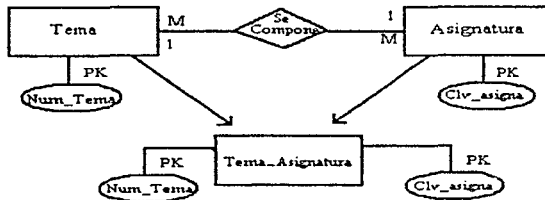
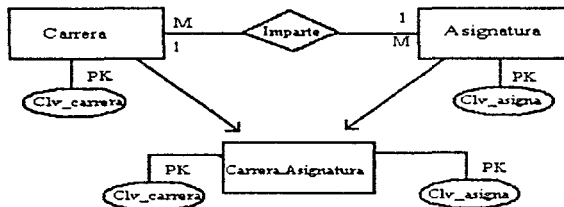
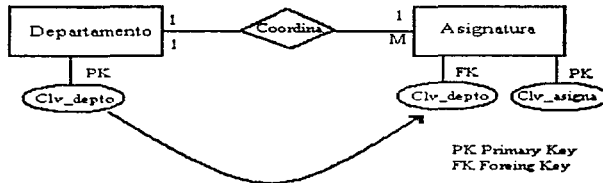
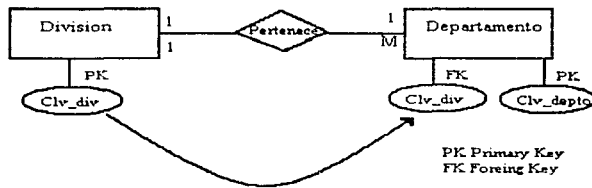
Relación 1 a M:

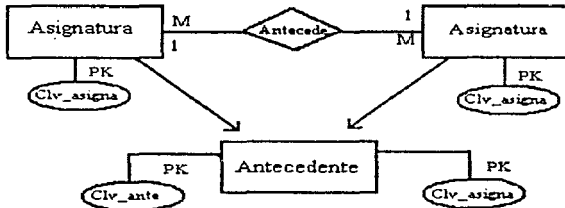
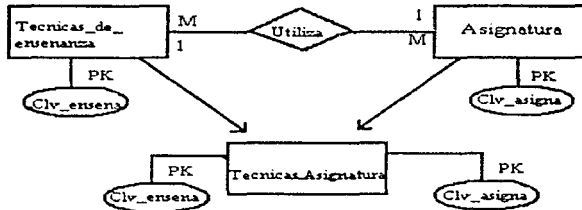
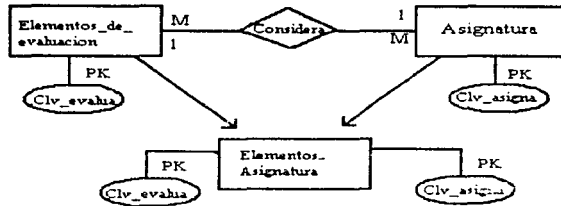
La llave primaria de la entidad donde sólo se involucra 1 elemento se migra hacia la entidad donde se involucran muchos elementos (M).

Relación M a M:

Se debe generar una nueva entidad cuya llave primaria es la combinación de las llaves primarias de las entidades que la originaron.

De acuerdo a lo anterior en nuestro modelo se dan los siguientes cambios:





Modelado de atributos

- 1) Descubrir atributos, concentrarse en adjetivos o adverbios.
- 2) Definir el alcance del atributo.

3) Documentar el atributo en la entidad; para colocar los atributos de manera correcta en una entidad se utiliza el método de normalización que permite verificar si nuestro modelo es correcto o bien si funcionará al ser implantado.

Los atributos para cada entidad se listan a continuación:

Asignatura

Civ_asigna (PK): Atributo considerado especialmente para distinguir de manera única cada una de las asignaturas.

Nom_asigna: Nombre de la asignatura.

Creditos_asigna: Número de créditos de la asignatura.

Objetivo_curso: Objetivo propuesto para la asignatura.

Civ_depto (FK): Clave del departamento que coordina la asignatura.

Hrs_total: Horas invertidas al semestre por la asignatura.

Hrs_teo: Horas de teoría invertidas a la semana para cada asignatura.

Hrs_prac: Horas de práctica invertidas a la semana para cada asignatura.

Semanas: Semanas invertidas al semestre por la asignatura.

Bibliografía: Libros recomendados para la mejor comprensión de los temas de la asignatura.

Plan: Año del plan de estudios en el cual fue aprobada la asignatura, o bin para el cual es válida.

Departamento

Civ_depto (PK): Atributo considerado para la identificación de manera única de los departamentos que componen a las divisiones.

Nom_depto: Nombre del departamento.

Civ_div (PK): Clave de la división a la que pertenece el departamento.

División

Civ_div (PK): Clave que distingue de manera única a las distintas divisiones.

Nom_div: Nombre de la división.

Abr_div: Abreviatura de la división, por lo regular las divisiones son identificadas por sus siglas.

Carrera

Clv_carrera (PK): Atributo que identificará de manera única las diversas carreras impartidas en la Facultad.

Nom_carrera: Nombre de la carrera.

Num_creditos: Número de créditos totales que deben ser cubiertos para concluir la carrera.

Tema

Num_tema (PK): Número de los diversos temas que componen a una asignatura.

Elementos_de_evaluacion

Clv_evalua (PK): Clave asociada a los diversos elementos de evaluación considerados por la asignatura.

Nom_evalua: Nombre del elemento de evaluación.

Tecnicas_de_ensenanza

Clv_ensena (PK): Clave asignada a las técnicas de enseñanza recomendadas para la impartición de la asignatura.

Nom_enseña: Nombre de la técnica de enseñanza.

Carrera_Asignatura

Clv_carrera (PK): Atributo que identificará de manera única las diversas carreras impartidas en la Facultad.

Clv_asigna (PK): Atributo considerado especialmente para distinguir de manera única cada una de las asignaturas.

Optativa: opción que identificará para cada carrera qué materias son optativas.

Tema_Asignatura

Clv_asigna (PK): Atributo considerado especialmente para distinguir de manera única cada una de las asignaturas.

Num_tema (PK): Número de los diversos temas que componen a una asignatura.

Nom_tema: Nombre del tema.

Objetivo_tema: Objetivo del tema.

Sugerencias: Sugerencias para la impartición del tema.

Contenido: Contenido del tema, puntos en los que se subdivide.

Hrs_tema: Horas que se proponen para dedicarse a dicho tema.

Antecedentes: Antecedentes para la mejor comprensión del tema.

Tecnicas_Asignatura

Clv_asigna (PK): Atributo considerado especialmente para distinguir de manera única cada una de las asignaturas.

Clv_ensena (PK): Clave asignada a las técnicas de enseñanza recomendadas para la impartición de la asignatura.

Elementos_Asignatura

Clv_asigna (PK): Atributo considerado especialmente para distinguir de manera única cada una de las asignaturas.

Clv_evalua (PK): Clave asociada a los diversos elementos de evaluación considerados por la asignatura.

Antecedente

Clv_ante (PK): Atributo considerado especialmente para distinguir de manera única cada una de las asignaturas, en particular a aquellas que son antecedente para alguna otra asignatura.

Clv_asigna (PK): Atributo considerado especialmente para distinguir de manera única cada una de las asignaturas.

Una vez encontrados los atributos para cada entidad se procede a normalizar.

Normalización:

Primera forma normal:

Dada una tabla T y con una llave primaria P y un atributo A, se dice que T está en primera forma normal si y sólo si, el valor del atributo A en cualquier renglón depende del valor de la llave primaria P en ese renglón.

En otras palabras, no se permiten valores constantes dentro de las tablas, bajo una misma columna.

Segunda forma normal:

Dada una tabla T en primera forma normal, con una llave primaria P en varias columnas, con componentes P1 y P2 y un atributo A, se dice que T está en segunda forma normal, si y sólo si, el valor del atributo A en cualquier renglón depende de los dos valores P1 y P2 en ese renglón. Las tablas con llaves primarias en una columna, siempre están en segunda forma normal.

En este caso se habla de la dependencia de cualquier atributo con respecto a las columnas que fungen como llave primaria compuesta, tampoco se permiten valores constantes.

Tercera forma normal:

Dada una tabla T en segunda forma normal con una llave primaria P y dos atributos A1 y A2, se dice que T está en tercera forma normal si y sólo si, el valor del atributo A1 en cualquier renglón no depende del atributo A2, a menos que A2 esté marcado como no duplicado (ND) y el valor del atributo A2 en cualquier renglón no depende A1, a menos que A1 esté marcado como ND.

Si existe dependencia entre atributos regularmente se debe crear una nueva tabla con los atributos que dependen entre sí y se establece una relación con la tabla original.

Haciendo la evaluación con respecto a los datos identificados para cada tabla y con base en las formas normales tenemos lo siguiente:

Para la tabla Asignatura:

El atributo "semanas" es una constante para las distintas asignaturas por lo que se eliminará y se contemplará como un elemento dentro de las consultas hechas al sistema.

Revisando cada uno de los atributos para las diversas tablas ya identificadas no se encontró ninguna otra incongruencia por lo que podemos pasar a la definición de cada una de ellas en Oracle.

Los datos anteriores, son parte del llamado "diccionario de datos", que es la documentación del sistema de base de datos, el diccionario de datos así como los modelos entidad-relación pueden ser generados por una herramienta CASE (Computer Added Software Engineer), que son programas especializados en la creación de modelos y la normalización de éstos, así como la generación de los diccionarios de datos que siempre resultan ser muy tediosos para los diseñadores.

V.2.3 Implantación

Dependiendo de la información que almacenará cada campo se le asocia un tipo de dato en la creación de la tabla.

Tabla	Campo	Tipo de dato
Asignatura	Clv_asigna	number(4)
	Nom_asigna	varchar(100)
	Créditos_asigna	number(2)
	Objetivo_curso	varchar2(500)
	Hrs_total	number(3)
	Hrs_teo	number(3,1)
	Hrs_prac	number(3,1)
	Clv_depto	number(4)
	Bibliografia	long
	Lab_prac	char(2)
Departamento	Plan	char(1)
	Clv_depto	number(4)
	Nom_depto	varchar2(120)
	Clv_div	number(2)
Division	Clv_div	number(2)
	Nom_div	varchar2(100)
	Abr_div	char(6)
Carrera	Clv_carrera	number(2)
	Nom_carrera	varchar2(40)
	Num_creditos	number(3)
Tema	Num_tema	number(2)
Elementos_de_evaluacion	Clv_evalua	number(1)
	Nom_evalua	varchar2(35)
Tecnicas_de_ensenanza	Clv_ensena	number(2)
	Nom_ensena	varchar2(50)

Tabla	Campo	Tipo de dato
Tema_Asignatura	Clv_asigna	number(4)
	Num_tema	number(2)
	Nom_tema	varchar2(150)
	Hrs_tema	number(3,1)
	Objetivo_tema	varchar2(300)
	Contenido	long
	Sugerencias	varchar2(2000)
Tecnicas_Asignatura	Antecedentes	varchar2(500)
	Clv_asigna	number(4)
Elementos_Asignatura	Clv_ensena	number(2)
	Clv_asigna	number(4)
Carrera_Asignatura	Clv_evalua	number(1)
	Clv_carrera	number(2)
	Clv_asigna	number(4)
Antecedente	Optativa	char(1)
	Clv_ante	number(4)
	Clv_asigna	number(4)

Al hacer el diseño y la implantación de una base de datos se deben realizar algunas tareas para su correcto funcionamiento:

- Se debe realizar el diseño de la base de datos sin considerar el *software* en el cual se vaya a implantar (esto es, el análisis de requerimientos y el diseño de acuerdo al modelo de base de datos que se utilizará).
- Se debe contemplar el modelo de los datos, es decir qué tipo de dato se almacenará en cada campo y en cada tabla.
- Se contemplará el espacio en el cual se almacenará la base de datos, es decir el tamaño que ocupará. La base de datos tendrá un tamaño adecuado a la información, los lineamientos para la determinación del tamaño están basados en lo siguiente:

Número de tablas * el número de campos que tiene * el tamaño de cada campo * el número de registros + un crecimiento adecuado a los datos + los índices que se consideren para el acceso más rápido a la información.

V.2.4 Mantenimiento

El subsistema de información está diseñado de acuerdo a las necesidades de almacenamiento y manejo de ésta, por lo cual el cambio en las futuras revisiones de planes de estudio bien pueden integrarse al cúmulo de información ya existente, la forma de introducir esta información es sencilla ya que se cuenta con una forma de captura diseñada en Access precisamente para hacer más cómodo el hecho de capturar tanta y tan variada información que se requiere para dar soporte al sistema de consulta, con dichas formas de captura se almacena en tablas de Access mismas que se exportan en forma de archivos de texto que posteriormente se introducen a las tablas en Oracle a través de un herramienta de este último llamado SQL Loader, cabe aclarar que el sistema fue pensado exclusivamente para consulta de información, por lo cual desde el subsistema de presentación no se puede alterar nada de la información almacenada en la base de datos.

Por otro lado la forma de captura está a disponibilidad del responsable del sistema a través de la instalación del sistema en Access del cual se presentan a continuación algunas de las formas de captura y las instrucciones para llevar a cabo la inserción de la misma en Oracle.

La base de datos en Access contiene sólo algunas de las tablas, aquellas cuya información es extensa y por lo que las formas de captura de Oracle de la versión con la cual se cuenta se quedan cortos para dicha captura.



La anterior es una vista general de la base de datos en Access y a continuación se presentan cada una de las formas de captura con las cuales tendrá que trabajar el administrador de la misma para la actualización de los datos y para la captura de todos los nuevos datos.

The screenshot shows a Microsoft Access window titled 'Microsoft Access - [Asignatura]'. The menu bar includes 'Archivo', 'Edición', 'Ver', 'Registros', 'Ventana', and 'Ayuda'. The toolbar contains various icons for file operations and data management. The main form area is titled 'ABASTÉCIMIENTO DE AGUA POTABLE' and contains several data entry fields:

- CLV_ASIGNA: []
- NOM_ASIGNA: []
- CREDITOS_ASIGNA: [3]
- HRS_TOTAL: [0]
- HRS_PRAC: [0]
- HRS_TEO: [0]
- CLV_DEPTO: [0]
- LAB_PRAC: []
- PLAN: [92]
- OBJETIVO_CURSO: []
- BIBLIOGRAFIA: []

At the bottom of the window, the status bar shows 'Registro 1 de 622', 'Presentación Formulario', and 'MAY NUM'.

Como se puede observar es la forma de captura para la tabla de Asignatura, misma que contiene un campo Varchar2(500) y un campo Long correspondientes a las columnas de objetivo_curso y bibliografía respectivamente, desde aquí se podrá insertar dicha información y almacenarla dentro de un archivo de texto con limitadores de campos para posteriormente insertarlos a las tablas de Oracle.

A continuación se muestra la forma de captura de la tabla derivada de la relación entre las tablas carrera, plan y asignatura, esto debido a que algunas asignaturas son optativas para ciertas carreras mientras que para otras son obligatorias, así como para algunas dependiendo del plan de estudios correspondiente tienen más o menos créditos, además dependiendo de la carrera algunas incluyen laboratorio y otras no, y también otras tienen alguna materia precedente.

CARRERA ASIGNATURA

CLV_CARRERA: 1

PLAN: 32

CLV_ASIGNA: 32

CREDITOS: 6

LAB_PRAO:

OPTATIVA:

Registro 1 de 159

Presentación Formulario

Forma de captura para la tabla Carr_Asigna_Plan en Oracle.

La siguiente forma de captura corresponde a la tabla que incluye a la relación de las tablas asignatura y tema, en donde se incluye el nombre del tema, así como su contenido, los temas de antecedente, su duración, etc.

TEMAS

CLV_ASIGNA: 1100

NUM_TEMA: 1

NOM_TEMA: NUMEROS REALES

HRS_TEMA: 75

OBJETIVO_TEMA: APLICAR LAS PROPIEDADES DE LOS CONJUNTOS CONSTITUYENTES DEL SISTEMA NUMERICO REAL PARA DEMOSTRAR

CONTENIDO: 1) EL CONJUNTO DE LOS NUMEROS NATURALES. CONCEPTO DE NUMERO NATURAL. POSTULADOS DE PEANO. METODO

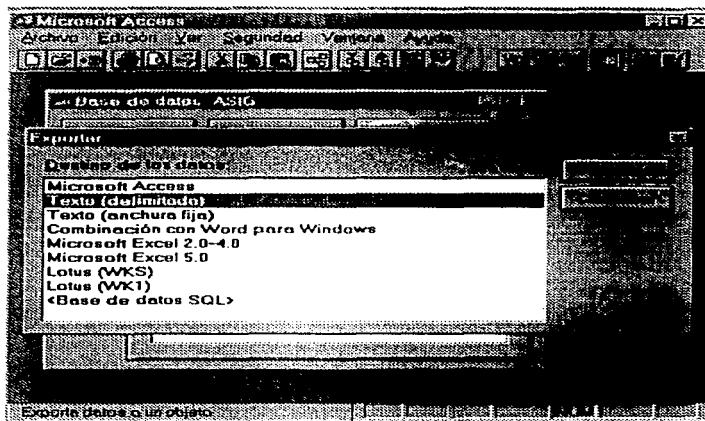
SUGERENCIAS:

ANTECEDENTES:

Registro 1 de 1

Presentación Formulario

Una vez que los datos han sido capturados en Access para las tablas ya mencionadas se deben exportar en formato de texto delimitado como se muestra en la imagen:



Una vez seleccionada dicha acción se procede a elegir la tabla y los separadores de campo así como a dar el nombre del archivo en el cual se almacenará, este debe tener extensión 'txt'.

Una vez que se tiene el archivo de texto se debe trasladar al servidor de Oracle, en nuestro caso haciendo un ftp hacia la máquina llamada Tork, una vez ahí se procederá a utilizar la herramienta SQL Loader de la siguiente forma:

Se requiere tener un archivo de control cuya extensión debe ser 'ctl' y cuyo contenido debe ser como el del siguiente ejemplo para insertar, dentro de la tabla asignatura:
archivo asig.ctl:

```
Load data
append
into table ASIGNATURA
fields terminated by ',' optionally enclosed by '"'
(clv_asigna integer external,nom_asigna char(120),creditos_asigna float external
(2),plan char(8),hrs_total float external,hrs_teo float external,hrs_prac float
external,bibliografia char(1600),objetivo_curso char(3000))
```

Como se puede observar se le indica el nombre de la tabla en la cual se agregarán los registros así como el carácter de terminación de campos y el orden en el que están en el archivo de texto y el tipo de dato que respetan, como se puede observar se manejan sólo dos tipos de datos: char para campos tipo carácter de cualquier longitud y numérico ya sea entero (integer), o flotante (float).

Debe existir por otro lado un archivo ejecutable, en este caso un script de Unix que debe contener lo siguiente:

Se recomienda que sea un archivo con extensión '.com' con las siguientes instrucciones:

archivo asig.com

```
#!/bin/ksh  
sqlldr claudia/claus control=asig.ctl data=asig.dat errors=1
```

donde sqlldr es el comando que ejecuta a la herramienta, debe ir seguido de una cuenta y un password válidos dentro de Oracle.

En la parte de control va el nombre del archivo de control, en data el nombre del archivo de datos, en nuestro caso el archivo .txt, y en errors el valor de 1, para que genere un archivo llamado *.bad donde incluye todos los registros que causan problemas o errores al insertarse.

Para el caso de las otras tablas, debido a que no son demasiado extensas ni su información compleja pueden actualizarse o insertarse datos en ellas directamente utilizando comandos de SQL (Insert, Update).

V.3 Subsistema de presentación

V.3.1 Análisis de requerimientos

El subsistema de presentación es aquél que estará en contacto con el usuario final, llamado también *frontend*, éste debe contemplar algunos aspectos entre los cuales podemos considerar que:

- a) Debe ser fácil de utilizar por el usuario, “debe de ser amigable”.
- b) Debe ser funcional, las opciones o funciones presentadas serán las mínimas necesarias para la obtención de los datos requeridos por el usuario.
- c) Debe ser eficiente, deberá responder a las peticiones rápidamente y con la mayor precisión acerca de los datos.

De acuerdo a la propuesta de solución dada, la forma de hacer el *frontend* para el sistema será a través de una página Web o *home page* que reflejará la consultas hechas a la base de datos a través del subsistema de enlace.

Considerando lo anterior presentaremos algunos de los puntos que permiten determinar si una página es eficiente o no.

- 1) La página debe desplegarse en el visualizador del Web lo más rápido posible, dando la impresión de una respuesta inmediata.
- 2) Los gráficos, si es que contienen, deberán desplegarse a la par que el texto, de no ser así, por lo menos permitirán el despliegue del texto.
- 3) La transmisión de los datos se hace a través de la red de acuerdo a las convenciones y protocolos utilizados por el Web.

La aplicación manejará gráficos que desplegarán los planes de estudio de cada carrera tal y como se entregan a los alumnos en los periodos de reinscripción.

Las consultas se mostrarán en texto de acuerdo a los resultados que arroje la base de datos a través del subsistema de enlace que es el programa que comunicará a los subsistemas de información y de presentación.

V.3.2 Diseño

Para el diseño de la *home page* se utilizará HTML 3.0 que es “HyperText Markup Language” que es un lenguaje que permite crear textos con características especiales para una página del Web.

Dentro de las cosas que podemos hacer con el HTML están:

- Dar formato texto
- Crear Listas
- Establecer vínculos hacia otras páginas
- Insertar imágenes
- Crear tablas

Para hacer lo anterior HTML utiliza elementos de código llamados etiquetas. Las etiquetas se insertan en el archivo de texto que contiene la información que se desea mostrar a través del visualizador; éstas siempre se encuentran delimitadas por los caracteres `<>`, generalmente en mayúsculas y se colocan al inicio y al final del texto al cual modifican, la etiqueta al final de dicho texto lleva una diagonal. A continuación mostramos un ejemplo:

Queremos que la palabra “HTML” aparezca en negritas en el visualizador, para ello escribiríamos el código de la siguiente forma:

Nosotros utilizamos `HTML` para crear una página que pueda ser vista en el Web.

Lo anterior se verá así:

*Nosotros utilizamos **HTML** para crear una página que pueda ser vista en el Web.*

Los documentos escritos en HTML tienen una estructura, que aunque no es estrictamente necesaria, para efectos de comprensión es importante seguir.

Debido a que la estructura de una página se divide en dos partes principales:

- La cabeza (*HEAD*)
- El cuerpo (*BODY*)

Éstas deben ser especificadas en el documento HTML.

La cabeza sirve para poner comentarios que permitan de alguna forma documentar el contenido de la página, como quién fue el realizador, cuándo se realizó, etc. En la cabeza se incluye también el título de la página. La cabeza es la primera parte de la página que ejecuta el visualizador, por lo cual podemos colocar código que queremos que se ejecute antes de que se muestre la página, por ejemplo código de programas en JavaScript o Java.

El título del documento es desplegado en alguna sección especial de la ventana del visualizador (Mosaic pone el título en una caja de texto, la cual se encuentra abajo de la barra de menú mientras que Netscape lo coloca en la barra de título en la parte superior de la ventana). El título es usado por la mayoría de los visualizadores para emprender búsquedas de documentos en particular o bien para marcar una referencia hacia un documento en particular (*bookmark*) se hace a través del título.

Por lo anterior se recomienda colocar un título que describa el contenido de la página, que no sea demasiado largo ya que los visualizadores tienen un máximo de entre 50 a 60 caracteres.

El cuerpo es la parte del documento en la cual se coloca el texto y las imágenes que mostrará la página, aquí se define la configuración de la página en cuanto al color del texto, del fondo, de los vínculos, los vínculos ya visitados, además de permitimos especificar una textura de fondo (*background*).

Para comenzar una página Web necesitamos cualquier editor que nos permita guardar la información en forma de texto simple, y en código, el cual aun no tiene el formato deseado. Los archivos deben iniciar con la etiqueta <HTML>. Lo que hace esta etiqueta es indicarle al visualizador que el archivo que desplegará contiene etiquetas HTML. Asimismo, la última línea del documento será la etiqueta final </HTML>.

La vista general de un archivo HTML según lo antes mencionado será:

```
<HTML>
<HEAD>
<TITLE>Título del documento</TITLE>
</HEAD>
<BODY>
<etiqueta> Texto o imagen deseada</etiqueta>
</BODY>
</HTML>
```

Los visualizadores pueden desplegar imágenes junto con el texto, lo que hace que los documentos se vean mejor y lo que es más importante, las imágenes comunican información que sería imposible o muy difícil de transmitir con texto.

Los gráficos se insertan con la etiqueta y el nombre del gráfico. No todos los visualizadores pueden leer todos los formatos de archivos gráficos. El formato GIF (Graphic Interchange File) es el más utilizado puesto que la mayor parte de los visualizadores gráficos pueden mostrarlos.

Otro formato que puede ser utilizado es el JPEG (Joint Photographic Experts Group) la ventaja de estos archivos es que ocupan menor espacio que cualquier otro formato de imágenes por lo cual el visualizador los despliega con mayor rapidez.

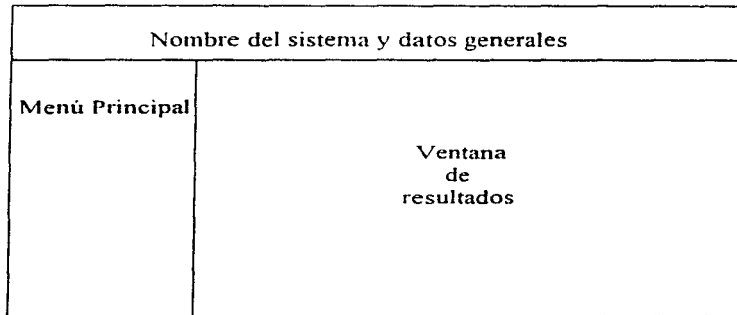
Por otro lado las imágenes al igual que los textos pueden establecer vínculos de tal manera que al dar un click sobre ellas pueden trasladarnos a otro documento. O bien pueden establecer un "mapa sensitivo", es decir, una imagen puede dividirse en regiones de tal forma que mantenga vínculos con más de un documento al establecer que cada región de la imagen haga referencia a un documento diferente.

Los mapas sensitivos se establecen desde HTML de acuerdo a coordenadas que son las que definen regiones de acuerdo a las cuales se establecen los vínculos con los diferentes documentos.

Además de todas las ventajas que HTML por sí mismo proporciona ha tenido actualizaciones de acuerdo a las capacidades que ha ido adquiriendo con el uso de otras herramientas para la creación de páginas cada vez más dinámicas tal es la inclusión de código en Java o JavaScript. Java es un lenguaje orientado a objetos, cuyos objetos son principalmente las partes de las páginas que un visualizador puede mostrar, diganse documentos, ventanas, formas, etc. y los llamados *applets* que son aplicaciones con animación (movimiento) para la presentación de la información.

JavaScript por su parte es un lenguaje orientado a eventos y trabaja con guiones (scripts), esto significa que el intérprete de instrucciones en JavaScript, en este caso el visualizador, las ejecuta una a una, al momento de leer el archivo HTML que contiene código, por el lado de los eventos JavaScript manipula los objetos que son o fueron creados por Java para programar acciones que ocurran sobre ellos. Tal será el caso de un vínculo que al pasar el mouse sobre él cambiara el color del fondo de la página (*background*).

Para el diseño se ha pensado estructurar la página de la siguiente forma :



Para poder mostrar una página con estas características emplearemos la definición del objeto *Frame* o cuadro dentro del código HTML, de la siguiente forma:

```
<html>
<head><title>Frame Template</title></head>
<frameset rows = "70,*">
<frame src="header.html" marginwidth="0" marginheight="0"
scrolling="no" noresize name="header">
<frameset cols="190,*">
<frame src="acategory.html" marginwidth="0" marginheight="0"
scrolling="no" noresize name="acategory">
<frame src="content.html" scrolling="auto" name="content">
</frameset>
</html>
```

Donde **frameset** es la etiqueta de definición de una forma distinta a la página normal.

Rows nos indica cuántos renglones (ventanas horizontales) se definirán, para hacerlo se indica el tamaño en pixeles o bien en porcentaje con respecto al tamaño de la ventana del visualizador, indicando tantos tamaños o bien tantos porcentajes como renglones se deseen.

Cols indica el número de columnas (ventanas verticales) que se desean y se especifican de la misma forma que los renglones.

Cada *frame* definido contiene o mostrará un archivo HTML diferente y éste puede cambiar de acuerdo a los eventos que se susciten para hacer referencia a cada uno de ellos. La opción de **frame src** indica cuál es el archivo HTML que contendrá cada *frame* y en el caso de la programación de cambios debido a algún evento que se suscite en la ejecución de la página hacemos uso de la convención de objetos definidos .

frame name se refiere al nombre (identificador) que utilizaremos para referirnos a cada *frame* y para cambiar su contenido de ser necesario.

Frame marginwidth se refiere al tamaño del margen que se quiere con respecto a otro *frame*. **Frame marginheight** se refiere al grosor que debe tener ese margen. **frame scrolling** es para permitir que parezcan las barras de scroll en caso de que la información no se pueda presentar toda en el tamaño del *frame* y por último el **frame noresize** es para impedir que el usuario pueda cambiar el tamaño del *frame*.

<pre><html> <head><title>header</title></head> <body bgcolor="#ceffbd"> </body></html></pre>		HEADER
<pre><html> <head> <title>a:category</title> </head> <base target="content"> <body bgcolor="#ceffbd"></pre>	<pre><html> <head><title>content</title></head> <base target="_top"> <body bgcolor="#ffffff"></pre>	CONTENT
<pre></body> </html></pre>	<pre></body> </html></pre>	
A CATEGORY		

Y para hacer referencia a cada uno de ellos se hace de la siguiente forma:

Es necesario tener el control de un *frame* que cambia información en otro *frame*. Si la etiqueta `<frame>` incluye `name="content"`, el *frame* que va a controlar el cambio de contenidos en éste *frame* debe contener la etiqueta `<base target = "content" >` y

por otro lado el frame cuyo código incluye `name="content"` debe incluir la etiqueta `<target = "_top">`, lo cual permitirá que el frame content se muestre por sí mismo cada vez que aparece por primera vez.

Ejemplo:

```
<frameset cols="20%,80%">
<frame src="index.html">
<frame name="content"
  src="content.html">
</frameset>
```

Código HTML que define los frames

Frame, index.html:

```
<base target="content">
<body>
</body>
```

**Contenido del archivo html index.html
(frame que no cambia y controla a "content")**

Frame, content.html:

```
<target="_top">
<body>
</body>
```

**Contenido del archivo html para el frame
"content"**

En esta forma podemos identificar que la programación que se realiza bajo HTML es una programación basada en objetos a los cuales podemos manipular a través de sus propiedades.

Por otro lado, al programar los eventos dependeremos del objeto a utilizar así como de lo que queremos que suceda. Por ejemplo, podemos tener alguna imagen sensitiva, es decir, que al dar un clic del ratón sobre ella ejecute una función, por ejemplo que despliegue el resultado de una consulta a la base de datos sobre una materia de un plan de estudios en particular.

O bien se puede tener un botón de función que al dar clic sobre él muestre una lista o bien una forma para realizar una consulta, para este caso el evento sería al "hacer clic", mientras que la función será alguna acción que hayamos elegido, por ejemplo

que cambie el contenido de algún *frame* para mostrar un lista como ya habíamos mencionado.

En el caso de nuestro diseño tenemos que en el *frame* Header estará el título de nuestro sistema y una liga hacia los datos generales del sistema. En el *frame* Content se mostrarán los resultados de las consultas como serán las imágenes sensitivas de todos los planes de estudio, la información sobre una materia en particular o bien la información acerca de cada una de las carreras impartidas en la Facultad de Ingeniería.

De acuerdo con lo anterior el *frame* Index contendrá un menú de acciones que siempre será visible al usuario para permitirle "navegar" entre las opciones del sistema.

V.3.3 Implantación

Como ya se mencionó en la fase de diseño el subsistema de presentación estará formado por una *home page* que en su presentación principal mostrará *frames* cada uno de los cuales presentará información referente al sistema de información de la siguiente manera:

Título e información general	
Menú Principal	Ventana de Resultados

El código para generar esta presentación se muestra a continuación y es el contenido del archivo "proy.html".


```
<html>
<head><title>Sistema de Consulta de Planes y Programas de
Asignatura</title></head>
<frameset rows="70,*" border=3 bordercolor="#ffff00"
framespacing=0>
<frame src="cabeza.html" marginwidth="0" marginheight="0"
scrolling="no" noresize name="header">
<frameset cols="190,*">
<frame src="menu.html" marginwidth="0" marginheight="0"
scrolling="no" noresize name="menu">
<frame src="escudo.html" scrolling="auto" name="content">
</frameset>
</frameset>
</html>
```

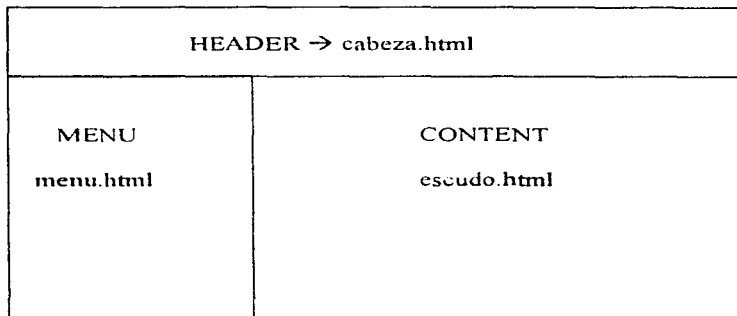
Las palabras en negritas son palabras reservadas de html, que definen 3 *frames* : header, menú y content, cada uno de los cuales está representando a su vez otro documento "html" que define su estructura como color de fondo, tipo de letra y contenido.

Los frames tienen las siguientes características:

Header es el frame de encabezado, tiene una altura de 70 píxeles y de todo el ancho de la pantalla de visión del navegador, no tiene definidos márgenes con respecto a la ventana del navegador y no permite el scroll, es decir que si la información que se despliega en él es más grande, no se pueden hacer desplazamientos horizontales, ni verticales. Y su contenido se encuentra en el archivo "cabeza.html".

Menu es una columna de un *frame* mayor definido por la instrucción: **frameset rows="70,*"** donde el 70 corresponde a la altura del *frame* Header y el * quiere decir que el resto de la pantalla mostrada por el navegador corresponderá a la altura de otro *frame*, mismo que es dividido a su vez en 2 columnas con la instrucción: **frameset cols="190,*"**, es decir que el tamaño del *frame* menú es de 190 píxeles de ancho por la altura que resta a la pantalla después de definir el *frame* header. Y su contenido está definido en el archivo menu.html.

Content es la columna restante del frame mencionado anteriormente, tiene dimensiones determinadas por los tamaños restantes de quitarle a la altura 70 píxeles para el *frame* header y 190 píxeles de ancho del *frame* menu y su contenido se encuentra en el archivo escudo.html.



Los contenidos de cada uno de los *frames* se muestran a continuación:

cabeza.html

```
<html>
<base target="content">
<body background="imagen/back.jpg" text=c0b000 link=800000 vlink=800000>
<center><font face="Times New Roman" size=5 color="#808000">
Sistema de Consulta de Informaci&oacute;n de los Planes de Estudio
de la FI</font></center><font face=""Times New Roman"
size=2 color="#C0B000" ><a href="infor.html">
<div align="right">Informaci&oacute;n general</div></a>
</font>
</body>
</html>
```

El código html define que el *frame* en donde se desplegarán los resultados de invocar a ligas (diganse otros archivos html) es "content", es decir, un frame diferente al que estamos definiendo, tendrá un color de fondo definido por el archivo "back.jpg", el color del texto esta definido por el número hexadecimal "c0b000", el color para el texto que es una liga tendrá el color definido por el número "800000", el tipo de letra a utilizar es "Times New Roman" y tiene una liga hacia el archivo "infor.html".

menu.html

```

<html>
<head>
<title> Sistema de Consulta de Planes y Programas de Asignatura </title>
<base target="content">
</head>
<body background="imagen/back2.jpg" text=800000 link=0000FF
vlink=0000FF
onload="scroll( );return true;">
<script lenguaje="JavaScript">

```

```

<!-- beginning of JavaScript applet-----

```

```

var m1=" Hola bienvenido al Sistema de Consulta de Planes de Estudio de la
Facultad de Ingenieria. ";

```

```

var m2="Para consultar da un clic sobre el boton de tu eleccion."

```

```

var scrtxt=m1+m2;

```

```

var lentxt=scrtxt.length;

```

```

var width=100;

```

```

var pos=1-width;

```

```

function scroll( ){

```

```

    pos++;

```

```

    var scroller=" ";

```

```

    if (pos==lentxt){

```

```

        pos=1-width;
    }

```

```

    if (pos< 0) {

```

```

        for (var i=1;i<=Math.abs(pos);i++){

```

```

            scroller=scroller+" ";

```

```

            scroller=scroller+scrtxt.substring(0,width-i+1);
        }

```

```

    }

```

```

    else {

```

```

        scroller=scroller+scrtxt.substring(pos,width+pos);
    }

```

```

}

```

```

    window.status=scroller;

```

```

    setTimeout("scroll()",50);

```

```

}

```

```

//----->

```

```

</script>
<form>
<center>
<br><font face="Times New Roman" size=5>Menú Principal </font>
<br> <br><br>
<map name="mapadeus">
<area shape="rect" coords="5,16,124,50" href="plan.html">
<area shape="rect" coords="5,73,124,108" href="carreras.html">
<area shape="rect" coords="5,131,124,166" href="asigna.html">
<area shape="rect" coords="5,189,124,222">
</map>

</body>
</html>

```

En este archivo se define el *frame* que permite acceder a la información de la base de datos a través de un menú que está representado por una imagen sensitiva misma que representa "botones de comando" con los nombres de los elementos a consultar: Planes de estudio, Carreras y Asignaturas, así como una opción de información. Al dar un clic con el ratón sobre cualquiera de ellos en el *frame* "content" aparecerán los resultados o bien las opciones del botón seleccionado. Además de que contiene un "applet" de java, que permite desplegar un mensaje en la barra de estado del visualizador, tal y como si fuera un letrero electrónico.

escudo.html

```

<body background="imagen/back2.jpg" text="800000" link="c9b000"
vlink="c9b000">
<table border=0 cellpadding=10 cellspacing=10 width="20%">
<tr align=center valign=center>
<td>
<table border=5 cellpadding=10 cellspacing=10 width="50%">
<tr align=center valign=center>
<td>
</table>
</center>
<td>
<table border=5 cellpadding=10 cellspacing=10 width="300%">
<td><center>
<h2>operaci&ocute;n del sistema de <br> consulta de informaci&ocute;n de
los planes de estudio</h2>
</center>

```

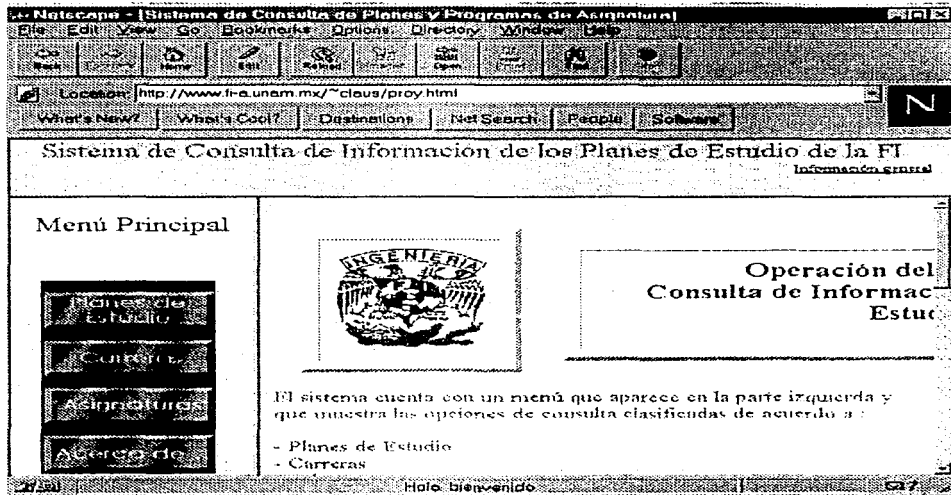
```

</table>
</table>
<font face="times new roman" size=4 color="808000">
El sistema cuenta con un men&uacute; que aparece en la parte izquierda
.....
<br><br><a href="http://www.fi-a.unam.mx/"target="top"> <img src =
"imagen/flecha.gif" border=0> ir a la p&aacute;gina principal de la fi</a>
<br>
 </a>
<a href="mailto:claus@cancun.fi-a.unam.mx" <br><br>
</font></body>

```

El contenido del archivo escudo.html, es un documento con las instrucciones de operación del sistema de consulta, además de contener una liga hacia la página principal de la Facultad de Ingeniería.

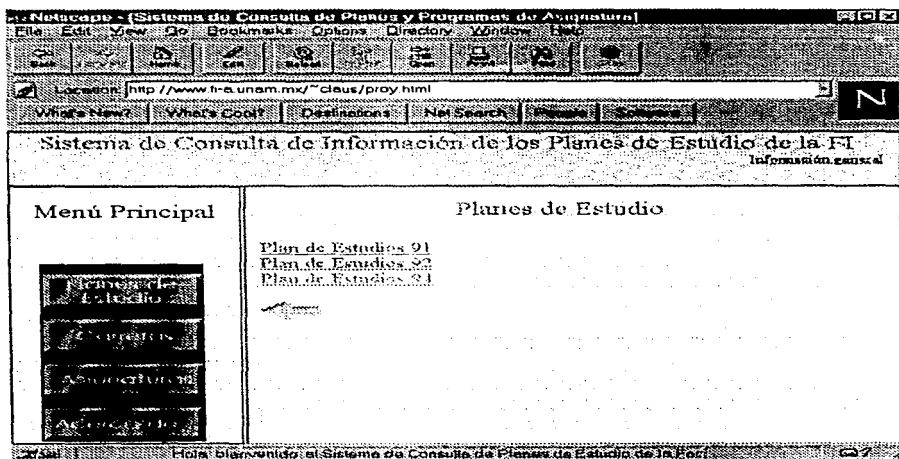
La vista general de la página con estos frames definidos es la siguiente:



Para cada opción del menú principal hay acciones a realizar de acuerdo a ello se ejecuta el código contenido en cada archivo "html" que se indica en el botón elegido:

Planes de Estudio:

En planes de estudio se ejecuta un archivo llamado plan.html que contiene a su vez ligas hacia otros archivos html que contienen la lista de todas las carreras de acuerdo a los planes de estudios correspondientes:



Cada liga va a los archivos planes91.html, planes92.html y planes.html mismos que contienen las ligas hacia los documentos html que definen los mapas sensitivos para consultar la información de las materias de cada carrera de acuerdo al plan de estudios seleccionado.

Menú Principal

- Planes de Estudio
- Carreras
- Asignatura
- Asesorías

Plan de estudios de la Carrera de Ingeniero Topógrafo

Examen Diagnóstico

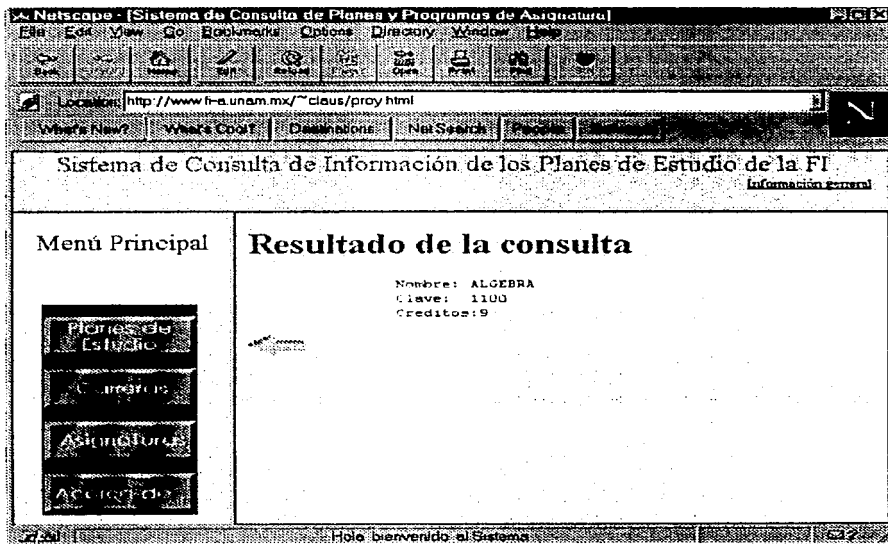
Cursos Prerrequisitos

Asignaturas Curriculares

Nivel	Sem.	Credit	Asignaturas Curriculares				
I	1	37	Álgebra (9)	Cálculo I (9)	Geometría Analítica (6)	Física Experimental (9)	Primer Examen Parcial
	2	42	Álgebra Lineal (6)	Cálculo II (9)	Estadística (9)	Química (11)	Segundo Examen Parcial
	3	40	Ecuaciones Diferenciales (9)	Cálculo III (9)	Química (6)	Termodinámica (9)	Análisis Gráfico
	4	38	Métodos Numéricos (9)	Electricidad y Magnetismo (10)	Dinámica (6)	Probabilidad (9)	Segundo Examen Parcial de la Licenciatura
	5	47	Matemáticas Avanzadas (9)	Topografía I (12)	Fundamentos de Topología (6)	Estadística (6)	Optativa y Análisis

Consulta de Planes de Estudio de la Facultad de Ingeniería. Para consultar haga clic aquí.

En la pantalla anterior al dar un clic del ratón sobre cualquiera de los cuadros que hacen referencia a cada asignatura se ejecuta la siguiente instrucción que permite acceder a la base de datos para obtener la información correspondiente, cabe señalar que el visualizador cambia la forma del apuntador del ratón al pasar por una liga, o en este caso por alguno de los recuadros que tienen asociada una liga, por ejemplo al dar un clic sobre el recuadro de la materia "ALGEBRA" se produce el siguiente resultado:



Y la instrucción en el archivo html correspondiente es:

```
<area shape="rect" coords="83,115,173,144" href="/cgi-bin/prueba.cgi
?25+94+1100" >
```

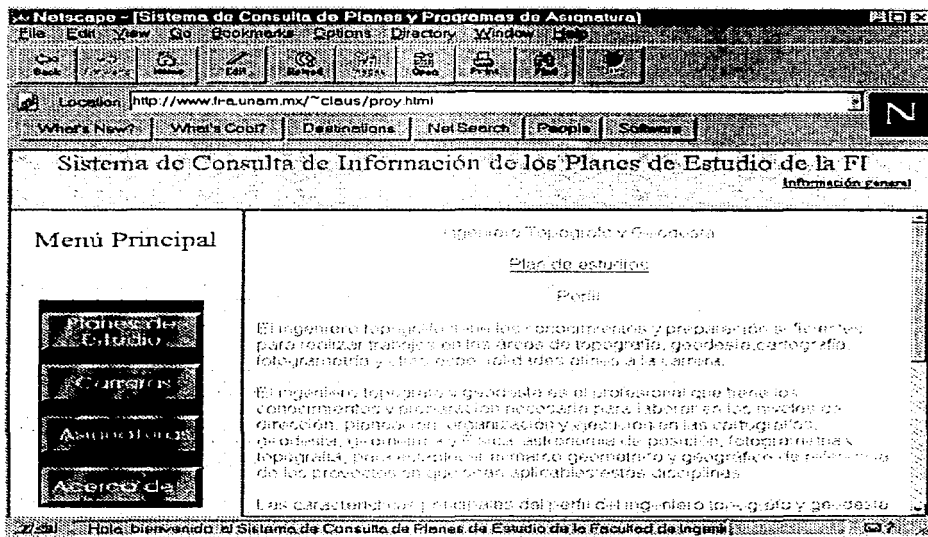
Donde se puede apreciar que se manda ejecutar un programa “cgi” que analizaremos en el siguiente tema, al cual se le pasan como parámetros; la clave de la carrera, el plan de estudios correspondiente y la clave de la materia.

Y en los resultados nos da el nombre de la materia, su clave, el número de créditos e indica si la materia lleva laboratorio o prácticas. En caso de materias optativas presenta la lista de éstas para la carrera seleccionada.

Carreras:

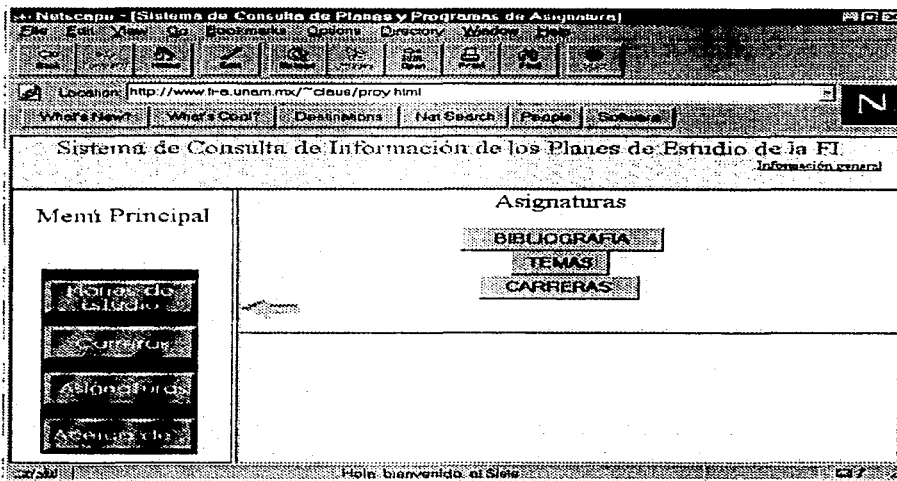
Para cada carrera se muestra un archivo con las características o bien el perfil que debe cubrir cada estudiante de la carrera correspondiente; primero se muestra una

lista de las carreras impartidas, una vez que se elige se muestra el archivo mencionado, con una liga hacia los planes de estudio tal y como se vió en el punto anterior.



Asignaturas:

Es la opción más importante dentro del sistema, ya que la finalidad es que los alumnos puedan consultar los planes y programas de asignatura, por lo que al elegir esta opción, el alumno recibe a su vez un nuevo menú para elegir qué datos quiere saber acerca de la asignatura que introduzca como cadena de búsqueda.



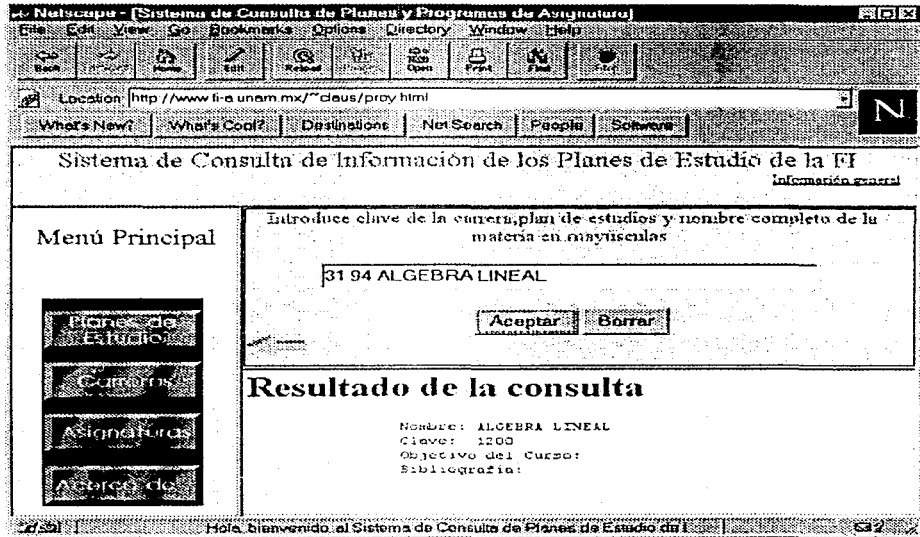
De acuerdo a la clasificación mostrada la información que se obtiene es la siguiente:

Para bibliografía: Clave de la asignatura, Nombre de la asignatura, Objetivo del curso de dicha asignatura y Bibliografía recomendada de acuerdo a los temas que se imparten en la asignatura.

Para temas: Clave de la asignatura, Número de tema, Nombre del tema, Horas empleadas por tema, Objetivo del tema y Contenido.

Para carreras: Clave de la asignatura, Nombre de la asignatura, Créditos, Si la materia incluye laboratorio o prácticas, Si es optativa y si tiene alguna materia que la preceda.

Para obtener dicha información el alumno debe introducir la información solicitada, ésta puede ser el nombre de la asignatura, la clave de la carrera e incluso el plan de estudios.



Para facilitar el “navegar” entre el sistema cada *frame* cuenta con flechas que permiten regresar entre documentos, además de que el menú principal siempre está visible para poder cambiar la acción que se está ejecutando, y se pueden utilizar todas las herramientas del visualizador para regresar, adelantar o detener el proceso de ejecución de algún documento.

Para acerca de: Se tiene información referente al sistema.

V.3.4 Mantenimiento

En cuanto al mantenimiento del subsistema de presentación dependerá realmente de los administradores y de las necesidades que se vayan generando, para hacerlo tendrán que programar en html, java o bien javascript o en cualquier otro generador de páginas Web, con las características adecuadas a la necesidades.

Para una referencia debemos decir que los archivos html se encuentran en el servidor llamado cancu.fi-a.unam.mx en el subdirectorio /users/unicafi/claus/WWW, los

archivos de imagen se encuentran en /users/unicafi/WWW/inimagen del mismo servidor, mientras que los archivos cgi se encuentran en el servidor de Web llamado cozumel.fi-a.unam.mx bajo el subdirectorío /WWW/httd/cgi-bin.

V.4 Subsistema de enlace

V.4.1 Análisis de requerimientos

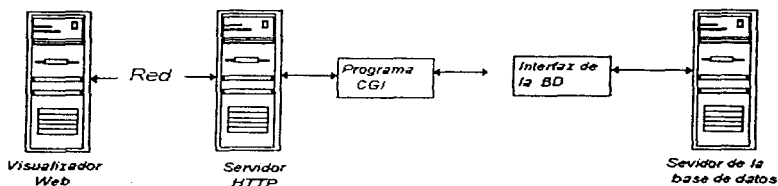
Para presentar la información contenida en la base de datos se requiere de la comunicación de los subsistemas de información y de presentación de los cuales ya se ha hablado. Este proceso de comunicación está dado por el subsistema de enlace, éste para poder funcionar requiere de un medio de acceso a la base de datos y una forma de comunicación con el visualizador que desplegará dichos datos.

Se requiere, en este caso, de un lenguaje de programación capaz de combinar el lenguaje de manipulación de datos (SQL Structured Query Language) para la base de datos como del lenguaje de formato para páginas Web (HTML), y que permita a su vez la comunicación entre diferentes máquinas, esto es, una comunicación remota.

De acuerdo con lo anterior, se requiere de un elemento o un conjunto de ellos que proporcionen dichas características, esto es, requerimos de un CGI, (Common Gateway Interface), que como su nombre lo indica es "una puerta de enlace para interfaces" y que en realidad es un conjunto de instrucciones o convenciones de comunicación entre éstas.

En el caso de nuestro subsistema de enlace se compone de dos elementos importantes; lo que es propiamente la interfaz de base de datos que es el *software* que proporciona el manejador de la base de datos para la obtención de éstos y el *gateway* (CGI) que se refiere al *software* entre Web y la ya mencionada interfaz de base de datos.

El esquema siguiente muestra lo antes mencionado:



Con respecto a la manera de interactuar entre una base de datos y un servidor de Web hay varias configuraciones dependiendo de las características de cada uno de ellos y a continuación se describen.

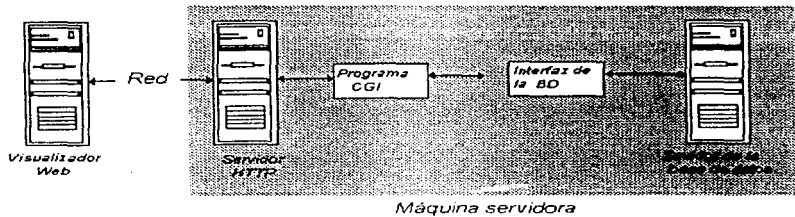
Formas de interacción de la base de datos y el Web

Utilización de una base de datos sin capacidad de redes:

En algunos casos una base de datos no tendrá capacidades cliente/servidor. Una base de datos puede ser capaz de manejar varios usuarios a la vez, pero sólo si los usuarios están en la máquina donde reside el servidor de la base de datos. Algunos proveedores de bases de datos ofrecen capacidades de redes como característica adicional, y cobran un cargo de licencia por cada máquina que tenga acceso a la base de datos, en red.

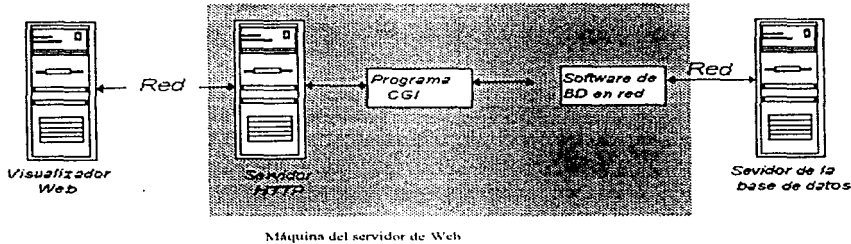
Ejecutar el servidor de HTTP en la máquina donde reside el servidor de base de datos puede aminorar el problema, pero sólo es posible si la máquina servidor tiene capacidad para manejar la doble carga. El servidor HTTP, los programas CGI y el servidor de la base de datos están ubicados en la misma máquina, y de esta forma los programas CGI pueden ser ejecutados para tener acceso a la interfaz de la base de datos que se desea utilizar. Las consultas a la base y los resultados devueltos no van y vienen a los programas CGI a través de la red, y de esa manera el tiempo de respuesta es óptimo.

Sin embargo, este método supone que los visualizadores Web que están accediendo la información deben estar en máquinas remotas. Si la base de datos es de mediano o gran tamaño, y hay muchos usuarios, definitivamente no es buena idea tener el servidor de base de datos y el servidor HTTP ejecutándose en la misma computadora.



Utilización de una base de datos con capacidad de redes:

Con este método los programas CGI se ubican en la misma máquina que el servidor HTTP y por otro lado tenemos al servidor de base de datos, esto debido principalmente a que los manejadores de bases de datos consumen demasiados recursos y de hecho así liberan de carga al servidor, para ello se requiere que el *software* de la base de datos sea responsable del soporte de redes y que éste se instale en cada máquina que tenga acceso a la base.



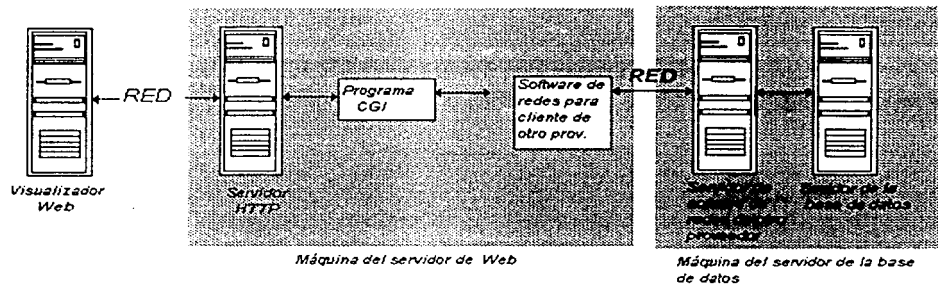
Utilización de software de redes de otro proveedor:

Otra forma de tener acceso a una base de datos desde un servidor de Web es utilizar *software* de redes de otro proveedor, o inclusive de *software* desarrollado por nosotros mismos.

Este *software* debe ser capaz de aceptar consultas de un programa CGI residente en la misma máquina; transferirlas por la red, a la otra mitad del *software* de redes;

hacer llegar las consultas a la base de datos; obtener los resultados, y regresarlos de nuevo a través de la red, para entregarlos al programa CGI.

Además, el *software* de redes también puede utilizarse como un *software* de traducción para la base de datos. Las consultas de la base de datos pueden convertirse en conjunto de instrucciones sencillas y ahora no importará el tipo de base de datos que se utilice.



Para utilizar una “puerta de enlace” (*gateway*) es necesario saber qué tipo de interfaz proporciona la base de datos. Sea la interfaz más sencilla, de línea de comandos, o la más compleja, la base de datos debe proporcionar alguna manera de obtener datos.

V.4.2 Diseño

Se construirá un programa CGI (Common Gateway Interface), que es un programa basado en un conjunto de variables y convenciones nombradas CGI de manera común y que se utilizan para transferir información en ambos sentidos entre una máquina servidora y una cliente o bien entre una aplicación servidora y una cliente. La acción de llamar un programa CGI desde un visualizador Web es muy sencilla para el usuario, sin embargo, desde la perspectiva del programador, el proceso es más complicado. Existen varios pasos a seguir para que un programa CGI funcione adecuadamente:

1. El usuario llama un programa CGI haciendo clic sobre un vínculo u oprimiendo un botón.
2. El visualizador Web solicita autorización al servidor de Web para ejecutar el programa CGI.

3. El servidor Web revisa la configuración y los archivos de acceso para asegurarse que el solicitante tiene permitido el acceso al programa CGI.
4. El servidor Web se asegura de que exista el programa CGI.
5. Si existe el programa, éste se ejecuta.
6. Cualquier resultado producido por el programa CGI se devuelve al visualizador Web.
7. El visualizador Web despliega el resultado.

Los programas CGI pueden escribirse prácticamente en cualquier lenguaje. Los criterios para el lenguaje son los siguientes:

- 1) El lenguaje debe estar soportado por el sistema operativo en el que esté corriendo la aplicación servidora, en este caso el servidor de la página Web.
- 2) El lenguaje debe tener facilidades necesarias para realizar las tareas que se necesiten de él; en nuestro caso la comunicación y la ejecución de consultas a la base de datos.
- 3) Tener familiaridad con el lenguaje.

Un programa CGI tiene tres funciones básicas de entrada-salida:

- 1) Recabar la entrada del servidor en forma de variables estandarizadas, datos de formularios y datos de consulta.
- 2) Proporcionar datos de salida al cliente.
- 3) Proporcionar información de negociación del contenido.

Cabe mencionar que el término “cliente” se refiere al visualizador de Web.

Para que los programas CGI puedan funcionar en un sistema, el servidor Web se debe configurar de manera que permita accesos a éstos. Algunos administradores de sistemas consideran el acceso con CGI como un posible riesgo de seguridad, y niegan la posibilidad a todos los usuarios de utilizarlo. En otros sistemas, el acceso con CGI se otorga a ciertos usuarios de manera exclusiva, y sólo los programas en CGI escritos por usuarios actualizados pueden ser ejecutados por el servidor.

CGI es una interfaz de doble sentido. Los datos pueden transferirse al programa CGI para su procesamiento y los programas CGI también pueden devolver información al visualizador Web para su presentación. Los programas CGI proporcionan a los programadores un modo de hacer páginas de Web interactivas. La información introducida por el usuario puede afectar el comportamiento del programa CGI, y los resultados devueltos por el programa son resultado directo de lo que introduce el usuario.

Un visualizador Web carga un documento HTML y lo despliega. En algunos casos se ejecuta un programa CGI cuando se carga la página, y los resultados se despliegan como parte de la página Web. En otros casos, el usuario oprime un botón o hace clic en un vínculo para activar un programa CGI y los resultados se despliegan como parte de otra página Web.

En la mayoría de los casos, se desea devolver alguna información al visualizador para que éste la presente. Para ello se necesita que el programa CGI devuelva un encabezado como primera línea de resultado. El encabezado le indica al visualizador Web cómo desplegar el resultado del programa CGI.

La información devuelta casi siempre es algún tipo de texto, sea sencillo o HTML. El texto sencillo se despliega de la manera normal. El tipo HTML como cualquier documento HTML.

Al encabezado debe seguirlo una línea en blanco que le indica al visualizador que la sección de encabezado terminó.

Dentro de los programas y convenciones que se pueden utilizar por los servidores de Web para incluir resultados de expresiones, archivos y programas dentro del documento HTML que se envía al cliente se tienen los llamados SSI (Server Side Includes) "Inclusiones del lado del servidor".

Los *scripts* (CGI) envían su salida hacia la salida estándar "stdout". Esta salida puede ser un documento generado por el *script* o bien, instrucciones para que el servidor obtenga la salida deseada.

Normalmente los *scripts* producen una salida, la cual es interpretada y enviada hacia el cliente. Una ventaja de esto es que el *script* no necesita enviar el encabezado de HTTP/1.0 para cada petición.

Algunos *scripts* pueden necesitar de una atención extra por parte del servidor, para que éste revise su salida y haga los llamados directamente al cliente. Para poder distinguir estos *scripts* de los otros CGI requiere que el nombre del *script* empiece con "nph-", (Non-Parsed Header) "Encabezado no analizado" con lo cual el servidor

no revisará su encabezado, y quiere decir que el *script* es responsable de regresar una respuesta válida al cliente.

Cuando alguien utiliza un visualizador para tener acceso a una base de datos, hay varios componentes que intervienen para transferir la consulta del usuario a la base de datos y devolver los resultados al visualizador. Los pasos que se ejecutan son los siguientes:

1. El usuario llama un programa CGI, generalmente haciendo un clic en un vínculo, u oprimiendo un botón en el visualizador Web.
2. El visualizador reúne toda la información escrita por el usuario para enviarla al programa CGI.
3. El visualizador contacta al servidor HTTP en la máquina donde reside el programa CGI, pidiéndole que localice a este último y le transfiera la información.
4. El servidor HTTP corrobora si la máquina solicitante tiene autorización de acceso al programa CGI.
5. Si el usuario tiene acceso, el servidor HTTP localiza el programa CGI y transfiera a éste la información del visualizador Web.
6. Se ejecuta el CGI.
7. El proceso del CGI convierte la información recibida a un formato que la base de datos sea capaz de entender.
8. El CGI usa el módulo de la base de datos para transferir la consulta a la interfaz de la base de datos.
9. La interfaz de la base de datos analiza la sintaxis de la consulta para asegurar que sea precisa.
10. Si la interfaz encuentra un error de sintaxis en la consulta, se envía un mensaje de error al programa CGI.
11. El mensaje de error se envía al servidor HTTP, el cual lo transfiere al visualizador Web para que éste lo despliegue al usuario. El proceso se detiene aquí.
12. Si no hay error, la interfaz envía la consulta a la base de datos.

13. La base de datos atiende la consulta y devuelve los resultados al programa CGI a través de la interfaz.
14. El programa CGI da formato a los resultados y los envía al servidor, por medio del CGI mismo, para su envío posterior al visualizador de Web.
15. El visualizador despliega los resultados.

Las interfaces proporcionan un método para interactuar con una base de datos. Los comandos SQL que crean tablas, insertan, actualizan o borran datos, devuelven resultados de una consulta, y son transferidos al motor de la base de datos para su procesamiento y, posteriormente, devolver los resultados. Los proveedores de bases de datos ofrecen interfaces de variados niveles de complejidad dependiendo de las necesidades.

Los CGI reciben datos transferidos desde un navegador Web mediante un servidor HTTP y los convierte a un formato que la base puede entender. La información convertida se transfiere a la interfaz de la base y ésta la ejecuta.

Usar una interfaz de base de datos requiere de un lenguaje de consultas, que en la mayor parte de los manejadores de bases de datos es una variación del Lenguaje Estructurado de Consultas, SQL (Structured Query Language). SQL es un lenguaje estándar de consultas, pero la mayoría de los proveedores le han agregado extensiones ajenas a la estándar, para aprovechar capacidades específicas ofrecidas por su manejador de base de datos.

SQL se utiliza para interactuar con la base de datos. SQL permite comunicarse con la base de datos y decirle qué hacer con los datos y sus principales comandos son:

1. **CREATE** crea una nueva base de datos, así como cualquier objeto que esté contenido dentro de ésta, tales como tablas, vistas, índices, etc. El comando CREATE registra una base de datos en el servidor de base de datos.
2. **DELETE** elimina registros de la base de datos.
3. **DROP** elimina bases de datos, u objetos contenidos en ésta, como tablas, índices y vistas.
4. **INSERT** inserta datos a la base de datos.

5. **SELECT** muestra datos seleccionados de la base de datos y es el comando más utilizado para consultas.
6. **UPDATE** actualiza o cambia datos existentes en la base de datos.

De hecho muchos *gateways* se limitan a reconocer exclusivamente la declaración **SELECT**. Esto evita la posibilidad de que por un descuido se borre la base de datos.

Tal será el caso del *gateway* que se construye para el subsistema de enlace y para el cual emplearemos el esquema de la utilización de una base de datos con capacidades de red. Así como se utilizará **SQL*Plus** como interfaz junto con **Pro*C**, que en este caso es el software que proporciona Oracle para interactuar con la base de datos.

V.4.3 Implantación

Para poder llevar a cabo la generación del o los CGI requeridos para la comunicación entre los subsistemas de información y de presentación se debe asegurar la utilización de éstos en el servidor de Web.

Para configurar el servidor Web de manera que reconozca *scripts* CGI, se necesita agregar la siguiente línea al archivo `srn.conf` ubicado en el directorio de configuración HTTP:

```
ScriptAlias /cgi-bin/ /serverpath/httpd/cgi-bin/
```

Donde `serverpath` es el directorio donde reside el *software* del servidor HTTP. Esta línea indica al servidor HTTP reconocer programas residentes en el directorio `/cgi-bin`, ubicado dentro del directorio principal del servidor HTTP. Si se permite a los programas CGI estar en otra parte que no sea el designado directorio principal CGI, se deben agregar los directorios de la misma manera, utilizando la seudoinstrucción `ScriptAlias`.

Todo esto para un servidor de Web que ejecute la versión construida por NCSA Centro Nacional de Aplicaciones para Supercómputo (National Center for Supercomputing Applications).

La mayoría de los servidores implantan una forma especial de llamar a los programas CGI, a través de ello es cuando se puede interceptar el encabezado `Non Parsed Header` o `NPH-CGI`. En la mayoría de los servidores se invoca cuando el programa CGI se encuentra en un archivo cuyo nombre empieza con las letras "nph".

Y como ya se había mencionado estos son un tipo de CGI que se encarga de regresar una respuesta válida al cliente, es decir, que el servidor no analizará su encabezado.

Actualmente existen principalmente 3 directivas del servidor que se especifican en los encabezados de los CGI:

- Content-type

Esto es el tipo MIME del documento que está regresando el *script*.

MIME (Multipurpose Internet Mail Extensions) y que fue creado originalmente para permitir que los mensajes de correo electrónico incluyeran tipos de medios diferentes al texto sencillo. Web también utiliza el estándar MIME para determinar el manejo de multimedia.

- Location

Es usado para especificar al servidor que se está regresando una referencia a un documento distinto al actual. Si el argumento es un URL, el servidor lo redireccionará hacia el cliente. Si el resultado es una ruta virtual, el servidor regresará el documento especificado como si el cliente hubiera solicitado el mencionado documento de manera original.

- Status

Es utilizado para proporcionar una línea de estado al servidor con respecto a HTTP/1.0, de tal forma que le informe de ello al cliente. El formato es "nnn xxxxx", donde "nnn" representa el código de estado en 3 dígitos, y "xxxxx" es la cadena de mensaje.

Ejemplos:

Content-type: text/html Indica que el CGI genera texto HTML

location: /users/doc.txt Ruta Virtual

location:http://www.fi-a.unam.mx/~claus/uno.txt
URL (Uniform Resource Location)

500 Server error Status

Con base en lo anterior nuestro subsistema de enlace consta de un CGI que es un programa un shell script de Unix y el programa interfaz con la base de datos

realizado en el lenguaje de programación propio de Oracle que permite generar ejecutables para efectuar acciones sobre sus bases de datos, Pro*C, como ya se mencionó anteriormente, en nuestro caso lo importante es realizar consultas, dichas consultas tienen que ver con las asignaturas de cada uno de los planes de estudio. Uno de los principales problemas que habíamos mencionado era el poder conjuntar el software de manejo de consultas de la base de datos Oracle con el lenguaje comandos o scripts que puede manejar el visualizador, con Pro*C, esto se simplifica ya que conjunta los elementos necesarios del lenguaje estructurado de consultas SQL, y puede interactuar ya que al generar archivos ejecutables puede ser ejecutado desde un *shell script* de Unix, recibir parámetros y permitir conexiones remotas entre los servidores que intervendrán en el sistema de consulta (Servidor de Web y Servidor de base de datos).

Mencionaremos a continuación la estructura que tiene un program en Pro*C.

Un programa en Pro*C contiene dos partes, ambas son requeridas por el procesador.

Prólogo de la aplicación

Cuerpo de la aplicación

El prólogo de la aplicación define variables y hace una preparación general para el programa en Pro*C. El cuerpo de la aplicación básicamente contiene llamadas al compilador de Pro*C incluyendo enunciados Insert y Update, para manipular datos de Oracle.

En la sección de prólogo de aplicación podemos encontrar a su vez tres secciones:

- 1.- Sección DECLARE, en donde se declaran las variables a utilizar en el programa, las llamadas variables host que hacen referencia a campos de una tabla en una consulta, entre otras cosas.
- 2.- INCLUDE SQLCA es el área de comunicación, la cual provee de un manejo de errores a través de las bibliotecas de Oracle que se pueden incluir.
- 3.- El enunciado CONNECT que permite establecer una sesión con el manejador de bases de datos.

Mientras que por otra parte la sección del cuerpo de la aplicación contiene enunciados que consultan y manipulan los datos almacenados en una base de datos Oracle, tales como el enunciado SELECT el cual permite hacer consultas para devolver uno o bien varios renglones de una tabla y que es en sí, el enunciado que nos interesa más para el sistema de enlace.

Con base en lo anterior y de acuerdo a las opciones de consultas planteados en el subsistema de presentación tenemos que existen 5 consultas principales, para los mapas sensitivos de los planes de estudio, para las materias optativas de los mismos, para la opción de bibliografía, para la opción de temas y para la opción de carreras de la opción de asignaturas.

Se mostrará a continuación la estructura de los programas cgi y programas de consulta a la base de datos.

De acuerdo al esquema de acceso que se implanta en el sistema; tenemos un programa cgi que conectará al visualizador con un programa que a través de un lenguaje de consultas accederá a la base de datos.

Los programas cgi están programados en Kshell de unix y permiten hacer una llamada al programa de acceso a la base de datos que está programado en Pro*C que es un lenguaje de programación que combina las características de un lenguaje de programación estructurada como lo es 'C' con las ventajas de un lenguaje estructurado de consultas como lo es SQL, haciendo de esta forma archivos ejecutables que acceden a la base de datos sin necesidad de tener una sesión interactiva en el manejador de base de datos para poder hacer consultas a ésta.

Programa cgi:

```
#!/bin/ksh

echo Content type:text/html
echo
eval `S*`
cat <<EOM
<html>
<head>
<body background = "http://www.fi-a.unam.mx/~claus/imagen/back2.jpg"
text="800000">
<h1>resultado de la consulta</h1>
<pre>
`remsh 132.248.54.3 /users/unicafi/claus/corre $*`
<a href="javascript:history.go(-1)"></a>
</body>
</html>
EOM
```

En un archivo cgi se debe de especificar de qué manera se esperan los resultados, para nuestro caso serán archivos html:

```
echo Content type:text/html
```

Lo principal de este archivo cgi es primero la recepción de los parámetros enviados a través del visualizador, y el llamado al programa de acceso a la base de datos:

```
eval `S*
```

```
`remsh 132.248.54.3 /users/unicafi/clus/corre S*
```

Por otro lado el programa "corre2 es a su vez otro script que llama al archivo en Pro*C

```
corre
```

```
#!/bin/ksh
```

```
# Set up the search paths:
```

```
PATH=$PATH:/users/oracle/bin:./
```

```
# Set up the shell environment:
```

```
# set -u
```

```
# trap "echo 'logout'" 0
```

```
ORACLE_HOME=/users/oracle;export ORACLE_HOME
```

```
ORACLE_TERM=hp;export ORACLE_TERM
```

```
ORACLE_SID=CURSOS; export ORACLE_SID
```

```
MENUPATH=$ORACLE_HOME/forms30/admin/resource
```

```
export MENUPATH
```

```
ORATERMPATH=$ORACLE_HOME/forms30/admin/resource
```

```
export ORATERMPATH
```

```
/users/unicafi/clus/proc/materia S*
```

Aquí se definen las variables de ambiente adecuadas para el correcto acceso al manejador de bases de datos y el llamado al archivo materia, que devolverá la información de la materia a la cual se le dio un clic del ratón sobre el plan de estudios sensitivo. Y cuyo contenido se analiza a continuación:


```
#include <stdio.h>
```

```
EXEC SQL BEGIN DECLARE SECTION;
```

```
    varchar uid[20];  
    varchar pwd[20];  
    char asigna[100];  
    char lab_prac[8];  
    char optativa;  
    char lab_prac[8];  
    char labo[5];  
    varchar clave[3];  
    varchar carr[2];  
    varchar plan[2];  
    short sal,i,creditos,carrera;
```

```
EXEC SQL END DECLARE SECTION;
```

```
EXEC SQL INCLUDE sqlca;
```

```
main(int argc, char *argv[])
```

```
{  
    strcpy(uid.arr,"claudia");  
    uid.len=strlen(uid.arr);  
    strcpy(pwd.arr,"claus");  
    pwd.len=strlen(pwd.arr);  
    strcpy(clave.arr,argv[3]);  
    clave.len=strlen(clave.arr);  
    strcpy(plan.arr,argv[2]);  
    plan.len=strlen(plan.arr);  
    strcpy(carr.arr,argv[1]);  
    carr.len=strlen(carr.arr);  
    sal=atoi(clave.arr);  
    carrera=atoi(carr.arr);
```

```
EXEC SQL CONNECT :uid IDENTIFIED BY :pwd;
```

```
EXEC SQL SELECT asignatura.nom_asigna,
```

```
    carr_asigna_plan.creditos,
```

```
    carr_asigna_plan.optativa,carr_asigna_plan.lab_prac
```

```
INTO :asigna,:creditos,:optativa,:lab_prac
```

```
FROM ASIGNATURA,carr_asigna_plan
```

```
WHERE (asignatura.clv_asigna=:sal) and ( ( carr_asigna_plan.clv_carrera=:carrera)
```

```
and (carr_asigna_plan.plan=:plan)
```

```
and (carr_asigna_plan.clv_asigna=:sal));
```

```

printf("\t\tNombre:\t");
for(i=0;(asigna[i]!=' ')||(asigna[i+1]!=' ');i++){
printf("%c",asigna[i]);}
printf("\n\t\tClave:\t%d",sal);
printf("\n\t\tCreditos:%d\n",creditos);
printf("%c",optativa);
for(i=0;(lab_prac[i]!=' ') &&(lab_prac[i]!='\0');i++)
{
labo[i]=lab_prac[i];}
strcat(labo,'\0');
if(!strcmp(labo,"L"))
printf("Asignatura con laboratorio incluido");
else if(!strcmp(labo,"L+"))
printf("Asignatura con laboratorio y su inscripcion es independiente");
else if(!strcmp(labo,"L*"))
printf("Asignatura con laboratorio incluido, requisito sin valor en creditos");
else if(!strcmp(labo,"P"))
printf("Asignatura con practicas incluidas");
else if(!strcmp(labo,"P+"))
printf("Asignatura con practicas y su inscripcion es independiente");
else if(!strcmp(labo,"P*"))
printf("Asignatura con practicas de fin de semestre sin valor en creditos");
else if(!strcmp(labo,"Pr"))
printf("Asignatura con proyecto incluido");
else if(!strcmp(labo,"PL"))
printf("Asignatura con practicas y laboratorio incluido");
else if(!strcmp(labo,"PP+"))
printf("Asignatura con practicas incluidas y con practicas de inscripcion indepe
ndiente");
else if(!strcmp(labo,"LPr"))
printf("Asignatura con laboratorio y proyecto incluido");
else if(!strcmp(labo,"*"))
printf("Asignatura de la Division de Estudios de Posgrado");
}

```

En el archivo se ven claramente las diferentes secciones que debe contener un programa en Pro*C, primero las bibliotecas de 'C' que serán incluidas, después la declaración de las variables a través de las cuales se le pasarán los parámetros a la consulta así como las variables en las cuales se almacenarán los resultados de las mismas, después se encuentra la sección de conectividad, la clave y el password a través de los cuales se tienen permisos de acceso a la base de datos.

Después se incluye la consulta, que es exactamente como hacer una consulta en forma interactiva con SQL*Plus de Oracle, sólo que aquí se debe incluir la cláusula INTO para indicar en qué variable se almacenará el resultado de la consulta para cada campo solicitado, como en el archivo anterior la consulta es:

```
EXEC SQL SELECT asignatura.nom_asigna,
carr_asigna_plan.creditos,
carr_asigna_plan.optativa,carr_asigna_plan.lab_prac
INTO :asigna,:creditos,:optativa,:lab_prac
FROM asignatura,carr_asigna_plan
WHERE (asignatura.clv_asigna=:sal) and ( ( carr_asigna_plan.clv_carrera =
:carrera)
and (carr_asigna_plan.plan=:plan)
and (carr_asigna_plan.clv_asigna=:sal));
```

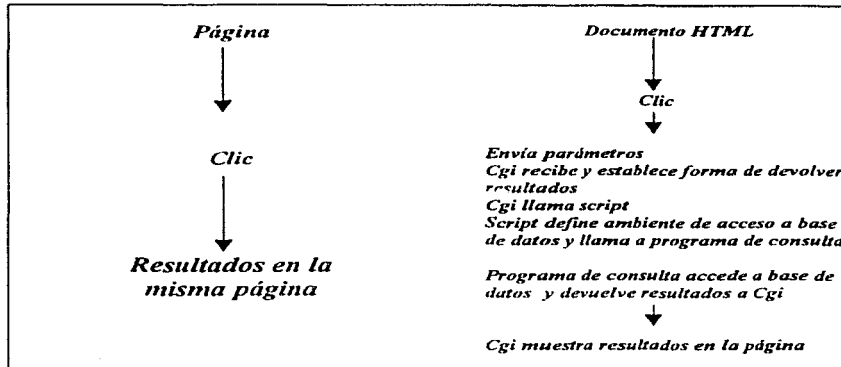
Lo solicitado en esta consulta es: Nombre de la asignatura, créditos de la asignatura, si la asignatura es optativa, si incluye laboratorio o prácticas. Lo almacenará en las variables asigna,creditos, optativa y lab_prac, las tablas a consultar son asignatura y carr_asigna_plan y devolverá el resultado sólo cuando el registro cumpla que la clave de la asignatura sea igual al número que almacena la variable sal, y que la carrera coincida con el valor de la variable carrera y el plan de estudios sea el que almacena la variable plan, estas variables realmente contienen los parámetros enviados desde el visualizador a través de la posición en la que está el ratón al momento de dar un clic.

```
<area shape="rect" coords="83,115,173,144" href="/cgi-bin/prueba.cgi
?23+94+1100">
```

En donde ?23+94+1100 indican que al programa prueba.cgi se le pasan como parámetros 23,94 y 1100 y que para el programa materia 23 será la carrera, 94 el plan y 1100 la clave de la asignatura almacenada en la variable sal.

Una vez almacenados los resultados en variables se escriben en la pantalla con la instrucción "printf" de 'C'.

El proceso es el mismo para cada uno de las consultas del sistema, se tiene un documento html que muestra la interfaz gráfica con el usuario, éste ejecuta una acción; da un clic sobre una zona sensitiva o introduce valores, esto provoca la ejecución de un cgi, que captura los parámetros y que su vez llama a otro programa que establece las variables de ambiente adecuadas para el acceso a la base de datos, la cual es consultada a través de un programa en Pro*C que devuelve los resultados a su vez al programa cgi, quien despliega los resultados con formato html para que el usuario vea los resultados como parte integral de la página en la que está trabajando.



V.4.4 Mantenimiento

Para el mantenimiento del subsistema de interfaz es necesario programar las consultas o bien modificar los programas ya hechos y llamarlos desde los archivos html correspondientes a la interfaz gráfica desde donde se ejecutarán tal y como se mostró en el punto referente a la implantación del subsistema de presentación y de interfaz.

Los scripts que definen el ambiente adecuado para acceder a la base de datos, así como los programas ejecutables y de código de Pro*c se encuentran en el servidor fork.fi-a.unam.mx bajo los subdirectorios /users/unicafi/claus y /users/unicafi/claus/proc respectivamente, cabe mencionar que estos archivos deben tener permisos de ejecución para ser ejecutados por cualquier usuario.

Los permisos pueden ser dados con un comando de unix de la siguiente forma:

Schmod 755 archivo

en donde el 755 es el número en hexadecimal que corresponde a tener todos los permisos por parte del dueño del archivo y permisos de ejecución para los usuarios del sistema en general.

Conclusiones

Conclusiones

Durante el proceso de investigación para el desarrollo del sistema he comprobado el desmesurado incremento en el uso de la "red de redes" (Internet), provocando con ello también el acelerado desarrollo de herramientas de diseño de páginas Web que permite llegar a un mercado mundial impresionante.

Las herramientas de desarrollo al inicio del presente trabajo aunque ya existían eran aún demasiado sofisticadas y poco publicitadas, ahora cabe mencionar, que todos aquellas herramientas de desarrollo que no cuenten con la facilidad de generar páginas, formas o bien que sus aplicaciones no puedan ejecutarse a través de un "navegador", *browser* o bien visualizador como querramos decirle, es una herramienta atrasada.

Es por ello que las compañías de desarrollo de manejadores de bases de datos se han preocupado por implantar en sus productos rutinas de generación de formas de captura, de actualización de sus bases de datos, y he aquí la importancia que a mi parecer tiene el presente trabajo.

Sabido es que la información en la actualidad es el elemento más valioso en todos los niveles de la vida del hombre, ésta junto con las comunicaciones, han revasado todas las fronteras; el desarrollo cultural, económico, político y en todos los aspectos de la humanidad está supeditado a la información que de todas partes del planeta y aún fuera de él generamos y por supuesto manejamos o por lo menos sabemos que existe; es por ello que el saber que en alguna parte del mundo existe la información que yo requiero y que tal vez yo cuento con información que alguien más pueda utilizar, da pie a esta transferencia de información que cada vez se complica más, aunque claro tiene grandes ventajas, las más de las cuales de tipo comercial y es así que podemos hablar de hacer movimientos bancarios, de ventas, reservaciones, o bien para la investigación podemos acceder a las bases de datos de información de revistas, bibliotecas, librerías, universidades, etc. a través de medios como el correo electrónico, gopher y el Web, este último el más amigable y ahora más popular.

El punto es una vez que se coloca la información de manera que todos puedan acceder a ella, cómo actualizarla al mismo ritmo que cambia en la realidad, cómo permitir el cambio de un número confidencial para una cuenta bancaria, cómo incluir un libro más al catálogo de una biblioteca sin alterar el formato que actualmente presentan los datos. La solución a los problemas de información ya estaba dada desde hacía tiempo a través de las bases de datos y ahora el reto era integrarlas al medio de transferencia ya mencionado, el Web.

El problema no es tan complicado si se analiza de la manera tradicional en donde el manejador de bases de datos es el "*back end*" y necesitamos de un "*front end*" para

ver la información de forma amigable, el punto en este caso es permitir que el visualizador sea un *front end* válido para el *back end*.

En una palabra el problema es de "conectividad", es decir, conectar un manejador de bases de datos, en nuestro caso Oracle con el visualizador de Web en este caso Netscape; la forma de solucionar esto era a través de programas que hicieran la vez de interfaces, los llamados CGI "Common Gateway Interface", dichos programas había que hacerlos en los *shells* de los sistemas operativos bajo los que estuvieran los servidores de Web o bien en lenguajes de programación compatibles, como 'C', también en los lenguajes que se habían ido desarrollando a la par de Internet, como Perl o javascript.

Sin embargo, el problema seguía siendo el acceso a las bases de datos, esto se hacía a través de métodos rudimentarios como las consultas prefabricadas o procedimientos almacenados, o programas generados en lenguajes que combinaran elementos de lenguajes de programación estructurada con SQL, en nuestro caso Pro*C.

Todo ello da una idea de lo complicado y laborioso que el mantener una página actualizada al ritmo de la actualización de bases de datos representa.

Ahora el panorama se torna más sencillo y abundante en posibilidades ya que con los productos incluidos dentro de los mismos manejadores de bases de datos, el generar formas de captura en páginas html, con seguridad implantada a nivel de usuario de la base de datos, así como la seguridad de transferencia que permiten los métodos de encriptación de los propios manejadores como del visualizador, nos dan la posibilidad de crear aplicaciones cada vez más complejas en tiempos menores ya que la interface con el Web está integrada al manejador, es decir, ya no requerimos de programar en html, en shells o en otros lenguajes, el generar las tablas en la base de datos y crear las consultas es suficiente para que a ellas se agreguen sus formas en el Web.

Por otro lado considerando los recursos con los que cuentan tanto la Unidad de Servicios de Cómputo Académico, como la Facultad de Ingeniería las alternativas planteadas siguen siendo válidas y siguen dando pie a la misiva de obtener más de lo menos, además de ser un reto para aquellos interesados en este tipo de aplicaciones, que espero sean muchos ya que se tiene mucho por hacer.

El sistema de consulta de información a través del Web de Internet bajo ambiente de bases de datos, es un sistema pensado en los alumnos de la Facultad de Ingeniería para la consulta de sus planes y programas de estudio, así como para todos aquellos que requieran de dicha información. Es un trabajo realizado como parte de las actividades de la Unidad de Servicios de Cómputo Académico cuyo compromiso así

como el de sus integrantes es el de brindar servicio a la institución a la cual pertenecemos.

El compromiso para con este sistema, es que continuará con su desarrollo aún después de la presentación del presente trabajo con el propósito de integrarlo a un sistema donde se dispondrá de otros los servicios que brinda la Facultad como el proceso de inscripción o consulta de su acervo bibliográfico.

Por mi parte agradezco el apoyo brindado para la realización de este proyecto, a través del cual he puesto en práctica todo lo aprendido a lo largo de la estancia en la carrera y aún más he aprendido el compromiso que todo profesionista debe tener para la realización de cualquier proyecto.

Apéndice A

APENDICE A

Modelos de Bases de datos : Jerárquico y de Red

En la actualidad el buen manejo de la información es primordial para el buen funcionamiento de una empresa o institución.

La manera de manipular esta información en equipos de cómputo ha sido muy diversa a través de la evolución de éstos, generando la necesidad de usar nuevas técnicas de optimización en su organización. Cuando los volúmenes de información empezaron a ser muy grandes y difíciles de manejar surge la necesidad de crear un *software* capaz de proveer de un método eficaz para el control y mantenimiento de dichos volúmenes de información. Es entonces cuando aparece el concepto de "Bases de datos" para denominar a toda la información y aparecen los "manejadores de bases de datos" los cuales deberían ver la información de manera global.

Ante esto se dieron a conocer diversos manejadores de bases de datos algunos desarrollados por proveedores líderes y otros por prestigiadas casas de *software*.

Estos DBMS Sistemas Manejadores de Bases de Datos (DataBase Management System) fundamentaron su funcionamiento en diferentes modelos de datos, siendo los tres más aceptados y conocidos, los modelos jerárquico, de red y relacional.

Reseña histórica

Con el avance de la tecnología de cómputo la necesidad de manejar la información en bases de datos se hizo más necesaria y su evolución se puede ver a través de ciertos acontecimientos, de los cuales hablaremos a continuación:

Una base base de datos puede considerarse como la más moderna técnica de almacenamiento de datos, que empezó con las tarjetas perforadas de Herman Hollerit, en 1880. Hollerit se encontró con el problema de completar el censo de 1880 de trece millones de norteamericanos antes de 1890, cuando vencía el siguiente censo. A partir del conocimiento del uso de las tarjetas perforadas en telares de Jacquard, el Dr. Hollerit inventó un método de almacenamiento de información basado en ellas. De esta forma empezó la era de los archivos de tarjetas mecanizadas, que se emplearon para el almacenamiento de la información durante los siguientes 60 años.

La primera computadora electrónica fue operativa en 1946. La ENIAC fue diseñada en la Universidad de Pennsylvania por los profesores Eckert y Mauchley para el

Departamento de la Defensa de Estados Unidos, fundamentalmente para calcular trayectorias y tablas de tiro.

En 1951 se construyó una nueva computadora llamada Univac-1 diseñada por las mismas personas para el censo de 1950 y para lo cual se requería tener un medio de almacenamiento más rápido, la cinta magnética cubrió dichos requerimientos. De hecho la Univac-1 ya contaba con un lector de cinta magnética y también motivaron la aparición de los archivos en cinta magnética.

Toda la terminología de sistemas de tarjetas, como archivos, registros y campos, se trasladó al sistema de cinta magnética. Los sistemas de procesamiento de datos usados en los años cincuenta eran subsistemas de nóminas muy sencillos, diseñados aisladamente y con independencia de otros subsistemas relacionados.

Los ficheros que utilizaban eran relativamente grandes, y el problema de escribir largas descripciones de datos, en cada programa para cada archivo, se resolvió con el verbo COPY de Cobol, que permitía a un programa copiar una descripción general de los datos previamente escritos en un archivo. A partir de entonces comenzó la historia de la evolución de las bases de datos.

La introducción de los discos magnéticos a mitad de los sesentas dio el empujón final a esta integración. Para acceder a un registro de una cinta magnética es necesario examinar secuencialmente todos los registros; mientras que en un disco es posible acceder a un registro directamente.

A mediados de los sesenta, el concepto de Sistema Manegador de Información (MIS Management Information System) ganó actualidad. El enfoque básico era la ejecución de los programas del paquete MIS sobre los archivos de salida por todos los sistemas pertinentes; pero pronto se encontró que, para una gran organización, el número de archivos de entrada del paquete MIS era excesivamente grande.

Un producto sobresaliente de aquella época fue el Integrated Data Store (IDS) Almacén de Datos Integrados, introducido por la General Electric en 1965. Como su nombre lo indica, el IDS se utilizó para crear grandes archivos integrados, que pueden ser compartidos por cierto número de aplicaciones.

Al IDS siguieron pronto otros paquetes MIS, basados en los archivos integrados para sistemas jerárquicos. El problema fue la escasez de coordinación entre los archivos de los sistemas jerárquicos. Pronto se dieron cuenta que lo que se necesitaba era una base de datos que contuviera una colección integrada de datos generalizada.

idealmente para todos los sistemas de una organización utilizando todos los programas de aplicación.

En el mercado de los años setentas aparecieron cierto número de sistemas manejadores de bases de datos, fundamentados en distintos modelos de éstos. El Codasyl se interesó por las bases de datos a finales de los sesentas y creó un grupo de trabajo para definir un modelo común de datos, ahora llamado modelo Codasyl o modelo de red.

Paralelamente al desarrollo del Codasyl se tienen otras ideas basadas en conceptos matemáticos. El logro fundamental de la investigación fue el modelo relacional debido al Dr. Edgar F. Codd que fue propuesto por primera vez en 1970.

Los principales productos existentes de los primeros modelos son:

IMS (Information Management System, Sistema Manejador de Información) de IBM basado en el modelo jerárquico.

ADABAS de Software AG., de la República Federal de Alemania (modelo estructurado inverso).

SYSTEM 2000 de M.R.I. de EE.UU. (modelo estructurado inverso).

TOTAL de Cincom de EE.UU. (modelo estructurado en red).

1978 ORACLE hace la primera implantación real del modelo relacional.

1986 El Dr. Date define la bases de datos distribuidas.

1993 ORACLE implanta las bases de datos distribuidas.

Instituciones implicadas en el desarrollo de las bases de datos

Codasyl o CODASYL (Conference On Data Systems Languages) es una organización internacional de usuarios de computadoras, fabricantes, casas de *software* y otros grupos interesados. Su objetivo principal es el diseño, desarrollo y definición de lenguajes de usuario comunes. Produjo el lenguaje Cobol. La implicación de su interés en la programación de las posibilidades del lenguaje Cobol.

El ANSI (American National Standards Institute, Instituto de Normalización Nacional Americano) es responsable de la estandarización de productos en Estados Unidos.

Su subcomisión X3 trata los sistemas de procesamiento de información, en los que la rama H2 es responsable de la estandarización de las bases de datos. La X# tiene también una Comisión de Planificación y Requisitos de Normalización (SPARC), que lleva adelante los planes. La SPARC ha producido la estructura ANSI/SPARC. La ANSI, al igual que la Institución de Normalización Británica (BSI), es miembro de la Organización Internacional de Normalización (ISO).

BCS, ACM, IEEE, IFIP y otras

Tanto la Sociedad de Computadoras Británica (BCS) como la Asociación para Equipos de Computadoras (ACM) han tenido una activa participación en el proyecto Codasyl. La BSC celebró una conferencia en octubre de 1971 para discutir el problema de los manejadores de bases de datos jerárquicos. Entre 1980 y 1983 la BCS sufragó los gastos para las conferencias nacionales como internacionales sobre investigación de bases de datos (junto con la Universidad de Aberdeen).

La ACM es conocida por sus actividades en todos los aspectos de la computación, incluyendo las bases de datos, tiene una publicación periódica, llamada ACM Transactions on Database Systems (TODS) .

El Instituto de Ingenieros Eléctricos y Electrónicos (IEEE) publica cierto número de revistas, algunas de las cuales son sobre bases de datos. También sufraga conferencias internacionales.

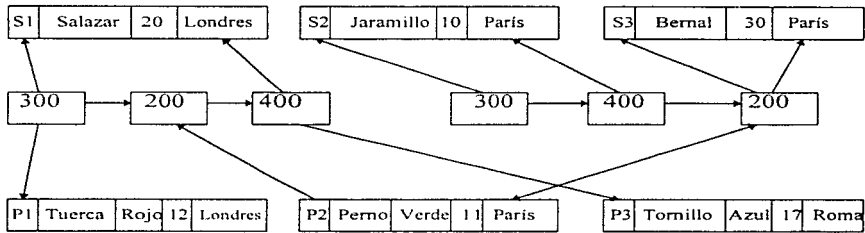
La ACM, CS y otras organizaciones son miembros del IFIP (Federación Internacional de Procesamiento de Información).

La contribución de los fabricantes de ordenadores y de las casas de software a la evolución de las bases de datos. En particular, IBM ha dirigido amplias investigaciones sobre bases de datos.

Modelo de red

Los datos en el modelo de red se representan por medio de “registros” y las relaciones entre los datos se representan con “ligas”, las cuales se pueden considerar apuntadores.

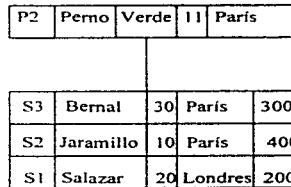
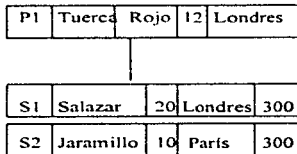
Los registros de la base de datos se organizan en forma de conjuntos de gráficas arbitrarias.



El modelo anterior nos muestra el problema de proveedores y partes, por cada proveedor existe un registro, lo mismo que por cada parte existe otro registro llamado conector, una ocurrencia en el conector representa una asociación entre un proveedor y una parte, contiene datos que describen esa asociación (la cantidad de la parte suministrada). Cada ocurrencia del conector se guarda en dos cadenas (una cadena de proveedores y una cadena de partes), la correspondencia entre el proveedor y los registros asociados del conector, es uno a muchos, que es una de las mayores características del modelo de red.

Modelo jerárquico

El modelo jerárquico es similar al modelo de red en cuanto a que los datos y las relaciones entre los datos se representan por medio de registros y ligas, respectivamente. El modelo jerárquico difiere del de red en que los registros están organizados como conjuntos de árboles en lugar de gráficas arbitrarias.



El mismo problema visto para el modelo de red, se representa a través de una estructura de árbol, en este caso, las partes van en un nivel superior al de los proveedores. Cada árbol se compone de una ocurrencia de registro de proveedor subordinada, uno para cada proveedor de la parte. Cada ocurrencia del proveedor incluye la cantidad correspondiente. El conjunto de ocurrencias puede contener cualquier número de miembros, incluso cero. El tipo de registro en el tope del árbol se conoce como "raíz", en general, la raíz puede tener cualquier número de dependientes de nivel inferior, y así sucesivamente hasta cualquier número de niveles.

Glosario

GLOSARIO

ACM Asociación para Equipos de Cómputo (Asociation Computer Machine) participante del proyecto de desarrollo del modelo de red de bases de datos.

ANSI Instituto de Normalización Nacional Americano (American National Standards Institute). Es responsable de la estandarización de productos en Estados Unidos.

Archie Sistema para localización de archivos que está disponible públicamente por FTP anónimo.

Applet Aplicación generada en Java (generalmete de animación).

Base de datos Colección de información a la cual se tiene acceso a través de un *software* especializado.

BCS Sociedad Británica de Computación (British Computer Society).

Broadcast Mensaje que manda una máquina hacia todas las demás máquinas de una red, haciendo una solicitud.

Cliente Aplicación de *software* que permite a un usuario obtener un servicio localizado en la red.

CGI Interfaz Común de Puerta de Enlace (Common Gateway Interface). Es la interfaz entre un servidor con Protocolo de Transferencia de Hipertexto (HTTP) y otros recursos. *Script* o programa ejecutable que permite la interacción del Web con otro *software* tal como bases de datos, etc.

CODASYL Conferencia de Lenguajes para Sistemas de Datos (Conference On Data Systems Language), nombre del proyecto de bases de datos basado en el modelo de red.

Datagrama Es un paquete de información que es enviado de una computadora a otra sin previo aviso. Los datagramas se usan normalmente en aplicaciones cuya cantidad de información a transmitir es pequeña y ocasional.

Dato Representación abstracta de algo.

DBMS Sistema Manejador de base de datos (DataBase Managment System) Conjunto de programas que constituye la interfaz entre los datos almacenados a bajo nivel en una base de datos y las aplicaciones, como las consultas de la información.

DNS Servidor de Nombres en Dominio o Servicio de Nombre de Dominio (Domain Name Service).
Servidor o servicio que permite hacer el mapeo entre nombres en dominio y direcciones IP.

Enrutador Máquina conectada a diferentes redes a un mismo tiempo y que sirve como puente de manera que las demás máquinas conectadas en las redes a las que puede acceder puedan comunicarse unas con otras.

FTP Protocolo de Transferencia de Archivos (File Transfer Protocol). Protocolo que define cómo transferir archivos de una computadora a otra.

GIF Formato de Intercambio de Gráficos (Graphics Interchange Format). Extensión asociada a archivos de imágenes.

Gopher Servicio basado en menú que sirve para explorar recursos de-Internet.

Hipertexto Un hipertexto es un conjunto de documentos cada uno de los cuales tienen ligas visibles a cada uno de los demás documentos que forman el conjunto.

Home Page Página dentro del Web con información referente a la persona organización o empresa que lo genera, escrita en HTML.

HTML Lenguaje de Marcado de Hipertexto (Hypertext Markup Language). Lenguaje para generar páginas dentro del Web basado en la teoría del hipertexto.

HTTP Protocolo de Transferencia de Hipertexto (Hypertext Transfer Protocol). Protocolo que utiliza el servidor de Web.

IEEE Instituto de Ingenieros Eléctricos y Electrónicos (Institute of Engineers Electric and Electronics).

IFIP Federación Internacional de Procesamiento de Información (International Federation of Information Process)

IMS Sistema Manejador de Información (Information Management System). Modelo de base de datos basado en el modelo jerárquico.

Información Conjunto de datos relacionados que reportan algo de nuestro interés.

Internet Con mayúsculas se refiere a la "red de redes" de cobertura mundial que están interconectadas entre sí usando el protocolo IP. Internet provee de servicios de transferencia de archivo, correo electrónico, sesiones remotas, etc.

ICMP Protocolo de Control de Mensajes en Internet (Internet Control Message Protocol). Es un protocolo que se considera parte del protocolo IP y que determina los errores que se generan al manejar el protocolo IP.

IP Potocolo Internet (Internet Protocol). El más importante de los protocolos sobre los cuales está basado Internet. Permite a un paquete viajar a través de múltiples redes hasta alcanzar su destino.

ISDN Red Digital de Servicios Integrados (Integrated Services Digital Network). Un servicio de telefonía digital. Con el servicio ISDN las líneas telefónicas a los hogares transportan señales digitales, en lugar de señales analógicas.

ISO Organización Internacional de Estándares (International Standar Organization). Organización que se dedica a fijar estándares sobre productos de la industria en general.

JAVA Lenguaje de programación orientado a objetos que permite una interacción entre el usuario y el Web a través de aplicaciones con animación llamadas applets.

JAVA SCRIPT Lenguaje de programación orientado a eventos y que se basa en guiones "scripts".

JPEG Grupo Unido de fotógrafos Expertos (Joint Photographic Experts Group), extensión común de archivos de imágenes surgida a partir de este grupo de personas.

LAN Red de Area Local (Local Area Network) red de computadoras de extensión territorial pequeña.

Línea Conmutada Comúnmente se refiere al tipo de conexión que se establece utilizando un emulador de terminal y un módem común. Comunicación asíncrona conmutada orientada a caracteres.

Línea dedicada Es una línea telefónica privada permanente que interconecta dos localidades. Las líneas en renta por lo general se usan para conectar redes de área local de tamaño moderado a un proveedor de servicios de Internet.

Mail Correo electrónico. Servicio de Internet de transferencia de mensajes.

MIME Extensiones de Multipropósito para correo Electrónico de Internet (Multipurpose Internet Mail Extensions). Estandarización de los formatos de archivos que se envían a través de Internet y que se originaron en el correo electrónico (mail).

NIC Centro de Información en Red (Network Information Center). Centro ubicado en Estados Unidos y que determina los dominios así como administra las direcciones IP.

OSI Interconexión de Sistemas Abiertos (Open Systems Interconnect). Estándar de interconexión de sistemas.

PERL Lenguaje de Extensión Práctica e Informes (Practical Extension and Reporting Language). Es un lenguaje desarrollado específicamente como herramienta de manejo de sistemas Unix.

PPP Protocolo de Punto a Punto (Point to Point Protocol). Es un protocolo que permite a una computadora usar los protocolos TCP/IP (Internet) con una línea telefónica estándar y un modem de alta velocidad (permitiendo una conexión a Internet con todos los servicios). El PPP es un nuevo estándar que sustituye a SLIP.

Protocolo Un protocolo es una definición de cómo se comportarán dos computadoras cuando se comuniquen entre sí. Las definiciones de protocolo van desde la colocación de los bits en el medio de transmisión hasta el formato de un mensaje de correo electrónico. Los protocolos estándar permiten que computadoras de diferentes fabricantes puedan comunicarse; las computadoras pueden usar *software* de distintos fabricantes y distintas presentaciones, siempre y cuando ambas estén de acuerdo con el significado de la información.

RDBMS Sistema Manejador de Bases de Datos Relacionales (Relational DataBase Managment System). Conjunto de programas que se encarga de la asignación de recursos con respecto a las bases de datos del tipo relacional.

Script Clase particular de programa que es interpretado en tiempo de ejecución, generalmente por el procesador de comandos.

Servidor *Software* que permite a una computadora ofrecer servicios a otra. Algunas computadoras establecen contacto con el servidor mediante el uso de un software "cliente" , también conocemos como servidor a la máquina que corre el software "servidor".

Sistema Conjunto de elementos organizados de tal manera que forman un todo orgánico.

Sistema basado en computación Conjunto de elementos organizados para llevar a cabo algún método, procedimiento o control mediante procesamiento de información.

SGML Estándar General del Lenguaje de Marcado SGML "Standar General Markup Language" que es un estándar internacional (ISO 8879) para procesamiento de texto e información.

SLIP Protocolo Internet sobre Línea Serial (Serial Line IP). Es un protocolo que permite a una computadora utilizar los protocolos de Internet (y contar con un domicilio válido de Internet) mediante una línea telefónica estándar y un modem de alta velocidad.

SSI Inclusiones del lado del servidor (Slides Server Includes). Técnica utilizada por algunos servidores Web para incluir los resultados de expresiones, archivos y programas externos dentro de un documento HTML que se envía al cliente (el visualizador).

TCP Protocolo de Control de Transmisión (Transmission Control Protocol). Es uno de los protocolos sobre los cuales se basa Internet. Para los técnicos, el TCP es un protocolo confiable orientado a conexión.

UUCP Unix to Unix copy. Es una herramienta para copiar archivos entre sistemas Unix.

UDP Protocolo de Datagramas de Usuario (User Datagram Protocol). Es otro de los protocolos sobre los cuales se basa Internet. Es un protocolo no orientado a conexión.

URL Localizador Uniforme de Recursos (Uniform Resource Locator). Es la forma común que utiliza el Web para especificar la ubicación de objetos con fines de vínculo, referencia y ejecución.

Verónica Es un servicio similar a Archie, que forma parte de Gopher. Verónica permite buscar en todos los servidores de Gopher (archivos, directorios y otros recursos).

WAIS Servicio de Información de Gran Cobertura (Wide Area Information Service). Herramienta que permite buscar información en bases de datos a través de Internet.

WAN Red de Gran Cobertura (Wide Area Network). Red de computadoras distribuidas en grandes extensiones territoriales.

WWW Telaraña a lo ancho del mundo (World Wide Web). Nombre con el que se identifica al servicio de información gráfica de mayor cobertura en la actualidad dentro de lo que es Internet. También conocido como Web.

Bibliografía

BIBLIOGRAFIA

Introducción a los Sistemas de Bases de Datos
Date, C.J.
Malpica, Jaime (traducción)
Ed. Adison-Wesley Iberoamericana
EUA 1986.

Database System Concepts
Korth, Henry F.
Abraham Silberschatz
Segunda Edición
Ed. McGraw-Hill
EUA 1991.

Database Analysis and Design
Hawryszkiewycz, J.T.
Segunda Edición
Ed. Memillan Publishing Company
EUA 1991.

Base de Datos
Gardarín G.
Segunda Edición
Ed. Paraninfo
Madrid 1990.

Fundamentos de Sistemas de Bases de Datos
Deen, S.M
Ed. Gustavo Gili
España 1985.

Conéctate al mundo de Internet
Krol, Ed
García, Hugo Edmundo, tr.
Ed. McGraw-Hill
Mexico 1995.

World Wide Web Unleashed
December, John
Ransall, Neil

Ed. SAMS
Indianapolis 1994.
Aprendiendo HTML para Web en una semana
Lemy, Laura
Magaña Pineda Luis Antonio(traducción)
Ed. Prentice- Hall Hispanoamericana
México 1994.

World Wide Web
Pfaffenberger, Byron
Ed. MIS
Ney York 1995.

Creación de Servidores de Bases de Datos para Internet con CGI
Rowe, Jeff
Ed. Prentice Hall Hispanoamericana
México 1996