

62

1992 Egen



UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO

FACULTAD DE INGENIERÍA

ANÁLISIS, DISEÑO Y PUESTA EN FUNCIONAMIENTO
DE LA MIGRACIÓN DEL ALMACENAMIENTO DE
HISTORIAS ACADÉMICAS DE LOS ALUMNOS
DE LA UNAM A CD-ROM Y SU
CONSULTA VÍA REDUNAM

T E S I S

QUE PARA OBTENER EL TÍTULO DE
INGENIERO EN COMPUTACIÓN

P R E S E N T A N :

GABRIEL IVÁN MONTES DE OCA MENDOZA

ANA CECILIA PADILLA CHÁVEZ

EDGAR VALENCIA FIGUEROA

Director de la Tesis: Ing. Juan Carlos Roa Beiza

México, D.F.

Septiembre, 1994

FALLA DE ORIGEN



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 está 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.

Hacemos patente nuestro agradecimiento a todas aquellas personas que directa o indirectamente han contribuido a la consecución de este trabajo y a nuestra formación profesional en general.

En particular agradecemos a las siguientes personas:

Sr. Alejandro Talavera Rosales.

Sr. Arturo Carrasco Ríos.

Ing. Celerino Domínguez Galicia.

Sr. Daniel Trejo Medina.

Sr. Diego Martín Zamboni.

Ing. Fernando Baz García.

Act. María Teresa Molina Charpenel.

Sr. Octavio López Chávez.

Act. Víctor Javier Arrieta Rosas.

Xerox de México, S.A. de C.V.

y en especial a nuestro director de tesis:

ING. JUAN CARLOS ROA BEIZA

por sus valiosas sugerencias y orientaciones.

A todos ustedes GRACIAS!!

Atte. Ceci, Iván y Edgar.

A mis padres María y Víctor:

*Por todas las cosas buenas y también las malas,
por todos los sacrificios realizados y el apoyo
brindado en todo momento,
por comprender los motivos por los que pasaba
menos tiempo con ustedes y el cual espero
recompensar con la culminación de mis estudios.*

A mis maestros:

*Por todas sus enseñanzas más allá
de la preparación académica,
por su apoyo y grandes consejos
de manera especial GRACIAS a
Cielo Davito, Consuelo Loto, Janet Harrison,
Enrique Arenas.*

A mis amigos:

*A Carlo por estos 12 años de sincera
amistad, ¡gracias por tu apoyo Poro !
A Diego y Daniel por su amistad y por
todo lo que aprendí de ustedes.
A todos mis compañeros de generación
por todo los agradables momentos que
pasamos juntos.*

A Edgar:

*Por todo el apoyo y cariño en
estos 5 años.
Por todos los momentos buenos
y también por los malos.*

A Juan:

*Por todo tu apoyo en el trabajo y
en la tesis y por encontrar a un gran
amigo con quien pelearme a gusto.*

A Fer:

*Por todo tu apoyo y paciencia,
por tu cariño.*

A la Dir. de Computo para la Admón. Académico:

*Por la oportunidad recibida de prepararme
profesionalmente*

"Cada solución genera nuevos problemas."

"Todo tiene necesidad de más tiempo del que usted piensa."

Conclusión de la ley de Murphy

Ana Cecilia Padilla Chávez

De manera particular expreso mi agradecimiento a:

- **Mis padres**, que siempre me alentaron a seguir adelante. Muchas gracias por sus consejos y cariño.
- **Mis hermanos, Laura, Yolanda, Héctor, Javier y Oscar** por su apoyo a lo largo de mi carrera.
- **Ana Cecilia** por todo su cariño a lo largo de 5 años, ¡ gracias Polla !
- **Ma. de los Angeles** por su comprensión y apoyo.
- **Gabriel Iván**, por su paciencia y buenas sugerencias a lo largo de la tesis.
- Los profesores de la Facultad de Ingeniería.
- La Dirección de Cómputo para la Administración Académica por todas las facilidades y a su personal.
- Todas mis amigas y a todos mis amigos.

*" Todo conocimiento comienza por los sentimientos".
Leonardo Da Vinci.*

*"Te diré lo que es el verdadero conocimiento:
cuando sabes saber que sabes;
cuando no sabes, saber que no sabes".
Confucio.*

*"Nunca he encontrado
una persona tan ignorante
que no pueda aprenderse algo de ella".
Galileo Galilei.*

Edgar Valencia Figueroa

En estas líneas quisiera patentar mi más grande agradecimiento a tantas personas que de una u otra manera han contribuido a mi formación profesional pero en especial a las siguientes:

A ti mamá: por todo tu cariño, comprensión, consejos, palabras de aliento, en una palabra por tu amor que me brindaste, para siempre seguir adelante en mi vida y no conformarme con eso. Gracias mil !!!!!

A ti papá: por tu cariño, tus consejos, tus sabias palabras y tu experiencia que me has transmitido desde siempre, y que han logrado hacer de mi lo que soy ahora y que nunca olvidare de aquí en adelante, para siempre alcanzar todo lo que desee. Muchas gracias !!!!!.

A ti Gus: por tu paciencia, tus consejos, tu tranquilidad, apoyo y cariño que siempre me has dado y que de mucho me han servido. Gracias !!!!

A ti Edgar: por tus consejos, tu fuerza, tu carácter, tu apoyo y cariño que siempre me has dado y que siempre me servirán. Gracias !!!!

A ti Horte: por que has sabido convertirte en parte de mi familia y desde siempre me has dado tu apoyo, consejos y cariño. Gracias!!!!

A Bety, Alejandro y a todos mis sobrinos por su cariño que siempre sabré recompensar.

A mis amigos y compañeros de tesis: Ceci y Edgar, y a Fer por su amistad, conocimientos y por tantos gratos momentos que hemos pasado juntos. Gracias.!!!

A todos mis amigos y amigas pero en especial a: Tere, Sammia, Grisel, Margarita, Ricardo y Carlos por su amistad a toda prueba. Gracias !!!

Al personal de la Dirección de Cómputo para la Administración Académica por las facilidades que nos dieron para la terminación de esta tesis.

Pero sobre todo a DIOS por la bendición de tenerlos a todos ellos.

La última cosa que uno sabe en la construcción
de un trabajo es que poner primero
Blaise Pascal

Rara vez ocurre lo que anticipamos;
suele ocurrir lo que menos esperamos
Benjamin Disraeli

No me digas lo duro que trabajas.
Dime cuanto logras realizar.
James J. Ling

Gabriel Iván Montes de Oca Mendoza

Prólogo

El avance vertiginoso en la tecnología computacional debe obligar a las instituciones a revisar continuamente sus implementaciones y adoptar los cambios necesarios, como con el propósito de hacerlas más eficientes. Eficiencia que, en este caso, se mide a través del logro de información expedita, confiable, segura y con los menores costos posibles.

La historia del desarrollo tecnológico a lo largo de la humanidad nos muestra que el surgimiento de nuevas tecnologías no conlleva necesariamente a la supresión de las anteriores, sino más bien a su complementariedad, lo que nos conduce a decir que no existe "la mejor tecnología", sino la más adecuada para cada caso en particular, según sus características.

Sin embargo, el llevar a cabo estudios para la adopción o no de nuevas tecnologías, implica una labor de investigación que no siempre es fácil de realizar. Es frecuente que el personal técnico mantenga su tiempo saturado en el cuidado y oportunidad de la producción; los tomadores de decisiones queden muchas veces paralizados ante el reto de seleccionar nuevas herramientas, sin arriesgarse a tomar decisión alguna por el temor de no ser oportunos en el momento de su implementación, ya que tanto el software como el hardware, gozan de innovaciones continuas, convirtiendo las mejores herramientas de hoy en deficientes el día de mañana.

A este tipo de retos se enfrenta continuamente la Dirección General de Servicios de Cómputo Académico, en donde se realizan estudios como estos, con la finalidad de "abrir brecha" y ganar experiencia que podamos transmitir a nuestras dependencias usuarias.

Diversas dependencias y campus de la Universidad Nacional Autónoma de México requieren para sus procesos rutinarios consultar información masiva, tanto de reciente generación como histórica; servicio que la Dirección General de Servicios de Cómputo Académico (DGSCA) a proporcionado de manera eficaz a través de microfichas. Sin embargo la caducidad del equipo empleado, los costos elevados de su reemplazo, el nivel de su utilización, y el desarrollo de nuevas tecnologías, obligaron a reconsiderar la metodología que debería de seguir imperando para ello.

El presente trabajo es el resultado de un estudio meticolosos llevado a cabo en la DGSCA, por tres jóvenes recién egresados de la Carrera de Ingeniería de Computación y de nuestro programa de Becas-DCAA, con el propósito de eficientar las labores académico-administrativas que se realizan en la Universidad Nacional Autónoma de México.

Ciudad Universitaria, septiembre 1994.

Act. María Teresa Molina Charpenel
Directora de Cómputo para la
Administración Académica, DGSCA, UNAM.

Prefacio

El principal objetivo de esta tesis es el presentar un trabajo desarrollado con el fin de modernizar el esquema de distribución de información a través de microfichas, utilizando herramientas de cómputo actuales, medios de almacenamiento de alto volumen como el CD-R, y herramientas de indexamiento, recuperación y visualización de información ampliamente utilizadas a nivel mundial, como lo son WAIS, GOPHER y MOSAIC. Asimismo, un trabajo como el presente muestra como se puede tener un esquema para el acceso de información general a través de la infraestructura de RedUNAM.

La tesis se divide en tres capítulos principales. El primero es una introducción que describe la importancia y el lugar que ocupan, dentro del esquema de información de la UNAM, los datos que hasta el momento se distribuían a través de microfichas. Asimismo se presenta la teoría básica detrás de las plataformas de cómputo que en dicho momento se perfilaban como las más actuales y cuya influencia impactaría en el diseño del proyecto.

Dentro del segundo capítulo, se presentan los recursos disponibles al momento de iniciar el proyecto, describiendo brevemente la infraestructura de cómputo y microfilmación con que se contaba, asimismo presenta una propuesta para la migración del esquema actual al nuevo.

El tercer capítulo es el cuerpo de la tesis, en el se especifica la metodología que se utilizó para adecuar la salida de información que se utilizaba para el esquema de microfilmación, principalmente en la serie de programas en lenguaje ALGOL dentro de la plataforma UNISYS, así como en los programas dentro del ambiente UNIX utilizados para convertir los archivos con que se alimentaba a la microfilmadora hacia formatos susceptibles de ser indexados. También se presenta la herramienta de indexamiento WAIS, utilizada ampliamente a nivel mundial para la organización y recuperación de información en grandes volúmenes. GOPHER y MOSAIC se presentan como las interfases para acceso al sistema desarrollado, proporcionando funcionalidad tanto en formato de texto a través del primero, como en la impresionante interfase gráfica que facilita MOSAIC.

Es mi deber resaltar, al ser testigo presencial, el gran esfuerzo de este grupo de trabajo, tanto en el desarrollo de los programas, como en la compilación de las herramientas de indexamiento e interfases.

El medio de almacenamiento que proporciona la tecnología de CD-R se presenta como la plataforma principal para resguardo de información, debido a su gran durabilidad, capacidad de almacenamiento, bajo costo y compatibilidad al poder ser leído desde cualquier manejador estándar ISO-9660.

Es difícil presentar tan solo con una descripción en papel (o en medios electrónicos) la alegría y el espíritu de satisfacción del grupo de trabajo que desarrolló este proyecto al realizar las primeras pruebas exitosas de su sistema piloto, satisfacción que los impulsa a imaginar y plantear nuevas aplicaciones de

los conocimientos y esquemas desarrollados durante la presente tesis; ésto último se presenta dentro de las perspectivas de desarrollo.

Al final de la tesis se encuentra un glosario de términos y un manual del usuario para el sistema piloto que se desarrolló, lo que hará más sencilla la labor de investigación para la persona que revise la presente tesis con el fin de apoyarse a su vez en su trabajo profesional. Asimismo se presenta una bibliografía en donde se podrá ahondar más en cada uno de los términos y conceptos presentados.

Es claro que un proyecto tan ambicioso como es la modernización del cómputo en nuestra máxima casa de estudios, apegándose a una serie de lineamientos tales como la eliminación en lo posible del uso de papel en nuestros trámites administrativos, y la adopción de plataformas más abiertas de cómputo con una utilización más adecuada de nuestros recursos de red, no es una tarea sencilla, y por lo tanto, este trabajo presenta el estado actual de uno de dichos proyectos, el cual sin embargo, seguirá desarrollándose y refinándose.

Una vez más, agradezco profundamente el esfuerzo y dedicación de este grupo de becarios, que al concluir este desarrollo no solo pudieron presentar su trabajo de tesis, sino que han proporcionado una plataforma de cómputo y conocimientos que servirán a la UNAM como punto de arranque para llevar a cabo más proyectos en el almacenamiento y recuperación eficiente de información académico-administrativa a través de su infraestructura de red. Agradezco también a la DGSCA, particularmente al Dr. Víctor Guerra Ortíz y a la Actuaría María Teresa Molina Charpenel quienes en todo momento apoyan la creación de nuevas ideas y desarrollos que generen el cambio y modernización

del cómputo en la UNAM, no solo desde el punto de vista de sistemas e infraestructura, sino también en la formación de recursos humanos, especialmente dentro del plan de becarios de la DCAA, del cual yo mismo formé parte.

Ciudad Universitaria, septiembre 1994.

*Ing. Fernando Baz García
Coordinador de Redes e Interoperabilidad,
Dirección de Cómputo para la
Administración Académica, DGSCA, UNAM.*

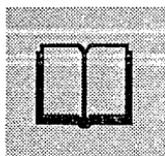


Tabla de Contenido

Prólogo	XI
Prefacio	XIII
Capítulo I. Introducción	1
1. Antecedentes.	3
1.1 Importancia de las historias académicas en la administración escolar.	
2. Teoría básica.	9
2.1 CD-ROM.	
2.2 Arquitectura Cliente/Servidor.	
2.3 La red INTERNET.	
Capítulo II. Planteamiento del problema	85
1. Recursos disponibles.	87
1.1 Descripción y características del equipo.	
1.1.1 Mainframes A12 de Unisys.	
1.1.2 Equipo de lectura óptica.	
1.1.2.1 Descripción general del equipo.	
1.1.3 Unidad de microfilmación.	
1.1.3.1 Descripción general del equipo.	
1.1.3.2 Formato de la información.	

1.1.4 Equipo SparcStation de SUN y el sistema operativo UNIX.

1.2 Descripción y características de la Red de Servicios Integrados de la UNAM.

2. Análisis del problema.

159

2.1 Planeación del sistema.

2.1.1 Condiciones actuales del sistema.

2.1.2 Entrevista.

2.2 Propuestas de solución.

2.2.1 Sistemas de almacenamiento masivo de información.

2.2.2 Sistemas de consulta de información.

2.3 Selección de la propuesta.

Capítulo III. Diseño de la propuesta

203

1. Migración de la información al ambiente UNIX.

205

1.1 Metodología.

1.1.1 El lenguaje de programación ALGOL en el equipo A12 de Unisys.

1.1.2 Especificación del programa en ALGOL.

1.1.2.1 Portabilidad del programa y ambiente de programación.

1.1.3 Transferencia de datos al ambiente UNIX.

1.1.4 Especificaciones del formato de la información para su consulta.

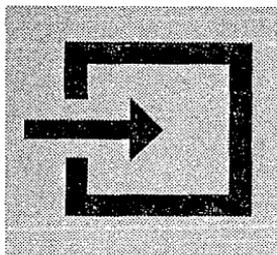
1.1.5 La herramienta AWK en el ambiente UNIX.

1.1.6 El programa en AWK.

1.2 Pruebas finales.

2. Implementación.	251
2.1 Descripción de los sistemas de recuperación y búsqueda.	
2.1.1 Aplicación WAIS en el indexamiento y recuperación de historias.	
2.1.2 Aplicación GOPHER en la presentación de historias y servicios.	
2.1.3 Aplicación MOSAIC como interfaz gráfica para el usuario.	
2.2 Proceso de compilación, depuración y adecuación de programas.	
2.3 Pruebas finales.	
3. Grabación del CD-ROM.	311
3.1 Características del hardware.	
3.2 Características del software.	
3.3 Criterios de grabación de información.	
3.4 Pre-master, pruebas y grabación.	
4. Pruebas del sistema final.	335
5. Consulta vía RedUNAM.	351
6. Perspectivas de desarrollo.	371
Conclusiones	375
Apéndice I	381
Make y Makefiles.	

Apéndice II	405
Manual del usuario.	
Glosario	425
Bibliografía	435



Capítulo I

Introducción.

Sinac, Sistema de Información Académica es el nombre bajo el cual se le conocerá a la aplicación resultante de la presente tesis: **Análisis, diseño y puesta en funcionamiento de la migración de la información de las historias académicas de los alumnos de la UNAM a CD-ROM y su consulta vía RedUNAM.**

A continuación daremos a conocer la importancia de la administración académica, así como una explicación de la teoría básica necesaria para comprender el presente trabajo.

1.

Antecedentes

En este primer tema daremos una reseña del desarrollo y la importancia que tiene la administración escolar en la Universidad Nacional Autónoma de México, así como una perspectiva de su desarrollo desde el nacimiento mismo de la Universidad hasta el momento actual y una pequeña exposición de como se lleva a cabo este proceso, además de los objetivos básicos que se tratan de cumplir con el presente trabajo.

1.1 Importancia de las historias académicas en la administración escolar de la UNAM.

La Universidad Nacional Autónoma de México, por su carácter de escuela de educación superior más grande e importante de México, es una de la instituciones con mayores retos en lo que corresponde a la calidad, confiabilidad, eficiencia y seguridad de sus procesos administrativos, tanto en lo concerniente a la administración central como a la administración escolar, procesos que son indispensables para llevar a cabo sus principales encomiendas que le ha asignado la sociedad mexicana, como la educación, la investigación y la difusión de la cultura de México y del mundo entero.

Dentro del amplio panorama que corresponde al universo de su administración, dividida en dos grandes áreas: la central y la escolar, la presente tesis se enfocará al área que corresponde a la administración escolar y más

específicamente, al proceso de los sistemas de historias académicas en cuanto al mejoramiento de la etapa de almacenamiento y consulta de información.

Para empezar remontémonos a la historia de la administración escolar de la UNAM. Esta historia podríamos decir que empieza en el mismo instante en que se concibe a la Real y Pontificia Universidad de México en el siglo XVI; claro que con una complejidad sumamente menor ya que no se puede comparar la población con que se contaba en ese entonces, de unas cuantas decenas de estudiantes, a cerca de 250,000 estudiantes hoy en día.

Pero acerquémonos en el tiempo al año de 1958 cuando siendo rector de la UNAM el Dr. Nabor Carrillo Flores se inician las actividades administrativas en cómputo al crearse la Sección de Máquinas de la Dirección de Servicios Escolares (actualmente la Dirección General de Administración Escolar, DGAE), donde su función era la de facilitar el trabajo de administración escolar generando las actas de exámenes y listas de los alumnos de la Universidad.

Como podemos observar desde hace 35 años la administración escolar está sumamente ligada al cómputo en la UNAM. Podemos decir que estas dos actividades, el cómputo y la administración escolar, están muy interrelacionadas y son dependientes entre sí; no podemos concebir los procesos administrativos actuales sin computadoras personales y mucho menos sin *mainframes* ó estaciones de trabajo, asimismo no podemos concebir la necesidad de tener instalada una plataforma de cómputo tan grande e importante sin estar dedicada una parte a la administración central y escolar.

Uno de los grandes problemas con los que se ha vivido desde hace mucho tiempo, desde antes de que llegara el cómputo a la UNAM, es el de tener un sistema de respaldo y almacenamiento de información confiable, sencillo y perdurable, que permita una consulta fácil y rápida por parte de los usuarios. A ésto le daremos solución utilizando una tecnología de vanguardia en México: **la grabación de CD-ROM.**

Pero para tener una mejor idea de lo anterior pasemos a dar una breve explicación del proceso que se lleva actualmente a cabo para la generación de uno de los documentos informativos más importantes de la administración escolar: **las historias académicas de los alumnos**; las cuales son un documento informativo sin valor oficial que expide la Universidad Nacional Autónoma de México, en el que se consignan las calificaciones que hasta la fecha han sido registradas en sus archivos.

Este proceso se lleva a cabo en el **mainframe** de la Serie A de la marca Unisys modelo A12¹, el cual data del año de 1987 que es cuando la **Dirección de Cómputo para la Administración Académica**, dependiente de la **Dirección General de Servicios de Cómputo Académico**, sustituye el equipo Burroughs B-6800 por el computador A12 y un A6 de Unisys.

Cabe destacar que los programas que generan las historias tienen por lo menos 20 años, en el transcurso de los cuales se les han hecho modificaciones, las cuales no son sustanciales, gracias a que los equipos de cómputo que se usaban anteriormente son totalmente compatibles con los actuales.

¹ En 1985 Burroughs y Univac se fusionan y crean la compañía Unisys la cual hereda a la familia B de Burroughs y la transforma en la familia Serie-A.

Pero pasemos a la descripción del proceso; primeramente la información más reciente se captura de las actas donde los profesores asientan las calificaciones de sus alumnos, para esto se utiliza un equipo de lectura óptica, el cual al tener apturados totalmente los datos se transmiten al equipo A12, en donde se le da un tratamiento de interpretación; una vez que ya están en un formato entendible se anexan a la información de cada uno de los alumnos para así actualizarla y poder después generar el archivo **backup**², el cual a su vez nos dará como resultado la impresión de las historias académicas en papel para ser repartidas a las escuelas y facultades y de ahí a los alumnos. Este archivo **backup** es grabado a cinta magnética para poder ser llevado a la unidad microfilmadora y así generar las microfichas que sirven -actualmente- como respaldo y almacenamiento de la información generada, de estas microfichas se hacen varias copias para satisfacer las necesidades de las dependencias usuarias de esta información.

Con lo anterior nos podremos dar cuenta de la importancia inherente que presenta la administración escolar en la UNAM, ya que gracias a ella es posible llevar un estricto control y conocimiento del rumbo y las acciones de todos sus alumnos, desde que solicitan su ingreso a la institución, ya sea en el nivel de bachillerato o en el de licenciatura, hasta que finalizan sus estudios dentro de la UNAM.

Gracias a la oportunidad que hemos tenido de pertenecer al programa de becas de la DGSCA-DCAA, nos hemos podido dar cuenta de muchas de las necesidades, carencias y malos procedimientos que sufre la UNAM en estos momentos, a la falta de presupuesto, a la burocracia que padece, etc.; pero

² Archivo del subsistema de control, manejo y operación de impresiones del sistema operativo MCP (Master Control Program) de la Serie-A de Unisys.

también nos hemos dado cuenta de los grandes aciertos, de la calidad de sus recursos humanos, de la idea permanente de estar a la vanguardia en tecnología y educación, etc. Es por eso que nos sentimos muy contentos que gracias a ésta situación podamos retribuirle a la Universidad todo lo que nos ha legado en conocimientos, aplicándolos en una solución a uno de sus grandes y añejos problemas: **el respaldo y almacenamiento de información**, utilizando tecnología de punta y aplicaciones de vanguardia en México.

Pero para poder pasar a lo que es el análisis, diseño y desarrollo de ésta situación es importante tener claros algunos conceptos básicos de las tecnologías utilizadas, como lo son CD-ROM, Arquitectura Cliente/Servidor, y la red más amplia a nivel mundial como lo es la Internet; así pues, pasemos a dar éstas bases teóricas.

2. Teoría Básica.

En este tema se presentan una serie de conceptos básicos que servirán para la mejor comprensión de este trabajo. Los conceptos a tratar son:

las características principales y el funcionamiento de un **CD-ROM**, características de la arquitectura **cliente/servidor** y una descripción de la red **Internet**.

2.1 CD-ROM.

Compact Disc Read-Only Memory (disco compacto de sólo lectura) mejor conocido por sus siglas **CD-ROM** es una adaptación directa de la tecnología del disco compacto de audio (CD) hacia el campo de la computación, específicamente hacia el almacenamiento masivo. Las múltiples ventajas que ofrece este dispositivo, provocaron una revolución en el campo de la distribución de información. Además, como dispositivo de almacenamiento, el CD-ROM también permitió el desarrollo de la **multimedia** ya que antes no existía una manera práctica de distribuir los grandes volúmenes de datos requeridos para el manejo de las bases de datos o de producciones que incluían gráficos, sonido y video.

Físicamente el CD-ROM es el mismo disco compacto que conocemos en la industria de la música, y debido a su utilización como dispositivo de almacenamiento, a su nombre se le agregó el término **Read-Only Memory**, el

cual es un término genérico para identificar un dispositivo de almacenamiento permanente e inmodificable.

Los CD-ROM son discos de 120 mm de diámetro fabricados en un material realizado en base a policarbonatos con una superficie brillante donde se almacena la información digital. Sólo uno de los lados del disco es utilizada para el almacenamiento. Un CD-ROM puede almacenar hasta 660 MB de información, lo que vendrían siendo 260 000 páginas de texto o 74 minutos de música y video.

El CD-ROM proporciona mayores beneficios comparados con otros dispositivos de almacenamiento; quizás los más significativos sean el costo y su portabilidad. Para aquellas organizaciones que distribuyen información como manuales técnicos, catálogos, directorios, etc., la versatilidad del CD-ROM como medio de distribución, por su facilidad de manejo y recuperación electrónica de información así como por su costo por unidad lo hacen más atractivo que otros dispositivos. Además es un medio de envío de información muy durable; a pesar de que no es indestructible, no está sujeto a los efectos de campos magnéticos. Otra ventaja es la convivencia que permite entre información tipo texto con audio, gráficos, imágenes y video, lo que lo hace un elemento indispensable en el campo de ***multimedia***.

PERSPECTIVA HISTÓRICA DE LOS DISPOSITIVOS DE ALMACENAMIENTO ÓPTICO

Los dispositivos de almacenamiento óptico son resultado de la mezcla de dos líneas de investigación. Por un lado tenemos, el desarrollo realizado por los

fabricantes de hardware de cómputo en busca de mejores dispositivos de almacenamiento; por el otro la industria de audio y video realizaba avances en métodos de grabación y reproducción ópticos para la nueva generación de sistemas de **videodiscos**.

A principios de los años 60's se empezaron a observar las limitaciones del almacenamiento magnético, por lo que se empezaron a realizar mayores esfuerzos en encontrar tecnologías alternativas. La introducción del rayo láser como medio de grabación de señales digitales en un medio sensitivo a la luz, hizo que las investigaciones se centraran en esta tecnología. Durante este período se generaron grandes expectativas sobre los dispositivos ópticos.

El objetivo final era obtener un dispositivo de rápido acceso y bajo costo que fuera un puente entre la costosa memoria principal y los lentos dispositivos magnéticos, como son los discos y cintas. Sin embargo, el avance en las investigaciones demostró que las limitaciones de este dispositivo estaban dadas por las propiedades del material y no por las propiedades de la luz en sí.

Como consecuencia de esto, se vinieron abajo muchas de las especulaciones realizadas; sin embargo, a estos dispositivos se les dió otro enfoque: los expertos vislumbraron que la principal área de aplicación no era como intermediario entre el almacenamiento interno y los sistemas magnéticos; sino como un dispositivo con gran capacidad de almacenamiento muy superior a las cintas.

Varios fabricantes comenzaron a experimentar con una gran cantidad de formatos y sistemas mecánicos de grabación/reproducción para el almacenamiento óptico; sin embargo, el desarrollo de los dispositivos de video y

audio fue el que dió la pauta para la posterior fabricación de los dispositivos de almacenamiento de datos y recuperación de información.

TIPOS DE DISPOSITIVOS ÓPTICOS

Los dispositivos ópticos de hoy en día son resultado del desarrollo paralelo generado por diferentes sectores del mercado. Esto nos lleva a que en la actualidad contamos con una amplia familia de dispositivos que han creado un gran impacto en el mercado de consumo del video y audio, los campos de educación, publicación y distribución de información.

Existen diferentes formas de clasificar todos estos dispositivos. La más simple de ellas, y en la que nos basaremos para describir a cada uno de ellos, es aquella basada en su capacidad de grabación. De esta manera tenemos tres tipos de dispositivos:

- Dispositivos de sólo lectura (read-only).
- Dispositivos de una sola escritura (write-once).
- Dispositivos borrables y grabables.

Dispositivos de sólo lectura.

Como su nombre lo indica, estos dispositivos no permiten al usuario final grabar información adicional, por lo que son ampliamente utilizados como dispositivos de distribución de información y reproducción de música y video.

Pertencen a esta familia los siguientes dispositivos que posteriormente explicaremos con más detalle:

- Videodiscos
- Discos Compactos (CD)
- CD - ROM
- CD - I
- CDV
- CD - R

Dispositivos de una sólo escritura.

A esta categoría pertenecen los discos denominados como WORM (Write-Once Read-Many) que son utilizados básicamente como dispositivos de almacenamiento en el campo de la computación. Con los discos WORM es posible para el usuario final grabar información posterior. Sin embargo, la nueva información se va anexando a la ya existente hasta que el disco se llena, es decir, no nos permite reutilizar una parte del disco que ya haya sido utilizada; de aquí su denominación de "una sola escritura, muchas lecturas".

Estas características lo hacen un dispositivo ideal para el almacenamiento de información sustituyendo con gran éxito a dispositivos como cintas y microfichas, ya que ofrecen la ventaja del acceso aleatorio de la información.

Dispositivos borrables y grabables.

Estos dispositivos ya se encuentran comercialmente disponibles, como aquellos presentados por 3M y Sony. Usan una gran variedad de técnicas de grabación entre las que sobresalen la magneto-óptica y de cambio de fase.

DESCRIPCIÓN DE LOS DISPOSITIVOS ÓPTICOS MÁS IMPORTANTES

Una vez que hemos descrito las características generales de los diferentes tipos, podemos hacer una descripción individual de cada uno de estos dispositivos.

Videodisco

Los videodiscos fueron originalmente desarrollados como una alternativa al videocassette para el mercado doméstico de reproducción de video. La producción de un videodisco involucra tres pasos principalmente:

- **Premastering**
- **Mastering**
- **Replicación**

El proceso de "**pre-master**" es el que se encarga de grabar el video, imágenes o información digital en un videotape master (principal) que es usado como entrada en el siguiente proceso. El proceso de "**master**" involucra la transferencia de información contenida en el videotape hacia el video disco. El proceso de "**replicación**" consiste en la creación de discos sub-master y en la producción de

grandes volúmenes a partir de ellos usando técnicas combinadas como son la presión, inyección en moldes y la duplicación óptica.

Los dos últimos procesos involucran una gran cantidad de inversión en sofisticado equipo y la necesidad de tener un cuarto 100% limpio.

CD

Los discos compactos aparecieron en el año de 1980 y comercialmente se les pudo adquirir en 1983. Este dispositivo tuvo un impacto en el mercado mucho mayor que el del videodisco por lo que se le considera el éxito comercial de la década.

El sistema de CD fue diseñado para competir en el mercado de reproducción de audio y se encuentra estandarizado por las técnicas implementadas por Phillips y la detección de errores diseñada por Sony y Phillips. Todas estas especificaciones se encuentran contenidas en un documento denominado **Red Book**; y precisamente fue la existencia de este estándar y de que todos los fabricantes de discos y unidades de reproducción se apegaran a éste, los que determinaron el éxito de este dispositivo.

El proceso de fabricación prácticamente es el mismo que el del videodisco, a excepción del formato de la información a grabar y que los discos maestros y de replicación son grabados de un sólo lado y tienen un diámetro de 12 cm. La principal diferencia entre el CD y el videodisco es que la información es almacenada en forma digital a diferencia de la forma analógica en que es tratada la información del videodisco.

CD - ROM

El desarrollo de la tecnología del CD ROM se debe principalmente a los avances del CD audio. La fabricación del disco en sí, así como los componentes ópticos, electrónicos y mecánicos de las unidades reproductoras de audio son muy semejantes a las del CD - ROM.

Las modificaciones básicas que se han realizado, son en primer lugar, el diseño de mejores técnicas de corrección de errores y la adaptación para proveer una interfaz que permita la conexión con equipos de cómputo.

Los parámetros de diseño para las unidades de CD-ROM son establecidas en el estándar denominado **Yellow Book**, desarrollado en 1983 por Philips y Sony. Los CD-ROM's y CD's son dispositivos de almacenamiento que utilizan el mismo procedimiento de grabación y replicación. Los discos tienen la misma dimensión y composición química, además de ser grabados y leídos en modo **CLV (Constant Linear Velocity)**.

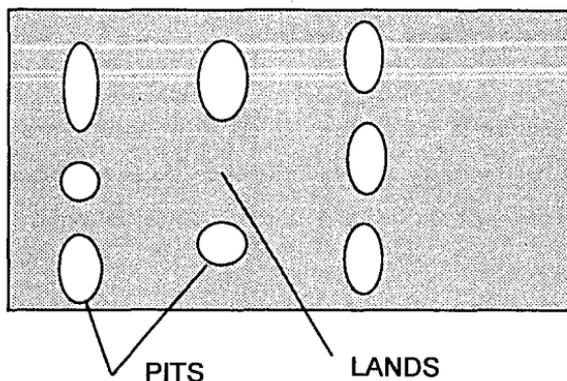
Las unidades reproductoras de CD y las unidades de CD-ROM tienen la misma velocidad de rotación además de utilizar el mismo sistema de modulación y detección/corrección de errores, definidos en los estándares **Red Book** y **Yellow Book** como **EFM** (Eight to Fourteen Modulation) y **CIRC** (Cross-Interleaved Reed Solomon Coding).

REALIZACION DEL CD-ROM MAESTRO

El primer paso en la fabricación del CD ROM es el proceso denominado **mastering**, el cual consiste en la transferencia de la información a un disco maestro a partir del cual se van a realizar todas las demás copias.

Este proceso consiste a grandes rasgos en la utilización de un disco de vidrio cubierto con material repelente a la luz, el cual es colocado en una máquina que lo hace girar. Los datos son codificados de acuerdo al formato que maneja el CD y esta señal es la que controla al modulador óptico. Este modulador es el que permite al rayo láser continuar su camino hacia el disco o desviarlo, todo esto de acuerdo a la codificación de los datos. Además de la rotación, la placa giratoria se mueve lentamente en relación al rayo o viceversa, para formar un track en espiral a lo largo del disco.

Cuando la transferencia de datos se ha realizado, el disco es tratado químicamente. Las áreas del disco expuestas al haz del láser son grabadas creando **pits** o depresiones en la capa del material repelente a la luz. Las áreas no expuestas no son afectadas y se denominan **lands**.



El proceso denominado *master* o realización del disco maestro, comienza con el pulido de una pieza de vidrio de 240 mm de diámetro y 6 mm de ancho. Este es cubierto con un protector, con material remitente a la luz y cocido en un horno. La preparación del disco maestro debe ser realizada en un ambiente extremadamente limpio con menos de 100 partículas de polvo por pie cúbico de aire (para darnos una idea de lo que significa esto, una sala de operación tiene cerca de 10000). Este ambiente libre de contaminación se mantiene a través del proceso completo de fabricación.

El material resistente a la luz sólo reacciona con un ancho de banda estrecho de luz, el cual es de la misma longitud de onda que el rayo láser azul utilizado para grabar. Para prevenir la exposición a los componentes de luz azul encontrados en la luz normal, el disco maestro de vidrio debe ser preparado bajo luces filtradas. Además de esto, se observa una estricta tolerancia en cuanto al ancho de la capa de este material, el cual es medido con un sistema láser.

La transferencia de datos del disco maestro es realizada por una máquina de fabricación de CD's *masters* que cuesta alrededor de un millón de dólares, la cuál es empleada para hacer CD's de audio y CD-ROM's. Un sistema típico consiste de controles y de un torno de cortado. El láser, el modulador óptico, el plato giratorio y todo el sistema óptico son montados en este último.

El plato giratorio rota a una velocidad de 500 rpm aproximadamente en el radio más interno de grabado (23 mm) y a 200 rpm en el radio más externo (59 mm) manteniendo una velocidad lineal de grabado constante de 1.4 m/s¹. Además del movimiento de rotación debe existir un movimiento relativo entre el rayo y el master de vidrio para crear el track espiral.

El corazón de la máquina es el láser utilizado en la grabación y modulación. El láser, emitido con una longitud de onda de 442 o 458 nanómetros (azul), es montado en la parte superior del torno. El láser es producido por argón o una combinación de helio y cadmio. El rayo pasa a través de espejos a un modulador óptico. Los datos ya codificador aparecen a la entrada del modulador. El modulador es de dos tipos:

Electro-óptico (E-O): éste opera con el efecto Pockel, el cual cambia la polarización del rayo de acuerdo a las señales eléctricas recibidas. Entonces pasa a través de un polarizador el cual permite sólo el paso de un rayo polarizado de sólo una forma.

¹La especificación de Philips de 1.2 m/s como mínimo y 1.4 m/s máximo. Variaciones dentro de un disco no debe exceder los .01 m/s.

Acústico-óptico (A-O): la señal codificada provoca que el modulador difracte el rayo hacia el vidrio exponiendo al material fotosensitivo a la luz (creación de un pit) o direcciona el rayo lejos del vidrio dejando un área no expuesta (land).

Se debe hacer notar que en este proceso aún no se forman directamente los pits, estos se forman con el proceso químico a que posteriormente se somete el disco maestro de vidrio.

Los controles que contienen la mayor parte de la electrónica son:

- Codificador EFM.
- Decodificador EFM.
- Terminal de computadora para el control del equipo.
- Sistema de pre-master.
- Unidad de cinta magnética.

Proceso químico

Es el siguiente paso y consiste de un revelado químico y metalización. El revelado es realizado mientras el master gira para asegurar una reacción uniforme. Durante este proceso los pits evolucionan lentamente hacia su forma final. Es vital detener este proceso cuando una adecuada geometría óptica del pit se ha alcanzado.

El master revelado es entonces secado, inspeccionado y preparado para una vaporización de metal, en la cual una delgada capa de plata o níquel es depositada por vaporización en el master. La metalización cumple dos funciones:

- hacer eléctricamente conductivo para la siguiente etapa del proceso y,
- poder ser reproducido en una unidad especial para monitorear la calidad de la señal e integridad de los datos.

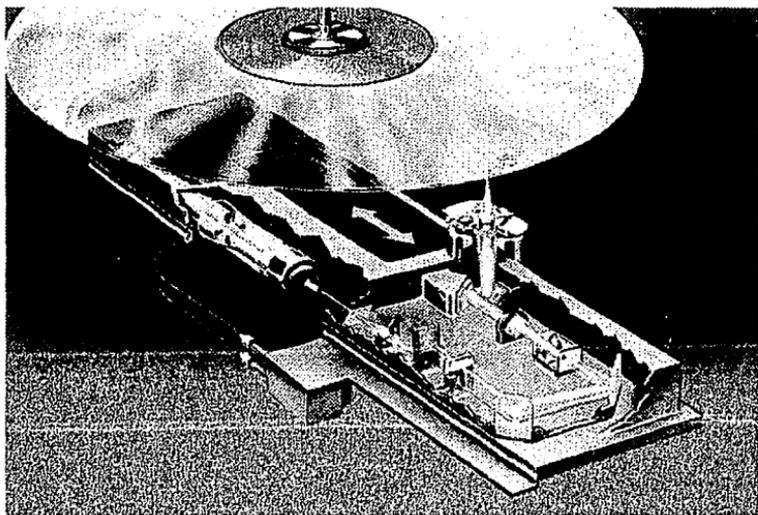
En la última etapa de fabricación se realizan replicas en plástico del disco maestro para su distribución y venta.

El master de vidrio es utilizado para formar impresores a través de un proceso llamado *electroforming*. Estos impresores forman parte de un molde para el proceso de creación de moldes por inyección. Los contornos de la superficie son impresos en un disco de plástico. Los datos en forma de pits y lands son transferidos de esta manera a grandes cantidades de discos.

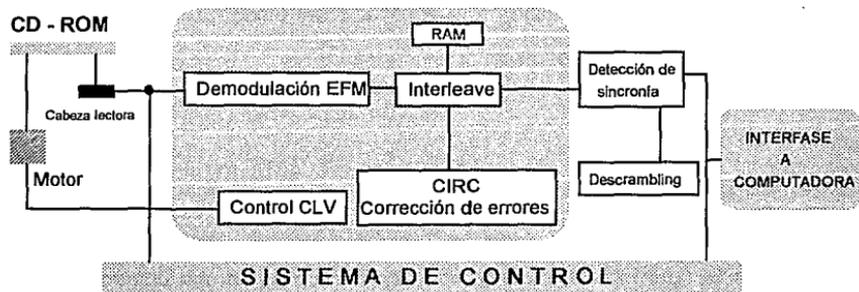
Los discos moldeados por inyección son cubiertos con una película delgada de aluminio (metalización) para reflejar el láser cuando sean leídos. La superficie es vulnerable a suciedad, rayones y condiciones ambientales por lo que es cubierta con una cubierta acrílica aplicada a la superficie del disco. El último paso es la impresión de la etiqueta en el disco.

HARDWARE DEL CD-ROM

La siguiente figura muestra los componentes internos de una unidad de CD - ROM, consideramos importante hacer una descripción de cada una de sus partes.



A continuación se muestra el diagrama esquemático de los principales componentes de hardware de la unidad de CD-ROM, los cuales se explicarán con más detalle más adelante:



- **Cabeza lectora**

Su función es la de traducir la información óptica del disco en señales eléctricas. La información es leída a través de un rayo láser que golpea la superficie del disco y detecta la intensidad del rayo reflejado. Esta intensidad varía, dependiendo si existe la presencia de una depresión (*pit*) o una zona plana (*land*). La profundidad de la depresión es la cuarta parte de la longitud de onda del láser. La luz reflejada por una depresión se encuentra defasada 180° con respecto a la luz que refleja las zonas planas o zonas que reflejan, lo cual causa una cancelación entre ellas. Al final tenemos que la luz reflejada por la depresión es mucho menor que la de las zonas planas circundantes. Estas emisiones de luz son leídas por fotodiodos que pueden detectar las diferencias de intensidad y convertirlas en señales eléctricas.

- **El rayo láser**

La palabra **láser** significa ***Light Amplification by Stimulated Emission of Radiation*** (amplificación de la luz por emisión estimulada de radiaciones). Se trata de rayos luminosos casi siempre comprendidos en la gama infrarroja, logrados por una aceleración de los átomos. En tales condiciones, la luz láser adquiere mayor intensidad, además de ser direccional (sus rayos viajan en dirección paralela), monocromática (es de un solo color) y coherente (tiene la capacidad de transmitir datos).

La luz es emitida por un semiconductor de unión **pn** realizado por una combinación de Al, Ga y As. La longitud de onda es de 790 nanómetros y con una potencia de 5 miliwatts.

La salida del semiconductor puede variar debido a la temperatura y al tiempo, debido a ello, para prevenir problemas, un fotodiodo es colocado cerca para monitorear la intensidad del láser. La salida del monitor es comparada contra un voltaje fijo de referencia. Si existe diferencia entre las entradas, la señal generada ajusta el rayo láser para obtener la salida deseada.

• Sistema óptico de lectura

El rayo pasa a través de una retícula difractinge² de microscopio que genera dos pequeños rayos, uno a cada lado del rayo principal. Esto produce el popular modelo denominado "tres rayos" también utilizado en los sistemas de audio.

Después de ser difractada, la luz pasa a través de un lente colimador³ que hace los rayos de luz paralelos para que puedan ser enfocados apropiadamente al disco por los objetivos.

A continuación pasa por un polarizador, cuya función es el permitir al rayo incidente pasar hacia el disco, pero redirecciona los rayos reflejados 90° hacia los fotodiodos para su interpretación. Esta lente esta diseñada para permitir que la luz polarizada horizontalmente pase a través de él sin cambios.

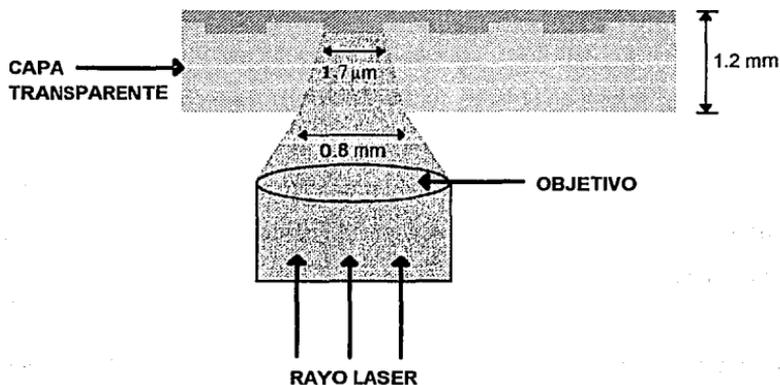
Pasa a continuación por una placa de una cuarta parte de la longitud de onda que da a la luz un cambio de fase de 90°.

El último paso es a través de los objetivos, que junto con el sustrato transparente del disco enfoca la luz para que sea lo suficientemente pequeña

²La difracción es la capacidad de las ondas para deflectarse o cambiar de dirección alrededor de obstáculos colocados en su trayectoria.

³Colimar es la obtención de un haz de rayos paralelos a partir de un foco luminoso.

para leer los pits y lands, pero no lo suficientemente pequeña para que sea afectada por la suciedad en la superficies del disco. El punto en la superficie externa del disco donde la luz entra es de cerca de 0.8 mm de diámetro. La capa transparente del disco es de 1.2 mm con un índice de refracción de $n = 1.5$. Esto significa que el punto de 8 mm de diámetro es reducido a un punto de $1.7 \mu\text{m}$ de diámetro en la capa reflejante.



Cualquier suciedad o marca en la superficie que se encuentre en la ruta del rayo será reducido por 6 veces en magnitud al tamaño de la capa reflejante del disco. Cualquier suciedad más pequeña que 0.5 mm no tendrá efecto al leer los pits y lands.

El reflejo del rayo de luz viajará en sentido inverso a través de los objetivos y de la placa con un cuarto de la longitud de onda pero al pasar por segunda vez por este último lente se le dará otro cambio de fase de 90° , el rayo ahora tiene una fase total de 180° con lo que se encuentra polarizada verticalmente.

Al alcanzar el rayo reflejado al polarizador, éste no permite el paso de rayos polarizados verticalmente así que lo refleja 90° hacia los fotodiodos a través de un lente convexo y uno cilíndrico. Estos dos lentes, junto con los objetivos son usados para mantener el enfocamiento adecuado del punto de luz en el disco. El rango de enfocamiento de la cabeza óptica es tan solo de $4 \mu\text{m}$.

- **Sistema de enfocamiento**

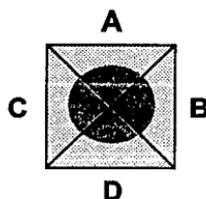
Utiliza el reflejo del láser principal para detectar y corregir errores de enfocamiento y mantener al láser dentro del rango de foco. Al pasar el rayo a través del lente convexo, el componente vertical es enfocado en el punto focal del lente mientras el componente horizontal no es afectado. En el lente cilíndrico, el horizontal es enfocado. Como resultado tenemos dos puntos focales diferentes uno para el horizontal y el otro para el vertical.

El fotodiodo colocado a la mitad entre éstos dos puntos tendrá una mancha circular en él. El sistema está diseñado para producir una perfecta mancha redonda en el fotodiodo cuando el rayo láser está perfectamente en foco. Cuando una variación en el peso del disco es encontrada, el rayo reflejado ya no está perfectamente alineado y se forma una mancha elíptica en el fotodiodo. Cuando el disco está más cerca del objetivo la mancha se extiende verticalmente y viceversa cuando el disco está más lejos.

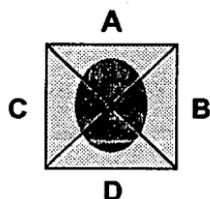
El fotodiodo realiza dos funciones con el rayo reflejado:

- La traducción de la información óptica a señales eléctricas.
- Mantener el sistema de lentes en foco.

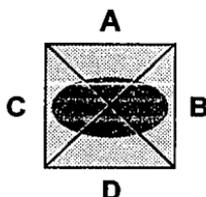
El enfocamiento lo realiza de la siguiente manera: el fotodiodo es separado en 4 cuadrantes los cuales pueden ser vistos como uno solo. Si un círculo perfecto cae en el fotodiodo es de pensarse que una cantidad igual de luz cae en cada cuadrante y su salida es la misma corriente. Si una elipse cae, la salida en corriente será diferente debido a la cantidad diferente de luz en cada cuadrante.



Diodo con 4 cuadrantes con la sombra proyectada por el láser cuando el disco está en foco.

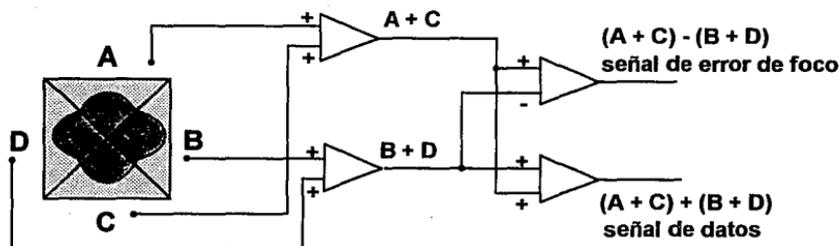


Diodo con 4 cuadrantes con la sombra proyectada por el láser cuando el disco y el lente están muy cerca.



Diodo con 4 cuadrantes con la sombra proyectada por el láser cuando el disco y el objetivo están muy alejados.

La suma de todos los cuadrantes es utilizada como la señal para convertir. Las señales de los cuadrantes A y C se suman, de manera similar los de B y D. Estas señales combinadas alimentan a un amplificador diferencial que genera la señal de error de enfocamiento (FER).



$$FER = (A + C) - (B + D)$$

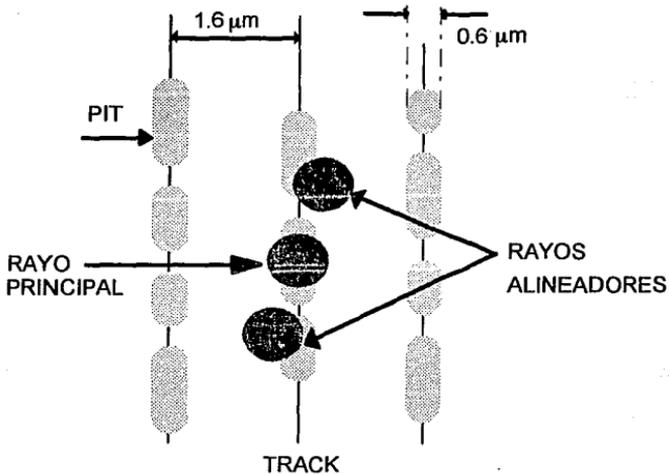
Esta señal de error controla un carrito electrónico conectado al lente objetivo para poder acercarlo o alejarlo del disco. Cuando el objetivo está demasiado cerca los cuadrantes A y C reciben más luz y el error será positivo con lo que el lente se aleja y lo contrario ocurre cuando está demasiado lejos. Este método de enfocado no es estándar en todos los equipos.

• Mecanismo de alineación

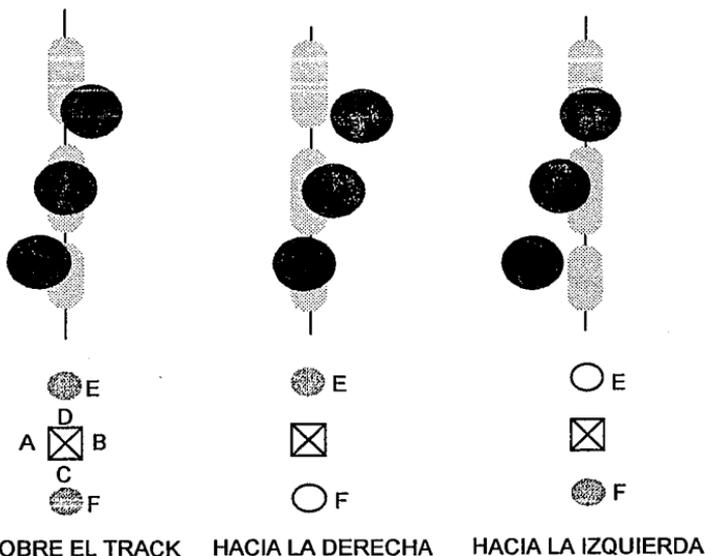
El otro problema que se presenta es la mala alineación radial. Los pits y lands del disco son organizados en pistas radiales y la distancia entre ellas es de $1.6 \mu\text{m}$ y el ancho de los pits es de $0.6 \mu\text{m}$. Considerando estas pequeñas distancias y excentricidades en el disco pueden causar una oscilación radial de $300 \mu\text{m}$. Existen dos diseños diferentes para corregir el alineamiento radial, un sistema usa el rayo principal y el otro utiliza dos rayos adicionales para mantener al principal en la dirección correcta.

Como se recordará desde el primer lente se obtienen dos rayos paralelos que siguen al principal a través del sistema completo y terminan a ambos lados del

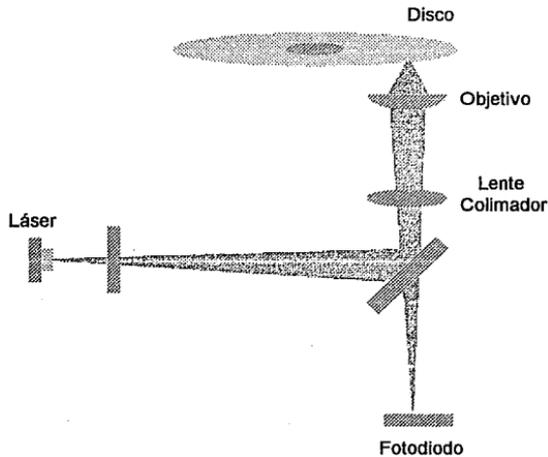
fotodiodo de cuatro cuadrantes sobre dos fotodiodos adicionales. La intensidad de estos dos rayos reflejados es utilizada para mantener la alineación correcta.



Como se muestra en la figura, uno de los rayos de a lado se posiciona para tocar parte del pit dentro del track y parte de la zona reflejante. El otro rayo es posicionado de manera similar pero hacia la derecha. Si el rayo principal está correctamente en el track, la intensidad del reflejo proveniente de los dos rayos será igual y su diferencia será cero. Si el rayo central no está centrado, un lado tocará más del área reflejante y el otro más del track, las salidas serán desiguales.



El control se lleva a cabo a través de un amplificador diferencial. Cuando el rayo está bien la señal de diferencia ($F - E$) es igual a cero. Dependiendo del signo de la diferencia manda las señales de control que mueve al lente radialmente para ajustar la linealidad.



- **Control CLV**

El control CLV es el servomecanismo que se encarga de controlar el motor que mueve al disco compacto con el fin de que la cabeza lectora accese la información. A diferencia de los discos magnéticos que rotan con una velocidad angular constante (**CAV Constant Angular Velocity**), el CD-ROM rota con una velocidad lineal constante (**CLV Constant Linear Velocity**). Esto significa que la velocidad del motor que gira el disco varía dependiendo de la posición de la cabeza lectora. Esta variación sirve para mantener una velocidad constante entre el disco y la cabeza no importando la posición en el disco. La relación entre velocidad lineal o velocidad del disco debajo de la cabeza (r), velocidad angular o velocidad del motor (w) y la distancia desde el centro del disco (r) la podemos expresar de la siguiente manera:

$$v = w r$$

La ventaja del sistema CLV es la cantidad de información que puede ser almacenada. En un sistema CAV los datos en el disco son más densos en el centro que en la orilla. En el sistema CLV la densidad de datos es la misma en cada punto del disco.

La desventaja principal de los sistemas CLV es su velocidad de acceso ya que, debido a los ajustes que debe realizar al variar la velocidad del motor, las búsquedas y saltos en diferentes localidades es más lenta.

Como lo indica el estándar del CD-ROM la velocidad lineal debe de ser de 1.3 metros/segundo. Para mantenerla, la velocidad de rotación debe de estar dentro de un rango de 500 rpm en el centro del disco, hasta 200 rpm en la orilla.

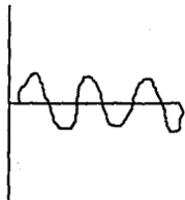
La lectura de la información se realiza a través del siguiente mecanismo: Los datos son almacenados en un buffer (memoria temporal) no importando a que velocidad sean leídos por la cabeza. Los datos son sacados de la memoria con una velocidad de 4.32 MB por segundo, lo cual es controlado por un cristal de cuarzo. El buffer, la vía de entrada y la de salida están diseñados para mantener el buffer a un 50% lleno. Si está por abajo de este porcentaje, una señal es enviada al controlador del motor para incrementar su velocidad hasta que el buffer esté al 50%. El proceso inverso se realiza cuando sobrepasa del 50%. El reloj asegura que los datos sean despachados a una velocidad controlada y que el sistema CLV se mantenga con una velocidad lineal constante de 1.3 m/s.

- **Modulación/Demodulación EFM**

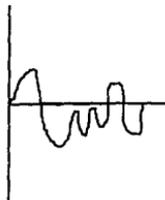
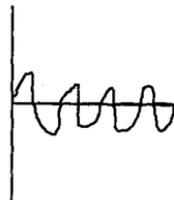
La suma de las salidas del diodo forman la señal modulada de los datos. La salida de los fotodiodos no es una señal cuadrada perfecta debido a que la

sombra del láser en el disco es mas larga que el ancho de la depresión, además las orillas de la depresión no pueden ser perfectamente verticales, lo que nos lleva a que la señal se parezca más a una senoide.

La señal EFM es amplificada y pasa a través de un filtro paso bajas. La suciedad en el disco reduce la amplitud de la señal reflejada, lo que ocasiona que las señales producidas por las depresiones y las de los lugares donde no hay, sean semejantes. El filtro paso bajas mantiene la señal cercana al umbral entre la depresión y la planicie.



DISCO LIMPIO

DISCO SUCIO
SIN FILTRODISCO SUCIO
CON FILTRO

La señal EFM se convierte a su forma digital con ayuda de un comparador y data strobe. El reloj es regenerado a partir de la señal con un sistema retroalimentado comparando contra la frecuencia del canal (4.3218 MHz). Una vez que ha sido regenerado, se generan otras señales que separan la información en datos e información de control.

La señal pasa entonces al demodulador EFM que convierte la señal a su forma original. La información del CD-ROM es organizada en bloques. Como parte del proceso de modulación se añaden bits adicionales para prevenir la presencia de

unos consecutivos y reducir las señales de baja frecuencia. Se desea que no estén presentes estas señales debido a que las unidades reproductoras trabajan con señales de baja frecuencia y alguna interferencia proveniente de la información puede causar problemas en el enfocamiento y en el control CLV.

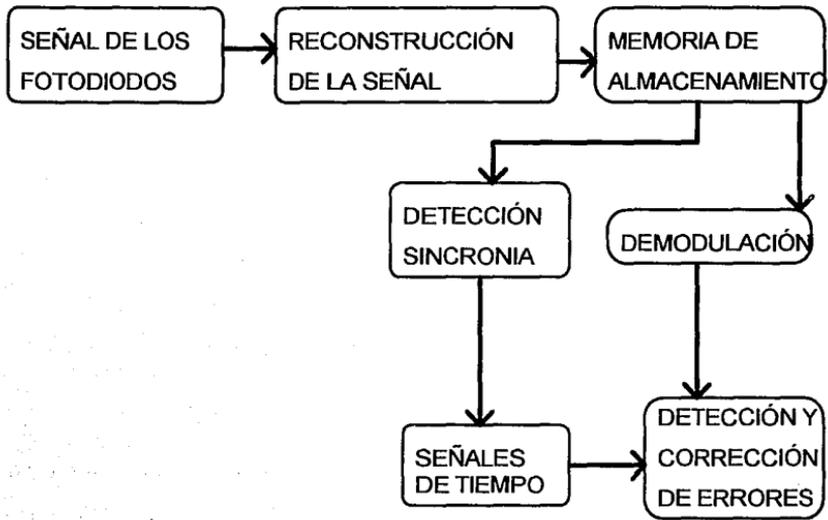
En los circuitos de procesamiento de datos, la información debe ser demodulada de 14 bits por byte a 8 bits por byte. La demodulación se hace a través de una tabla en ROM o utilizando un arreglo lógico. Después de la demodulación la información está lista para pasar al proceso de detección de errores y corrección.

Detección y corrección de errores

Los errores pueden ser clasificados en tres categorías:

- 1) errores aleatorios de un solo bit
- 2) múltiples errores aleatorios
- 3) errores en palabras o bloques completos.

DIAGRAMA DE BLOQUES DEMODULACIÓN



El procedimiento utilizado para la detección y corrección de errores es el estándar denominado **CIRC (Cross-Interleaved Reed-Solomon Coding)** llamado así honor a los creadores del método. El procedimiento consta de tres códigos de protección: **métodos de redundancia de información, múltiple paridad y el método denominado interleaving**. A continuación se explicará cada uno de ellos.

La redundancia de información consiste en generar tres copias de la información y hacer una comparación contra la original. Si existen diferencias, el valor correcto será el que gane la mayoría.

En el método de paridad múltiple, denominado código de bloque, se agrupan varias palabras en bloques para formar una matriz. Se añade una fila y una columna más que contienen la suma por renglón y columna respectivamente. De esta manera se puede identificar perfectamente que bit fue el que cambió. Un ejemplo de matriz de paridad múltiple es el que se muestra a continuación:

1	2	3	6
4	5	6	15
7	8	9	24
12	15	18	45

Tanto en los CD de audio como en los CD-ROM una fuente común de error es la suciedad en el disco y el tipo de error producido se presenta en bloques o palabras completas. Para prevenir esto, la información pasa a través de un proceso denominado *interleaving* antes de ser grabado en el disco. Este proceso lo que hace es reacomodar la información de tal manera que piezas consecutivas de información no sean colocadas en forma continua en el disco. Como consecuencia tenemos que cualquier mancha en el disco no afectará partes continuas de información.

Estos tres procedimientos son los que se llevan a cabo antes de grabar la información en disco. Al recuperar la información el proceso inverso al *interleaving* es el primero que se debe realizar. Los datos son almacenados en RAM y leídos en orden inverso al algoritmo de *interleaving*. Este procedimiento regresa la secuencia original de la información, así como el resto de la información de corrección de errores.

El código Reed-Solomon pertenece a la categoría de códigos por bloque que explicamos anteriormente. La manera en que hace de decodificación es la siguiente: los datos se encuentran organizados en bloques. Cada bloque de datos consiste en 12 bytes de sincronización (información que identifica al bloque), 4 bytes de dirección, 2048 bytes de datos y 288 bytes para la corrección y detección de errores. De estos 288 bytes 4 son usados para la detección de errores, 276 para la corrección y es resto no son utilizados. Dentro de cada bloque la información es organizada en matrices y se generan las palabras de paridad para cada renglón y columna.

12 BYTES SINCRONIA	4 BYTES BLOQUE DIRECCIONES	2048 BYTES DATOS	288 BYTES ECC		
			4 BYTES DETECCION	276 BYTES CORRECCION	8 SIN USO

- **Detección de sincronía y Descrambling**

La unidad de CD-ROM tiene que acceder aproximadamente 270,000 bloques diferentes, para lograrlo utiliza una dirección contenida en cada bloque que tiene el siguiente formato: un número que representa los minutos (0-60), los segundos (0-59) y un número de bloque (0-74) lo que nos forma 4 bytes de direccionamiento. A continuación se encuentran los 12 bytes de sincronía formados por "unos" a excepción del primer y último byte que deben ser detectados antes de efectuar el proceso inverso al *scrambling*.

El término *scrambling* significa mezclar, cuando hacemos referencia a este término en el almacenamiento de información en CD-ROM significa que vamos a mezclar bits a lo largo de la información con el objeto de igualar las longitudes

entre las depresiones y zonas planas. Al igualar las longitudes se decrementan las señales de bajas frecuencias. El proceso contrario llamado descrambling consiste en almacenar en RAM la información para eliminar dichos bits.

CD - I

Sus especificaciones fueron anunciadas por Philips y Sony en 1986. En su momento creó gran confusión, pues se le creía como el sucesor del CD-ROM. El estándar del CD-ROM, el libro amarillo, sienta las bases para el CD-I. No es un sucesor ni un subconjunto de CD-ROM, lo podemos definir como un uso especial del CD-ROM, es decir, dentro de la familia del CD-ROM, el disco compacto interactivo es la rama enfocada al área de consumo dirigida al entretenimiento y educación. En resumen el formato de grabación es idéntico al de un CD-ROM, solamente proporciona ciertas especificaciones para representar formas comunes de información para que puedan ser utilizadas para la aplicación especial.

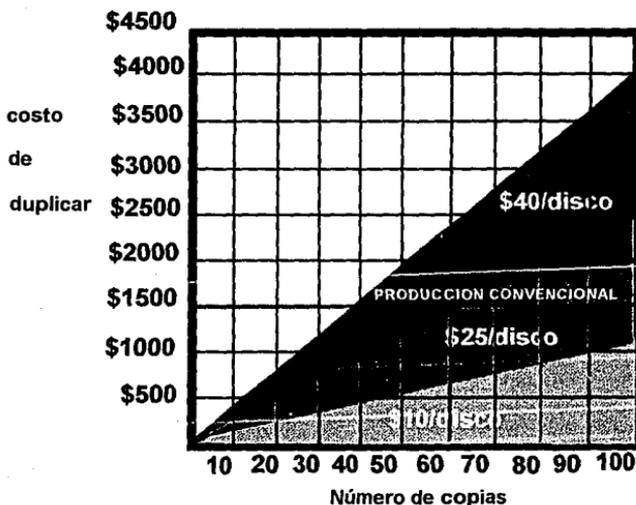
Además de la definición del medio, también se definió todo el equipo necesario para leerlo, el proceso de audio y video, el uso de microcontroladores especiales y del sistema operativo encargado de controlar todo esto. Este dispositivo está diseñado para contener en sí una computadora encargada de controlar todas sus funciones, sólo hay que conectarlo al televisor para funcionar.

CD - R

Compact Disc Recordable (Disco Compacto Grabable) es una nueva tecnología que le permite al usuario final realizar la grabación de su información a un bajo costo. Las aplicaciones principales del CD-R son:

- Realización de prototipos de productos nuevos.
- Realización de productos finales para distribución limitada.
- Almacenamiento de información.

Como se puede observar este producto está básicamente orientado a la producción de bajos volúmenes de discos compactos. Como podemos apreciar en la siguiente gráfica, la producción de CD's es eficiente y costeable si el número de copias es relativamente bajo. Estas gráficas representan los costos para precios de 40, 25 y 10 dólares por unidad.



Independiente del costo, existe otro factor importante en la determinación de usar este tipo de tecnología en lugar del método tradicional empleado en los CD-

ROM's: el tiempo de grabación. Cada disco debe ser grabado independientemente y este proceso le lleva un periodo de una hora en promedio.

El disco compacto para grabación consta de 4 capas: la capa superior es un material protector, la capa reflejante hecha de oro (Au), la capa de grabación y el sustrato. Cuando se realiza la grabación, el rayo láser quema las capas reflejante, de grabación y el sustrato. Como resultado de esto, la capa de grabación se funde y deja una hendidura en el sustrato, grabando de esta manera la información.



CAPA PROTECTORA
CAPA DE ORO (REFLEJANTE)
CAPA GRABACION

SUSTRATO

Esta tecnología ofrece ventajas y desventajas respecto a sus otros competidores: WORM y MO (Magneto-Optical). La principal desventaja del CD-R es que aún no ofrece la posibilidad de la grabación incremental. Otra de las desventajas es el costo de las unidades; mientras los unidades de grabación de CD-R cuestan alrededor de \$10,000, el precio de las unidades de grabación de WORM y MO están alrededor de los \$3,800 y \$4,500⁴. Sin embargo el principal atractivo del CD-R lo constituye su portabilidad, debido a los estándares que se han implementado en cuanto al formato de grabación, podemos estar seguros que la información que grabemos podrá ser leída en cualquier plataforma. Esto no sucede con el almacenamiento magnetoóptico como es el caso de los discos

⁴Precios de finales de 1993

Bernoulli y Syquest, los cuales son propietarios y en cuanto a los WORM, también el mercado está lleno de productos incompatibles entre sí.

Photo CD

Antes de explicar el funcionamiento de este dispositivo es importante explicar dos conceptos que se refieren a la forma en que se va a grabar la información: ***Single Session*** y ***Multisession***.

Cuando los datos son grabados en el CD-ROM, ya sea a través de la realización de un disco maestro o con la tecnología del CD-R, debe existir un encabezado denominado ***VTOC (Volume-Table-Of-Contents)***. Este es el que le indica a la computadora donde radican los datos en el disco, el nombre del disco y otro tipo de información. Este encabezado debe estar presente en el disco para que pueda ser leído y utilizado. De esta manera, cuando el encabezado es escrito, ***una sesión*** de grabación es concluida, por lo que todos los discos son, por lo menos, ***discos de una sola sesión (single session discs)***.

El sistema Photo CD está diseñado para escribir ***múltiples sesiones*** en un solo disco, es decir, lo que permite realizar es la grabación de un rollo fotográfico en el Photo CD, lo que constituiría la primera sesión. Tiempo después se puede grabar un nuevo rollo en el mismo disco junto con su VTOC lo que constituye la segunda sesión y así sucesivamente. Como se puede concluir las unidades lectoras de estos dispositivos deben ser lo suficientemente inteligentes para buscar a lo largo del disco completo en busca del último VTOC para poder acceder todas las sesiones grabadas en el disco.

La extensión lógica de esta capacidad se relaciona con el uso del CD-R para almacenamiento y otras aplicaciones de una sola escritura. El estándar denominado **Orange Book** el cual define las reglas para el CD-R, en parte, define la forma en que las múltiples sesiones son creadas y accesadas para todos los discos CD-R.

La capacidad de escribir múltiples sesiones en un disco de una sola escritura (CD-R) está dada por el software de grabación y el sistema de escritura. La capacidad para leer múltiples sesiones en un drive de CD-ROM esta dado por el **firmware** y el dispositivo en sí, además del **software** que el **drive** provee.

CD-ROM XA

El término XA significa **extended architecture** y fue desarrollado para ampliar la capacidad del CD-ROM de reproducir de manera sincronizada información mixta. Un ejemplo de esto lo constituye la sincronización que debe existir entre el audio y el video de ciertas aplicaciones sofisticadas de **multimedia** (la coordinación entre las palabras y el movimiento de los labios de una persona hablando).

Dentro del estándar del CD-ROM los datos de diferente tipo son presentados en bloques independientes, por ejemplo, una imagen será transmitida a la computadora como un solo archivo y el sonido como otro causando un retraso en el tiempo en el que la información de audio es entregada. XA fue desarrollado para resolver el problema al intercalar estos dos tipos de datos de manera que la presentación pueda ser debidamente sincronizada.

Usando esta tecnología se lee una parte de la información gráfica seguida de una parte de video, el procesador las identifica y las envía a su correspondiente buffer. La ilusión creada mediante este procedimiento es que se envían los dos tipos de información simultáneamente.

XA tiene además varios niveles/calidades de audio. El audio es almacenado utilizando el método ADPCM (*Adaptative Differential Pulse Code Modulation*). Este método sólo almacena las diferencias entre datos sucesivos de audio, lo que permite una codificación adecuada y que utiliza menos espacio. La unidad reproductora debe tener a su vez un decodificador ADPCM, el cual se coloca en la unidad lectora o en la tarjeta de interface de la unidad.

En resumen podemos decir que un sistema CD-XA debe cumplir con tres condiciones:

- Debe ser capaz de leer en el track dos tipos de datos, uno es el que entrega la cadena primaria de información (denominada **Modo 2 Forma 1**), y el otro que entrega la información dependiente del tiempo (**Modo 2 Forma 2**).
- Almacenar los dos tipos de cadenas, y entregar la información requerida al CPU y al manejador de video.
- Decodificar, descomprimir y enviar las señales de audio a las bocinas para su reproducción.

2.2 Arquitectura Cliente/Servidor

INTRODUCCIÓN

En este mundo competitivo es necesario para cualquier empresa tomar ventaja de cada oportunidad para reducir costos, aumentar la calidad y ofrecer un mejor servicio. Resulta indispensable contar con medios que permitan tener y generar a tiempo toda la información necesaria para tomar la mejor decisión. El uso de las computadoras en las diferentes ramas de la industria fue un hecho que incrementó de manera extraordinaria la eficiencia y rapidez en diversas actividades.

En los últimos años, la implementación y mejoramiento de los sistemas de comunicación entre los equipos de cómputo, como es el caso del desarrollo de las redes de computadoras, han permitido que se tenga disponible una gran cantidad de información de los más diversos temas y proveniente de cualquier lugar en un lapso corto de tiempo. Esta disponibilidad de información es una característica esencial en el proceso de toma de decisiones, por lo que se desarrollaron implementaciones para controlar y permitir el acceso a la información.

Uno de los modelos de cómputo que más revuelo han producido en estos últimos años es el denominado **modelo de cómputo cliente/servidor** el cual es aplicable a los más diversos campos como bases de datos, redes, sistemas y hardware.

El **cliente** es algo que requiriere que se lleve a cabo una tarea, y un **servidor** por el otro lado, es aquello encargado de realizar la tarea requerida. En términos de cómputo, un servidor es un programa que responde a las peticiones de otra máquina a través de lo que se conoce como **SRPI** (Server Requester Programming Interface). A diferencia del cliente, el servidor debe existir y debe ser buscado. Este primer concepto de cliente y servidor no difiere mucho del teleproceso, sin embargo como se desarrollará más adelante, las características de la arquitectura cliente/servidor se oponen a las del modelo de teleproceso o modelo de cómputo basado en mainframes.

DESARROLLO

Durante muchos años, el procesamiento de datos consistió en aplicaciones relativamente sencillas orientadas a transacciones, diseñadas para automatizar aplicaciones repetitivas en los negocios tales como manejo del libro de contabilidad, envío de estados de cuenta, etc. Programas en modo batch o en línea corrían en mainframes y manejaban este tipo de tareas fácilmente.

Este modelo de cómputo basado en la utilización de mainframes presenta las siguientes características:

- Procesamiento centralizado y compartido por todos los usuarios.
- Interacción con terminales. La terminal solo transporta las entradas hacia el host y despliega lo que envía como respuesta.
- Administración y mantenimiento complicado realizado por un pequeño grupo de expertos.

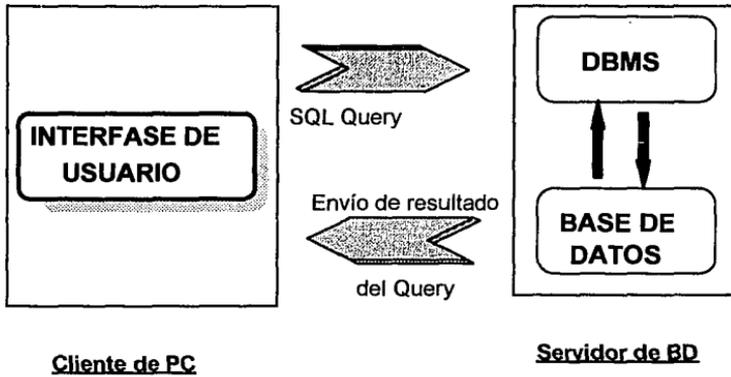
Es importante destacar que lo que denominamos procesamiento en mainframes no está necesariamente ligado a una plataforma de hardware en especial. Este puede ser realizado en cualquier computadora, lo que importa aquí es la manera en la cual se realizó el diseño de la aplicación.

Sin embargo a principios de 1980 la manera en que la gente empezó a usar computadoras cambio dramáticamente. Cada día, fue mayor el número de usuarios que, por la naturaleza de sus actividades, eran más bien considerados como el usuario final; estamos hablando de analistas financieros, de mercado, gerentes, vendedores, etc.

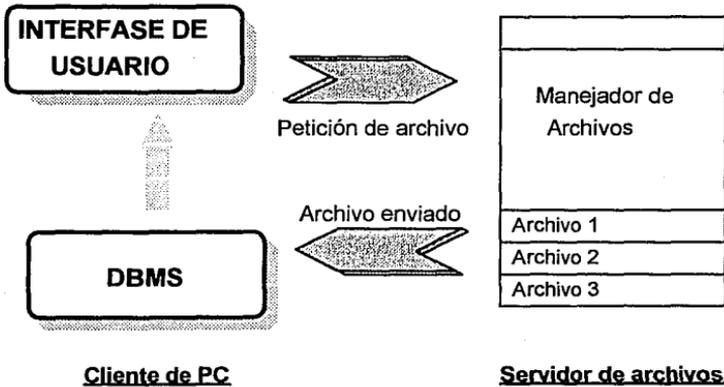
Para este nuevo tipo de usuarios se creó un nuevo mercado en el que se desarrollaron equipos como la PC y aplicaciones sumamente amigables y funcionales. Para aquellos que requerían trabajar con los sistemas en mainframes les resultaba difícil el trabajo en terminales, por lo que se demandó el tener mejores interfaces de trabajo. Además, el aumento de complejidad e interacción con el usuario de los sistemas transaccionales tradicionales también generaron la necesidad de un sistema que, además de mejorar el ambiente de trabajo, hiciera las transacciones de manera más eficiente y rápida.

La arquitectura cliente/servidor es un modelo de cómputo en red en el cual ciertas funciones requeridas por "**clientes**", generalmente estaciones de trabajo individuales dentro de una **LAN**, son atendidas por los procesadores más adecuados dentro de la red, a los cuales denominamos "**servidores**". El término servidor aparece en muchos lugares y bajo varios contextos; podemos hablar de:

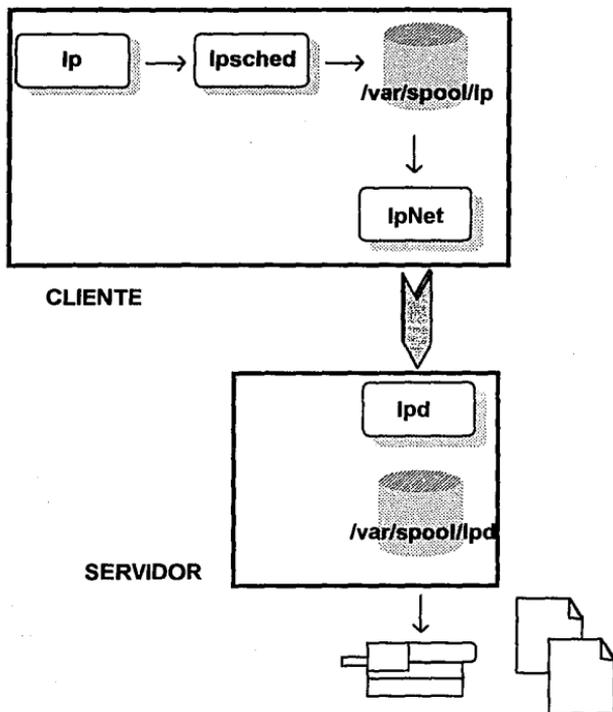
• Servidores de bases de datos



• Servidores de archivos:



- Servidores de impresión



Como habíamos mencionado, esta arquitectura no sólo se refiere al hecho de tener clientes y servidores, para poder llevar a cabo una implementación cliente/servidor son necesarias una serie de consideraciones las cuales se enumeran a continuación:

- Una **interfaz gráfica amigable** residente en el lado del cliente.

- **La división adecuada de las actividades de la aplicación entre el cliente y el servidor.**
- Que el sistema cliente/servidor tenga **capacidades de red**.
- Se debe distinguir perfectamente el cliente del servidor.
- Cliente y servidor pueden trabajar en plataformas separadas, pero podrían hacerlo en una sola plataforma.
- El cliente es quien inicia las acciones.
- El servidor de la base de datos es quien provee la protección de datos, seguridad, respaldo, recuperación y capacidad de SQL.
- Si el cliente o el servidor necesitan ser actualizados, no debe ser necesario actualizar ambos. Si la plataforma de hardware del cliente llega a ser muy poderosa, el desempeño del servidor puede convertirse en un cuello de botella.

Uno de los puntos fundamentales de esta arquitectura y el que lo hace diferente del teleproceso es la ejecución de ciertas partes de la aplicación en el cliente y otras en el servidor. Una buena implementación de ésta depende del tipo de actividades que se decida que se lleven a cabo en un determinado lado. Con el objeto de entender las limitantes de cómputo en el mainframe y servidores de archivos, así como las variedades de arquitecturas cliente/servidor vamos a ver las partes en que podemos dividir una aplicación:

- **Servicio de presentación.**
- **Lógica de presentación.**
- **Lógica de trabajo o aplicación.**
- **Lógica de datos.**
- **Servicio de datos.**
- **Servicio de archivos.**

Servicio de presentación

Es provisto por un dispositivo que acepta entradas del usuario y despliega aquello que la lógica de presentación le indique.

Lógica de presentación

Es la encargada de controlar actividades como la navegación entre formas, validación preliminar de datos, etc.; podemos decir que en sí controla la interacción entre el usuario y la computadora. Se encarga también del manejo de objetos en las ventanas y de los errores de presentación.

Lógica de trabajo o aplicación

Es un conjunto de decisiones, cálculos y operaciones que la aplicación debe llevar a cabo. Maneja cosas tales como la decisión de aceptar una orden, la evaluación de un préstamo, etc.

Lógica de datos

Es la expresión de las operaciones a ser realizadas en las bases de datos que son necesarias para llevar a cabo lo indicado por la lógica de aplicación. Por ejemplo para una base de datos relacional, estas expresiones son instrucciones SQL tales como "select", "update", "insert", etc.

Servicio de datos

Son las acciones que el DBMS o una aplicación cualquiera toma al llevar a cabo la lógica de datos incluyendo manipulación de datos, definición de datos, etc.

Servicio de archivos

Accesa el disco y recupera la información buscada por el DBMS o la aplicación. El servicio de archivos es llevado a cabo por funciones del S.O.

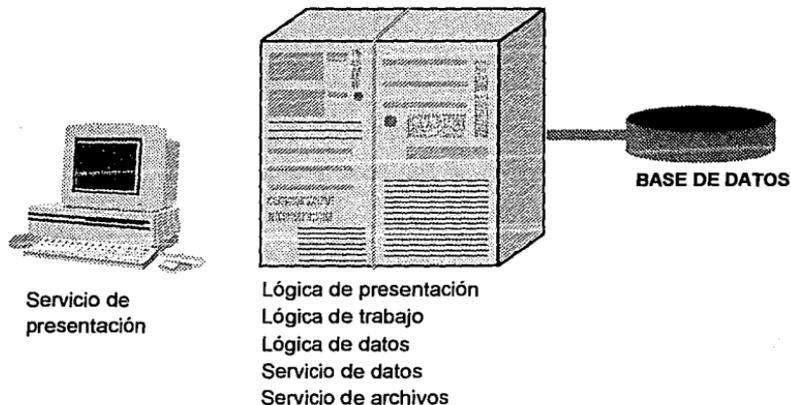
A continuación veamos una perspectiva de los diferentes modelos que han surgido con base en las divisiones que se han hecho de las tareas anteriormente descritas. Esta división se refiere a que tipo de actividades se van a efectuar en cada uno de los lados del modelo cliente/servidor.

En *ambientes que utilizan mainframes*, la terminal provee los servicios de presentación mientras que el host provee las funciones restantes. En este esquema existen dos serios problemas. El primero de ellos es que la interfaz de usuario esta limitada al estilo de interacción de una terminal. Aunque no es imposible, si es bastante difícil tener una interfaz gráfica eficiente debido a que todo el procesamiento reside en un host central y la gran carga que implicaría así como su transporte por la red son puntos a considerar.

En segundo lugar, por cada usuario adicional y por cada aplicación se genera una carga substancial al mainframe. Esto resulta en una pérdida de escalabilidad, lo que significa que el *host* está limitado en el número de usuarios y aplicaciones

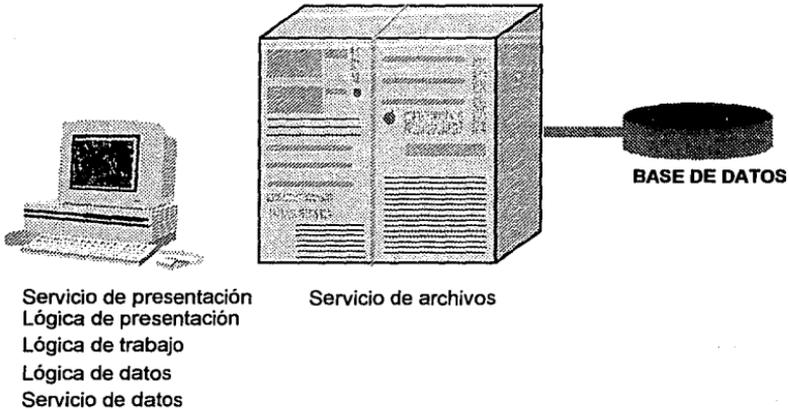
que puede manejar. Inclusive la carga de red es mayor porque no sólo envía datos sino también las instrucciones que permiten desplegar los datos en la pantalla.

ARQUITECTURA DE MAINFRAME



La **arquitectura de servidor de archivos** soluciona estos problemas pero introduce sus propias fallas. Al dividir los componentes de la aplicación, pasa todo a excepción del servicio de archivos a la PC. Los clientes manejan toda la presentación, aplicación y funciones de base de datos. El servidor de archivos sólo recupera los archivos necesarios y los pasa hacia el cliente.

SERVIDOR DE ARCHIVOS



Sin ningún retraso, ocasionado por el transporte a través de la red, entre la lógica y los servicios de presentación, es más fácil desarrollar un buen **GUI** para una aplicación. Y debido a que el servidor tan sólo recupera archivos el hecho de agregar usuarios adicionales y aplicaciones sólo ocasiona un pequeño incremento en la carga del CPU. Además como cada usuario tiene su propio procesador resulta en un mayor poder de procesamiento de la red en general.

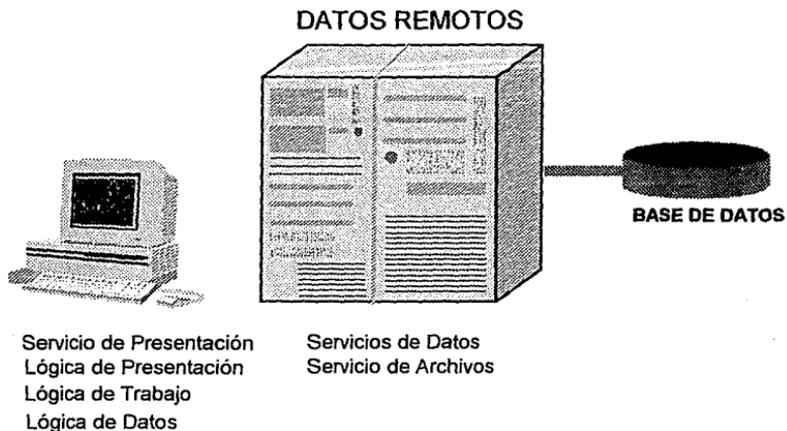
Sin embargo la arquitectura tiene dos problemas: la **demand** de **procesamiento** a la computadora local son muy altas y en segundo lugar es el gran **incremento de la carga de red**. Por ejemplo supongamos que tengo una base con las siguientes tablas: clientes, ordenes, parte y pedidos. Una petición razonable sería la lista de cada cliente con cada parte que el cliente ordenó y el valor total de cada parte ordenada. Esto requiere el movimiento del contenido de las cuatro tablas completas sin importar cuantos datos estén involucrados en realidad.

La **arquitectura cliente/servidor** está diseñada para encarar estos problemas al separar los componentes de una aplicación y colocarlos donde éstos sean más eficientes. Debido a que existen diferentes formas de realizar esta división es necesario entender las diferentes arquitecturas y para qué clase de aplicación es apropiada.

Two-tiered Client/Server o arquitectura de dos enlaces

La gran mayoría de configuraciones cliente/servidor usan un modelo two-tiered o de dos enlaces. Como mínimo una arquitectura cliente/servidor supone que ambos, los servicios de presentación y la lógica de presentación residen en el cliente. El resto de los servicios se colocan del lado del cliente o del servidor de acuerdo al diseño del modelo. La decisión depende del tipo de aplicación y sus requerimientos.

La primera posibilidad es colocar sólo los servicios de datos y de archivos en el servidor y el servicio de presentación la lógica de presentación y aplicación y la lógica de datos en el cliente. Este camino es llamado remote DBMS.



Como coloca el menor número de demandas en el servidor, el modelo de DBMS remoto provee la mejor escalabilidad. Sin embargo, para aplicaciones complejas que involucran gran interacción con la base de datos, puede llevarnos una gran carga en el procesamiento del cliente y en el ancho de banda de la red. El resultado completo de las instrucciones SQL deben regresar al cliente para su procesamiento porque la lógica de decisión se encuentra en el cliente. Este modelo complica la administración porque todo el código de la aplicación reside en cada cliente. Cuando se modifica el código esté se debe cambiar en todos los clientes.

Se puede reducir la carga en el cliente y en la red al mover parte de la lógica de trabajo al servidor (**el modelo de división de lógica**). Dando un paso adelante, se puede mover toda la parte de lógica de trabajo al servidor, dejando sólo los servicios de presentación y lógica (**modelo de presentación remota**).

El decidir entre estos tres modelos, el diseñador del sistema debe determinar la mejor localización para la lógica de trabajo. Generalmente no es una buena idea

permitir que las restricciones de hardware dominen la planeación. El ahorro en el desarrollo, operación y administración de un diseño apropiado son mayores que el costo de mudarse hacia servidores y clientes más poderosos.

Fallas del modelo Two-Tiered

Esta arquitectura presenta algunos problemas en aplicaciones complejas con muchos clientes, lógica intrincada, bases de datos heterogéneas y entradas heterogéneas en las bases. Los problemas que se presentan en esta arquitectura son:

- La carga de administración aumenta con un mayor número de usuarios.
- El uso de procedimientos almacenado en una lógica compleja.
- Problemas con la utilización de programas batch.
- Problemas con la integridad de la base bajo diferentes DBMS.

Arquitectura de Tres Enlaces (Three-Tiered Architecture)

ARQUITECTURA THREE-TIERED



Servicio de Presentación
Lógica de Presentación

Lógica de Trabajo
Lógica de Datos

Servicios de Datos
Servicio de Archivos

En esta arquitectura el cliente se dedica a la lógica de presentación y los servicios y tiene un **API (Application Program Interface)** para invocar la aplicación en un capa intermedia. El servidor es dedicado a los servicios de datos y de archivos (los cuales se pueden optimizar sin el riesgo de usar procedimientos almacenados). La capa intermedia es un servidor de aplicaciones en el cual la lógica de trabajo se ejecuta y desde la cual la lógica de datos invoca los servicios de datos.

El servidor de aplicaciones puede manejar las transacciones y asegurar la integridad de la base de datos distribuida al utilizar un proceso denominado **two phase commit**. El servidor de aplicaciones también puede manejar colas asíncronas para asegurar la confiabilidad de transacciones completas. También centraliza la lógica de la aplicación para facilitar la administración y proveer recursos por nombres y no por localización.

COMPONENTES DE LAS APLICACIONES CLIENTE/SERVIDOR--EL CLIENTE

El cliente es la estación de trabajo personal, es decir, una estación de trabajo como una Apple, IBM PS/2, NCD X-Terminal, SparcStation Sun, etc., pueden ser en algún momento clientes. No existe una tecnología específica para el cliente.

Durante los últimos 10 años, el desempeño de las estaciones ha crecido dramáticamente. Este crecimiento en poder permite que aplicaciones más sofisticadas puedan correr desde el equipo personal.

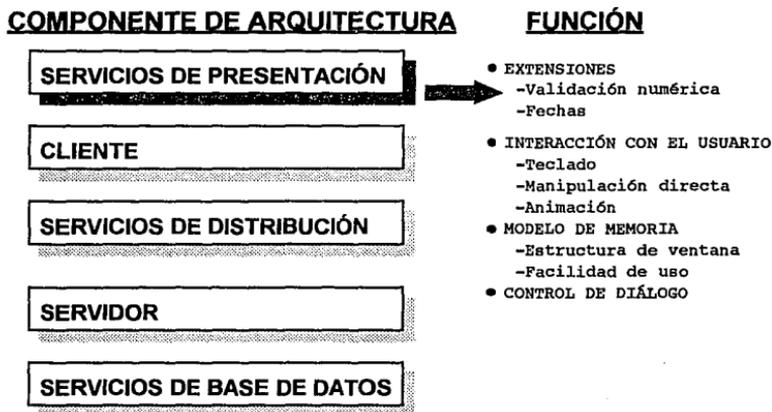
Las comunicaciones y la velocidad de las redes han crecido de igual manera en los últimos tiempos. Con el advenimiento de redes de áreas **LAN** y **WAN** de

alta velocidad y la fibra óptica, el desempeño y la confiabilidad han crecido substancialmente.

Las estaciones cliente pueden utilizar ambientes como DOS, Windows, Windows NT, OS/2, o UNIX. El cliente por lo general provee funciones para producir localmente aplicaciones, cuando está conectado en red puede acceder los servicios provistos por el sistema operativo de red. El cliente puede ser usado como terminal para acceder aplicaciones residentes en otro host.

IMPLEMENTACIÓN DE LA ARQUITECTURA CLIENTE/SERVIDOR

Un ambiente cliente/servidor tiene varios componentes. Los principales servicios son presentación, cliente, distribución, servidor y base de datos. Las tareas realizadas por cada uno se presentan a continuación:



COMPONENTE DE ARQUITECTURA**SERVICIOS DE PRESENTACIÓN****CLIENTE****SERVICIOS DE DISTRIBUCIÓN****SERVIDOR****SERVICIOS DE BASE DE DATOS****FUNCIÓN**

- CAPTURA DE DATOS Y
- VALIDACIÓN SIMPLE.
- ACCESO A TABLAS DE CÓDIGO.
- FRONT END DEL CLIENTE.
 - Recepción/Envío de eventos de la Interfaz de usuario.
 - Recepción/Envío de eventos asíncronos.
 - Manejador de errores.
- MANEJADOR DE MEMORIA
- COMPARTIDA.
- TRANSCRIPCIÓN DE SESIÓN.
 - Log de mensajes.
 - Debugger.
- SEGURIDAD.

COMPONENTE DE ARQUITECTURA**SERVICIOS DE PRESENTACIÓN****CLIENTE****SERVICIOS DE DISTRIBUCIÓN****SERVIDOR****SERVICIOS DE BASE DE DATOS****FUNCIÓN**

- COMUNICACIÓN PROGRAMA A PROGRAMA.
- MANEJO DE MENSAJES.
 - Síncrono/Asíncrono.
 - Ruteo transparente.
 - Concordancia de petición/respuesta.
 - Traducción de datos.
 - Manejo de timeout.
 - Garantía de entrega.
- INDEPENDENCIA DE PROTOCOLO.
- CONOCIMIENTO DE DONDE SE ENCUENTRAN LOS RECURSOS LÓGICOS Y COMO ACCESARLOS.

COMPONENTE DE ARQUITECTURA

SERVICIOS DE PRESENTACIÓN

CLIENTE

SERVICIOS DE DISTRIBUCIÓN

SERVIDOR

SERVICIOS DE BASE DE DATOS

FUNCIÓN

- FUNCIÓN DE TRABAJO REUSABLE.
- FUNCIÓN DE TRABAJO PORTABLE.
- ACCESO A TABLAS DE CÓDIGO.
- FRONT END DEL SERVIDOR.
 - Recepción/Envío de mensajes de servicio.
 - Respuesta a mensajes.
 - Recuperación.
 - Manejador de errores.
- SEGURIDAD.
- SEPARACIÓN DE LA BD (MÓDULOS DE I/O)

COMPONENTE DE ARQUITECTURA

SERVICIOS DE PRESENTACIÓN

CLIENTE

SERVICIOS DE DISTRIBUCIÓN

SERVIDOR

SERVICIOS DE BASE DE DATOS

FUNCIÓN

- IMPLEMENTACIÓN CLIENTE/SERVIDOR.
- TWO PHASE COMMIT.
- INTEGRIDAD DE DATOS Y RECUPERACIÓN.
- SEGURIDAD.
 - Recepción/Envío de mensajes de servicio.
 - Respuesta a mensajes.
 - Recuperación.
 - Manejador de errores.
- SEGURIDAD.
- SEPARACIÓN DE LA BD (MÓDULOS DE I/O)

2.3 La red Internet

Definitivamente, nuestra sociedad esta basada en la información. Ahora más que nunca el mover enormes cantidades de información salvando grandes distancias es una de las más importantes necesidades. Más empresas en los 90's así como personas relacionadas con el área de la computación se han dado cuenta de la gran necesidad de compartir datos e información, para sus empresas y para si mismos, **la información es poder**. Así entonces se abre un campo casi infinito: investigadores compartiendo sus datos y miles y miles de documentos de cualquier rincón de la Tierra, grupos de trabajo conduciendo conferencias "virtuales" sin salir de su lugar de trabajo, etc.

Tenemos la posibilidad de que con solo presionar unas teclas, podemos entablar una conversación en tiempo real con una persona en Japón, mandar un documento para que alguien en el extranjero pueda criticarla y aportar alguna idea nueva, indagar si una computadora está en sesión en Australia o simplemente ver si nuestro mejor amigo en Inglaterra está en sesión, todo esto en solo **!!! treinta segundos !!!** Además sin la necesidad de subir a algún avión o tener algún servicio de paquetería.

El primer problema que la gente afronta cuando interactúa con una red, es el darle a entender que todo es posible. Aún usuarios experimentados se sorprenden cuando descubren un nuevo servicio o un nuevo programa que hace sus vidas (y trabajo) más fácil y divertido. Todo esto es posible gracias a la red más grande que existe en el mundo: **la red Internet**.

Sin embargo, surge la pregunta, ¿qué es la Internet? Además esta pregunta es frecuentemente hecha cuando se tiene por primera vez contacto con ella. La **Internet** puede ser pensada como un conjunto de protocolos, un conjunto de circuitos, un conjunto de recursos compartidos e incluso como una actitud de comunicación e interconectividad. Algunas definiciones son las siguientes:

- Es una **red de redes**, basadas en el protocolo **TCP/IP**.
- Es una comunidad de personas que desarrollan este tipo de redes.
- Es una colección de recursos que pueden ser accedados por estas redes.

Actualmente la red Internet es un recurso global que conecta a millones de usuarios alrededor del mundo, esta comenzó como un proyecto del Departamento de Defensa de Estados Unidos hace ya 20 años.

Este proyecto surge como una necesidad de conectar una red experimental llamada **ARPANet** y otras redes pequeñas y enlaces vía radio. En sus comienzos fue experimental en el sentido de que se buscaba no perder la comunicación entre los diferentes centros militares aún y cuando se estuviera bajo un ataque. La comunicación siempre ocurre entre un par, es decir, entre un origen y un destino.

Debido al gran éxito que se tuvo con este tipo de red, varias Universidades de E.U.A. trataron de implementar las propias, tomando en cuenta lo hecho con ARPANet. Además por esos tiempos se lanzaba el sistema operativo **UNIX** el cual tenía integrado herramientas de comunicación usando el protocolo **IP**. Esto se verá más a fondo en el capítulo 3; Transferencia de datos al ambiente UNIX.

A continuación discutiremos algunos aspectos básicos de las redes.

- **Dominios.**

Uno de los aspectos más difíciles en el uso de las redes es decidir a donde quiere uno ir. Los nombres que uno escucha y los lugares a donde uno puede tener acceso pueden dejar a cualquiera muy confundido.

Si alguien de nosotros pregunta por alguna dirección de un amigo o familiar se espera que le den el nombre de la calle, número exterior e interior (si existe), colonia y código postal. Con esta información fácilmente podemos dar con él. De manera similar las computadoras cuentan con esta información, la forma general es la siguiente:

La dirección de correo electrónico de una persona:

`usuario@algunlado.dominio`

El nombre de una computadora:

`algunlado.dominio`

La parte de usuario es usualmente el nombre de la persona que tiene una clave en el sistema (computadora). El `algunlado.dominio` nos dice el nombre del sistema y donde se encuentra éste, además de una descripción del giro del mismo. Los dominios generalmente son los siguientes:

<code>com</code>	Organizaciones comerciales.
<code>edu</code>	Instituciones de educación.
<code>gov</code>	Agencias del gobierno de EUA.
<code>mil</code>	Agencias militares

org Organizaciones privadas.

Entre otras.

Además para cada país, con excepción de E.U.A., se tiene su dominio, por ejemplo:

mx México.
au Australia.
fr Francia.
ca Canadá.

La terminología apropiada para el nombre del dominio de algún *site* (lugar en donde se tiene una máquina conectada a red) se denomina FQND (*Full Qualified Domain Name*). Normalmente el nombre que se tiene es un indicador claro del giro de la organización. Por ejemplo el FQND del Instituto de Tecnología de Massachussetts es mit.edu, el de la Universidad Nacional Autónoma de México es unam.mx.

- **Direcciones.**

Cada máquina tiene una *dirección* la cual es *única* dentro de la Internet, llamada *Dirección IP*. A lo que se refiere es a un número de 32 bits que generalmente se representa con 4 números comúnmente separados por un punto "." como 132.248.27.10. La ARPANet solo tenía capacidad para 256 sistemas. En la década de los 80's se vio la necesidad de incrementar este número debido al rápido crecimiento de la red, así nació el esquema de 32 bits.

Cada pieza de la dirección IP (como 132, del ejemplo anterior) se denomina **octeto**, representando uno de los 4 grupos de 8 bits. Las primeras dos o incluso tres piezas (como 132 . 248) significan la red en la que nuestra computadora está, generalmente denominada **subred**.

Hasta ahora sabemos que las computadoras dentro de la red pueden ser referenciadas por su FQDN o su dirección IP. ¿Cómo esperamos que un usuario recuerde ambas? El usuario no tiene que hacerlo, para ello la Internet fue diseñada para emplear cualquier método. Además para el usuario es más sencillo reconocer palabras que números, por lo que el FQDN es mapeado a la dirección IP, ya que las computadoras manejan mejor números que palabras. Para tal efecto, cada dominio es *servido* por una computadora la que funge como un *servidor de nombres*, la cual provee la información necesaria para transformar una dirección FQND a IP y viceversa.

- **Las Redes.**

Internet La Internet es la "red de redes". No existe una red conocida como "La Internet", sino que es una colección de redes más pequeñas (SuraNet, RedUNAM, etc.) todas ellas interconectadas (*Internetworking*) para formar una gran red, comunicándose entre ellas mediante el protocolo TCP/IP. Todas las actividades se tienen en tiempo real.

UUCP La red UUCP es una asociación de sistemas comunicándose con el protocolo UUCP ("Unix-to-Unix Copy Program"). Se basa en la conexión de dos sistemas a intervalos específicos,

llamado **poleo**, y ejecutando cualquier trabajo que se encuentra programado.

Bitnet "Because It's Time Network" es un conjunto de sistemas con ligas *punto a punto* todas corriendo el protocolo NJE. Continúa creciendo, sin embargo debido a que los costos de conexión de la Internet han caído considerablemente, esta ha perdido fuerza.

Algunos de los servicios que se tienen en la red son los siguientes:

• **Correo Electrónico (EMail) .**

El deseo de comunicarse es la esencia de la red. El correo electrónico o EMail es una de las aplicaciones más usadas dentro de la red. Permite a los usuarios escribir y recibir mensajes no teniendo que preocuparse por cuando llegarán estos.

El EMail maneja el concepto de direcciones que anteriormente vimos. La dirección de EMail da toda la información (suficiente) para que un mensaje llegue a su destino en cualquier parte del mundo. Algo curioso es que no siempre estos mensajes van dirigidos a seres humanos; en algunas ocasiones van dirigidos a máquinas que proveen algún servicio, un conjunto de personas, etc.

El EMail maneja ciertos símbolos (%@!:) que tienen diferentes significados. Usando el formato de la Internet la dirección contiene el carácter @ (que se lee como "at", en) o si se usa UUCP contiene el carácter ! llamado "bang".

Así por ejemplo si se quiere contactar a John Lennon en el sistema `tzetzal.dcaa.unam.mx` su dirección EMail sería:

`john@tzetzal.dcaa.unam.mx`

Sin embargo si tiene una clave en un sistema UUCP llamado `condor` su dirección sería:

`condor!john`

Otro símbolo que se utiliza frecuentemente es el `%` que es utilizado como un método extra de ruteo de mensajes. Por ejemplo, si la máquina UUCP `aguila` está conectada a `sur.america.com` pero esta última no tiene un dominio dentro de Internet, un usuario `paco` puede ser "alcanzado" escribiendo la siguiente dirección:

`paco%aguila@sur.america.com`

Esta forma es significativa. La dirección nos dice que un sistema local enviará correo electrónico a `sur.america.com`. Una vez ahí, la dirección `paco%aguila` se transformará en `paco@aguila` la cual puede ser una dirección válida. Entonces se dice que `sur.america.com` recibirá y manejará el correo del host `aguila` y será enviado localmente al usuario `paco`.

Veamos un poco más acerca de la forma de enviar correo y de recibirlo. La implementación que haremos del programa `mail` es de una máquina corriendo una versión genérica de `UNIX`.

Una persona que quiere enviar correo lo haría de la siguiente forma:

```
tzetzal:1> mail cecilia@tzetzal.dcaa.unam.mx
```

```
Subject: Saludos !!
```

Como estas Cecilia, te mando este correo como una demostración de su uso.

La próxima vez que Cecilia verifique su correo tecleando simplemente mail verá lo siguiente:

```
tzetzal:1> mail
```

```
"/usr/spool/mail/cecilia": 1 messages 1 new 1 unread
```

```
?
```

lo cual da información a Cecilia de quién envió el correo, cuándo fue enviado y el objeto o tema del mensaje (*subject*). Ella probablemente respondería al mensaje tecleando el comando reply:

```
? reply
```

```
To: edgar@tzetzal.dcaa.unam.mx
```

```
Subject: Re: Saludos !!
```

Tu mensaje llego correctamente, te mando este como demostración del REPLY.

Ceci.

Como se puede observar un mensaje de mail tiene una estructura específica la cual es común en cualquier tipo de sistema, un ejemplo sería:

```
From jefe@tzetzal.dcaa.unam.mx Wen Jun 7 10:30:00 1994
```

```
Received: from tzetzal.dcaa.unam.mx by aramis.dcaa.unam.mx
```

with SMTP id AA211901

Date: Wen, 7 Jun 1994 10:29:09

From: El Jefe <jefe@tztetzal>

Message-Id: <9406072105.AA06631@tztetzal.dcaa.unam.mx>

To: edgar@tztetzal, cecilia@tztetzal

Subject: Ya los vi.

Ya se que saben utilizar el mail, ahora pónganse a trabajar!!!

Atte: EL JEFE.

La primera línea del From: y las siguientes de Received: nos dan la dirección real y quien nos envía el mensaje. El título de Date: nos da la fecha y hora en que el mensaje fue enviado. Después viene otro From: que nos da el nombre de la persona que envía el mensaje. El Message-ID: normalmente no es de interés para el usuario y solamente es un identificador único del mensaje. El título To: nos indica el o los destinatarios. Puede haber un Cc:, es decir, llevar una copia de nuestro mensaje a alguien. Finalmente un Subject: o tema del mail.

El orden exacto de esto puede variar de sistema en sistema pero siempre debe de contener esta información para asegurar que el correo llegue a su destino.

A continuación mostramos un ejemplo del correo electrónico en tztetzal (Servidor UNIX de la DCAA). La ventana que se muestra es exactamente la misma que se le da al usuario cuando invoca el comando mail para leer su correo. Dado que el servicio de correo es muy importante y que la interfaz que se da al usuario es muy pobre, más adelante se presentará un programa de lectura

de correo mucho más versátil implementado en la Universidad de Washinton, E.U.A.

```

xterm
Lutzjal 3:25 mail
From: root Mon Jul 11 07:47 CDT 1994
Date: Mon, 11 Jul 1994 15:35:10 +0200
From: "Casper H.S. Dijk" <casper@f11.uva.nl>
Subject: Re: Old patches still hanging around
To: Multiple recipients of List <OOORIS-OOORIS@ZIRI-BYMO.BITNI.Duop.ee.coged.edu>
Content-Length: 329

mark@ipotech.com (Mark Thomas) writes:

>According to the FAQ, it is usually not necessary to back out old patches
>before installing new versions. OK. I've done that thrice with 101318, since:

>101318-42 broke truss;
>101318-45 Fixed truss, but broke syslogd;
>101318-50 seems to work OK.

>My problem is that showrev -p shows ALL three versions, even after I
>performed an "rm -r /var/sockd/patch/101318-4*". Are the old patch revs
>forever embedded in the showrev output?

They're embedded in the pkginfo files installed for the instances
of the packages installed for the patches.

I've you pick up ftp.f11.uva.nl:/pub/solaris/ftp/install/*, there's a
Perl script hidden somewhere called /bin/patchdata that will remove all
traces of all obsoleted and superseded patches. It greatly cleans
up the showrev output.

Casper
?
```

Como podemos observar, el prompt del mail (?) está esperando de algún comando para realizar alguna acción sobre el correo. El programa para manejar el correo llamado **PINE** ofrece al usuario un ambiente más amigable para la lectura del correo. Dicho programa se muestra a continuación:

```

xterm
-----
PINE 3.87  MAIN MENU                               Folder: INBOX 94 Messages

?  HELP                Get help using Pine
C  COMPOSE MESSAGE    Compose and send a message
F  FOLDER INBOX       View message(s) in current folder
L  FOLDER LIST       - Select a folder to view
G  ADDRESS BOOK       Update address book
O  ABOUT              Configure or update Pine
Q  QUIT              Exit the Pine program

Copyright 1989-1993. PINE is a trademark of the University of Washington.
[Folder "INBOX" opened with 94 messages]
? Help                PrevtMsd                ReINbox
C OTHER COMP. I First Fldrs I  NextMsd                RBlock

```

Este programa a través de menús nos permite navegar fácilmente por todos los mensajes de correo que se tienen almacenados, y que, como veremos en las listas de correo a continuación, facilita la lectura del mismo.

• Listas de Correo.

En algunas ocasiones la gente está interesada en compartir experiencias laborales o discutir algún tema en cuanto se da la oportunidad. Una forma moderna de intercambio de información y experiencias es mediante el uso de una *lista de correo*, usualmente una dirección *EMail* que redistribuye todo el correo que le llega a una lista de direcciones que tiene. Entonces un usuario interesado en los temas que se discuten en la lista puede *suscribirse* a ella sin ningún costo. Para poder suscribirse a estas listas se debe mandar una petición a la cuenta que administra dicha lista, es decir, la dirección de la lista puede ser `admin-sun@eecs.nwu.edu`, en donde se pueden enviar los mensajes a los demás

usuarios para poder discutir y presentar puntos de vista, sin embargo para mandar peticiones de inscripción, desinscribirse, etc., se hará a otra cuenta generalmente con la extensión `-request`, así `admin-sun-request@eecs.nwu.edu` recibirá todas las peticiones del usuario referentes a actividades administrativas.

En la red **BITNET** se tiene un programa (sistema) que hace la labor administrativa automáticamente llamado *listserv*. Así, en vez de que un administrador humano tenga la labor administrativa, esta se automatiza mediante el uso de comandos, los cuales sirven para inscribirse, desinscribirse, pedir información de las listas, tópicos de interés, etc.

Por ejemplo si se quiere obtener toda una lista de listas disponibles se puede mandar un *mail* a `LISTSERV@BITNIC.BITNET` con el siguiente comando dentro de él:

```
list global
```

El comando fundamental es el `subscribe` el cual nos da acceso a la lista de nuestro interés. Dentro del cuerpo del correo electrónico se debe de incluir la siguiente línea:

```
subscribe nombre_de_la_lista nuestro_nombre_completo
```

El programa al poco tiempo responderá con un mensaje indicándonos que ya somos miembros de la lista y recibiremos el correo de ella.

El comando obvio para desuscribirse es `unsubscribe`, el cual nos borrará de la lista. Si queremos obtener toda una ayuda de los comandos de `listserv` tenemos que enviar un *mail* que contenga el comando `help`.

Entonces una vez que estamos inscritos a una lista de correo, los mensajes nos llegan por EMail a nuestra cuenta, utilizando el programa **PINE** veríamos algo similar a esto:

PINE 3.87		FOLDER INDEX		Folder: INBOX Message 20 of 94	
	20	Jun 24	A. Mesman	(5,998)	CALL FOR PAPERS: "UNIX and the Law"
	21	Jun 24	Refugio Vallejo G.	(1,889)	Re: your mail
	22	Jun 25	Pete Hartman	(1,969)	Re: Need Message compiled for Solaris (Solaris
	23	Jun 28	Bruce Barton	(2,624)	Compiling freemTS for Solaris (Solaris
A	24	Jun 28	Denise Blackley	(1,977)	HELP: topher "X" command crashes file
	25	Jul 7	Diego Martin Zumbi	(2,133)	Introduction of HELP Reporter
N	26	Jul 8	Mark Mozobelli	(2,494)	Solaris 2.3 and Preprocessor 2.3.1: do
N	27	Jul 8	Mark Mozobelli	(2,479)	Tom Xteras
N	28	Jul 8	Benjamin Ruiz	(813)	
N	29	Jul 9	Richard Mathews	(1,744)	Re: Bill: Solaris2.1.386 +1.0 driver bo
N	30	Jul 9	Nobél Nakagawa	(1,140)	2.3 Uniprocessor vs Multiprocessor
N	31	Jul 8	Huy T Vu	(1,457)	Can't run man on SUN Solaris 2.3
N	32	Jul 8	Stainless Steel Ro	(1,664)	Re: Can't run man on SUN Solaris 2.3
N	33	Jul 9	Francis He	(1,375)	Preventing interrupts from each accept
N	34	Jul 8	Mark Williams	(1,461)	Reading sendmail troffed instruction
N	35	Jul 8	Huang Saha	(2,783)	display oddity ...
N	36	Jul 9	Ravens Pathana (ark)	(1,331)	Re: DR's reverse mapping
N	37	Jul 9	Casper H.S. Dijk	(1,886)	Re: 2.3 Uniprocessor vs Multiprocessor
N	38	Jul 8	Tom Limoncelli	(3,777)	Re: Inn working, protocol error? Help

Help Main Menu PrevMsg PrevPage Delete Reply
 OTHER CMDS ViewMsg1 NextMsg NextPage Undelete Forward

En nuestra pantalla veremos toda una lista de encabezados de mensajes (*headers*) los cuales tocan un tema en particular, en ocasiones las personas que los escriben solicitan ayuda en algún tópico muy específico, otros para hacer preguntas generales, sin embargo el objetivo se cumple: poner a disposición de la comunidad de la Internet un medio de comunicación fácil y rápido. Además de esto se encuentran también en la lista las respuestas a algunas de las preguntas que la misma comunidad hace, con lo que se tiene una constante retroalimentación de información y consejos de un valor invaluable; ya que a este nivel, nos puede contestar desde un alumno de alguna universidad hasta el más distinguido Doctor en el área.

- **El servicio de ARCHIE.**

Debido a que se tienen muchas máquinas conectadas a la Internet, en las cuales se guardan miles de documentos, programas, imágenes, etc. que pueden ser de interés para alguien y que muchas veces no sabemos la localización de las mismas; un grupo de personas de la Universidad de McGill en Canadá creó un sistema de consulta llamado **archie**. La función principal es la de examinar y buscar estos archivos en muchas máquinas que ofrecen el servicio de transferencia de archivos de forma anónima. Debido al gran éxito del programa fue creciendo hasta convertirse en todo un servicio que da la Internet. El servicio es accesado mediante una sesión interactiva de **telnet** o mediante preguntas por correo las cuales son comandos de archie.

Actualmente el servicio de **archie** cubre 800 máquinas en la Internet conteniendo millones de archivos que en conjunto forman una base de datos de 50 gigabytes y esto crece diariamente. Para acceder a **archie** podemos hacer un **telnet** a los siguientes servidores:

```
archie.ans.net          (NY, USA)
archie.rutgers.edu     (NJ, USA)
archie.mcgill.ca      (Canadá)
archie.funet.fi       (Finlandia)
... entre otros ...
```

Cuando uno de los servidores de el prompt de **login**: se debe de teclear **archie** para poder acceder al servicio. A continuación se desplegará un mensaje de bienvenida así como información de interés, después de esto se quedará en el prompt **archie>** en donde el usuario puede teclear comandos.

Para mostrar su uso, veamos las siguientes pantallas, primero establecemos la conexión vía telnet de la siguiente forma:

```
Uziel21 2:> telnet 198.49.45.10
Trying 198.49.45.10 ...
Connected to 198.49.45.10.
Escape character is '^I'.
      InterNIC Directory and Database Services

Welcome to InterNIC Directory and Database Services provided by AT&T.
These services are partially supported through a cooperative agreement
with the National Science Foundation.

First time users may login as guest with no password to receive help.

Your comments and suggestions for improvement are welcome, and can be
mailed to admin@ds.internic.net.

AT&T MAKES NO WARRANTY OR GUARANTEE, OR PROMISE, EXPRESS OR IMPLIED,
CONCERNING THE CONTENT OR ACCURACY OF THE DIRECTORY ENTRIES AND
DATABASE FILES STORED AND MAINTAINED BY AT&T. AT&T EXPRESSLY
DISCLAIMS AND EXCLUDES ALL EXPRESS WARRANTIES AND IMPLIED WARRANTIES
OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE.

*****
DSO will be rebooted every Monday morning between 8:00AM and 8:30AM est.

SunOS UNIX (ds)

login: █
```

Una vez establecida la conexión, el sistema remoto nos pide una clave para tener acceso al mismo, en este punto tecleamosarchie como se muestra a continuación:

```

kterm
Your comments and suggestions for improvement are welcome, and can be
mailed to admin@ps.internic.net.

AIEI MAKES NO WARRANTY OR GUARANTEE, OR PROMISE, EXPRESS OR IMPLIED,
CONCERNING THE CONTENT OR ACCURACY OF THE DIRECTORY ENTRIES AND
DATABASE FILES STORED AND MAINTAINED BY AIEI. AIEI EXPRESSLY
DISCLAIMS AND EXCLUDES ALL EXPRESS WARRANTIES AND IMPLIED WARRANTIES
OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE.

*****
D50 will be rebooted every Monday morning between 8:00AM and 8:30AM est.

*****

SunOS UNIX (d5)
login:archie
*****

Welcome to the Internet Directory and Database Server.

*****

# Bongip Information Systems, 1993
# terminal type set to 'vt100 24 80'.
# 'erase' character is '^?'.
# 'search' (type string) has the value 'sub'.
archie>
archie>
archie> █
```

Una vez teclada la clave, el sistema responde con un mensaje de bienvenida y algunas instrucciones. Hasta este punto nos presenta el prompt dearchie> mencionado anteriormente. A continuación el sistema esta listo para recibir algún comando, en este caso, por ejemplo, una búsqueda; el comando prog nos permite hacer una búsquedas de archivos que se encuentran en la Internet, en la siguiente ventana hacemos una búsqueda del programa de indexamiento wais:

```

xterm
SunOS UNIX (ds)
login: archie

*****
Welcome to the InterNIC Directory and Database Server.
*****

# Bungip Information Systems, 1993
# Terminal type set to "vt100 24 80".
# "erase" character is "?".
# "search" (Type string) has the value "sub".
archie>
archie>
archie> prog wais
# Search Type: sub.
# Your queue position: 4
# Estimated time for completion: 02:16
working... █

```

archie empezará a buscar en su base de datos alguna referencia de wais, una vez encontrada muestra los resultados como sigue:

```

xterm
Host freebsd.cdrom.com (192.216.222.5)
Last updated 06:41 10 Jul 1994

Location: /3/languages/perl/scripts/gopher
DIRECTORY drwxr-xr-x 512 bytes 01:00 10 Dec 1993 wais

Host ftp.uni-kl.de (131.246.9.95)
Last updated 05:46 3 Jul 1994

Location: /pub2/packages/doc.tum/infosystems
DIRECTORY drwxr-xr-x 512 bytes 23:00 23 Sep 1992 wais

Host ftp.cc.columbia.edu (128.59.39.2)
Last updated 06:02 13 May 1994

Location: /
DIRECTORY drwxr-xr-x 512 bytes 16:04 28 Jan 1994 wais

Host ccsun.unicamp.br (143.106.1.5)
Last updated 07:35 2 May 1994
█

```

La información que nos da es la siguiente: nos indica en que máquina (host) se encuentra dicho archivo, nos da su FQDN y su dirección IP; después nos da la fecha de la última actualización, la ruta o **PATH** en la que se encuentra el archivo y a continuación nos indica si se trata de un archivo o de un subdirectorío dando permisos, tamaño, hora y fecha de creación y nombre del mismo. Teniendo esta información, podemos hacer un **FTP** a dicha máquina (host) para traer el archivo de interés.

• TELNET

Telnet es el principal protocolo Internet para crear conexiones con máquinas remotas. Nos da la oportunidad de estar virtualmente en una computadora y estar realizando un trabajo en otra, la cual puede estar a unos cuantos metros o quizá hasta el otro lado del mundo.

Su uso es muy sencillo y solo basta con conocer el FQDN o la dirección IP de la máquina a la cual queremos tener acceso. Dependiendo de las implementaciones del programa y el sistema que se utiliza, éste se puede llamar de diferentes formas sin embargo, el más común es el `telnet`. La forma de uso es la siguiente:

```
telnet algunlado.dominio
```

Para dar un ejemplo, podemos hacer un `telnet` a nuestra propia máquina si es que sabemos su dirección, tomemos `tzetzal.dcaa.unam.mx`.

```
telnet tzetzal.dcaa.unam.mx
```

Entonces nuestro sistema responderá con algo similar a esto:

```
Trying 132.248.27.10....
```

Connected to tzetzal.dcaa.unam.mx
Escape character '^_']^_].

ESTA TESIS NO DEBE
SALIR DE LA BIBLIOTECA

El *caracter de escape*, en este ejemplo "^_]" (Control]), es el caracter que nos regresará a nuestro sistema interrumpiendo la conexión con el sistema remoto, para suspender la sesión, etc.

Muchos de los programas de *telnet* dan la posibilidad de añadir un tercer parámetro, el cual es el *puerto* por el cual se efectuará la conexión. Normalmente este puerto es el 23 y es el default. Sin embargo muchas veces se tiene la necesidad de establecer una conexión a un puerto diferente, ya que en él existe algún tipo de servicio (por ejemplo consultas a WAIS, GOPHER, etc.). Usando:

```
telnet algunlado.dominio puerto
```

nos conectará a algunlado.dominio utilizando el puerto respectivo. El presente trabajo se vale de dicha característica para la implementación de seguridad y acceso al sistema de consulta de historias académicas.

Un ejemplo de conexión se muestra a continuación:

```

xterm
Lizetzal 5:27 telnet lizetzal.dcaea.unam.mx
Trying 132.243.27.10...
Connected to lizetzal.dcaea.unam.mx.
Escape character is '^]'.

UNIX System V Release 4.0 (Lizetzal)

login: edgar
Password:
Last login: Mon Jul 11 09:30:54 1993 from pallis
Sun Microsystems, Inc.  SunOS 5.2.2    Generic March 1993

***** ATTENTION DEBARRIOS *****

Recuerden que para manejar el importante archivo de respaldo,
el comando es: rcp.

***** HORARIOS DE RESPALDO *****

Para ver los horarios de respaldo, durante los cuales el servidor no
pueda ser accedido, teclee:

      news respaldo

*****
you have mail
Lizetzal 5:27
    
```

Además de estos servicios se tienen muchos otros, como pueden ser **finger**, **whois**, **ping**, **talk**, **Servicio de noticias**, **Servicios Comerciales**, etc. Además de muchos servicios que se están implementando en este momento dentro de la Internet.

Debido a que toda esta información es muy valiosa para el usuario que por primera vez tienen contacto con la red, se tienen archivos distribuidos en la Internet denominados **RFCs (Request For Comments)**, los cuales dan información de servicios, estándares, etc. de la Internet. El proceso en general en la creación de un RFC es que alguien describa algún servicio o algún tema de interés y mandarlo por correo a Jon Postel (postel@isi.edu). Jon actúa como un arbitro de la propuesta. Después es comentada a la comunidad de la Internet para ver quien quiere tomar parte de la discusión de dicho documento. Después

de las críticas y revisiones, se tienen varias versiones del documento, una vez aceptado se le asigna un número de RFC.

Los RFC pueden dividirse en 5 grupos: requeridos, sugerencias, direccionales, informativos y obsoletos.

Los *requeridos* (por ejemplo el RFC-791; el protocolo IP) deben de ser implementado en cualquier máquina que desee conectarse a la Internet.

Las *sugerencias* son normalmente implementaciones dentro de sistemas de la red, que, si bien no son necesarios para su funcionamiento, pueden impactar su desempeño (*usability and performance*).

Los RFC *direccionales* se refieren a discusiones que fueron aprobadas, pero la implementación de la discusión nunca se llevó a cabo. Puede ser debido a la falta de una necesidad global de la Internet, tecnología no desarrollada, etc.

Los RFC *informativos* contienen información acerca de la Internet y de su operación, como el RFC-990 -asignación de direcciones-, o el RFC-1462 -¿Qué es la Internet?-

Existe un subconjunto de RFC que se denominan ***FYI (For Your Information)***, los cuales están escritos en un lenguaje informal que es diferente del usado en los RFC's. Los temas que se tienen son los de respuesta a preguntas comunes, experiencias, problemas técnicos, bibliografía recomendada, etc.

Finalmente, como la Internet ha crecido y su tecnología ha avanzado, algunos RFC's caen dentro de lo obsoleto. Estos RFC's *obsoletos*, sin embargo, no pueden ser ignorados, ésto se debe a que cuando un nuevo RFC sale y convierte a otros obsoletos, generalmente el nuevo hace solamente referencias a los anteriores, por lo que para la buena comprensión del nuevo, se requiere leer o documentarse acerca de los obsoletos.

RFC's y FYI's se encuentran disponibles vía FTP anónimo en las siguientes máquinas:

- `nic.ddn.mil` bajo `/rfc/rfc-xxxx.txt`, donde 'xxxx' es el número del RFC.
- `ftp.uu.net` bajo `/RFC`.

También se tienen disponibles vía EMail escribiendo a `service@nic.ddn.mil`, con el "*subject:*" `send RFC-xxxx.TXT`, donde 'xxxx' es el número del RFC solicitado.

Para finalizar solo resta describir **WWW World Wide Web**. El proyecto WWW mezcla las técnicas de adquisición de información e hipertexto para el desarrollo de sistemas de información de una forma muy fácil pero muy poderosa. El mundo de WWW consiste de documentos y ligas. También se incluyen índices de archivos, los cuales son un tipo especial de documento, que en vez de consultar algún tipo de información, vemos listas de archivos.

Al acceder un documento podemos observar dentro de él ligas a otros documentos en forma de hipertexto. Estas ligas pueden ser a archivos locales o a

archivos distribuidos en las máquinas que forman la Internet, por lo que su campo de acción es a nivel mundial. Para poder seguir una liga, simplemente accedemos a ella vía *mouse* o con alguna combinación de teclas. Las ligas no simplemente son a documentos, sino que pueden ser ligas a **hipermedios**, es decir, pueden ser ligas a documentos, imágenes, gráficas, sonidos e imágenes en movimiento. Como se puede pensar, las aplicaciones de esta nueva forma de consulta de documentos no tiene fin, ya que en un solo documento podemos anexar texto con explicación, la imagen a la que nos referimos, apoyada por audio y animación en video.

Así, la Internet nos da la oportunidad de entrar a un mundo de información y conocimientos muy amplios, de cualquier tópico y con la profundidad que se desee.



Capítulo II

Planteamiento del problema.

En este capítulo discutiremos el planteamiento del problema, primeramente describiendo los recursos disponibles con que se cuenta actualmente, para después pasar a lo que se refiere al análisis del problema al que queremos dar solución.

Para ello, se hace una descripción del sistema actual de microfilmación de las historias académicas, para después dar paso a las propuestas de solución y finalmente a la elección de una de ellas.

1.

Recursos Disponibles.

A continuación se presenta una descripción del equipo con que cuenta la Dirección de Cómputo para la Administración Académica así como las capacidades del mismo.

1.1 Descripción y características del equipo.

La **DCAA** se caracteriza por tener un equipo de cómputo acorde con las actividades que esta desempeña, así como una gran plataforma de software que ayuda a dichas actividades.

Por ello, consideramos importante la descripción de las características de hardware y de software de los equipos utilizados debido a que estas juegan un papel fundamental en el desempeño de los sistemas que en ellos corren. La descripción se centra principalmente en tres equipos que se utilizan actualmente en el proceso de generación y almacenamiento de la información de las historias académicas, además de un equipo que se utilizará para dicho propósito y de la red de Cómputo Universitaria.

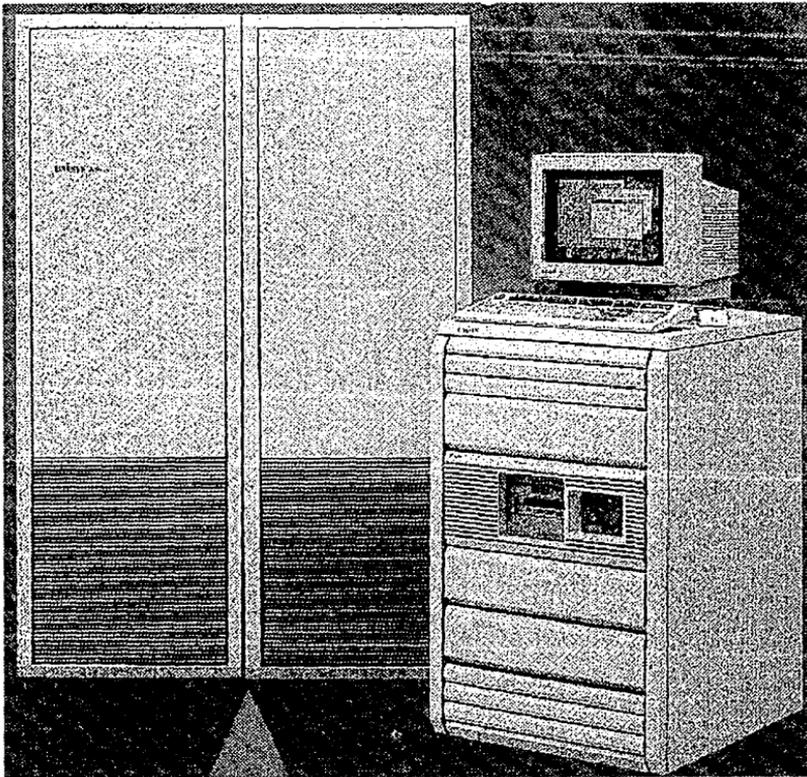
1.1.1 Equipos A-12 de Unisys.

La Universidad Nacional Autónoma de México para desarrollar muchas de sus funciones administrativas y de investigación, necesita de diferentes equipos de cómputo con diferentes capacidades que van desde las microcomputadoras (PC's) hasta una Supercomputadora, pasando por estaciones de trabajo y macrocomputadoras (mainframes). Dentro de esta última categoría es donde la UNAM tiene una amplia experiencia ya que desde la década de los 50's ha tenido este tipo de equipo; y desde finales de los 60's ha tenido dentro de sus principales sistemas para la administración central y escolar mainframes de la misma familia¹: Serie B5000 y B6000 de Burroughs pasando después a la familia Serie A de Unisys. De esta última daremos sus principales características como familia así como algunas consideraciones particulares de los sistemas que se tienen en la Universidad Nacional Autónoma de México.

La línea de equipos Serie A de Unisys presentan características arquitectónicas adicionales y mayor rendimiento, manteniendo la compatibilidad con sus antecesores de las series B5000, B6000 y B7000², esto se logró empleando en el diseño y la construcción de los equipos la más avanzada tecnología.

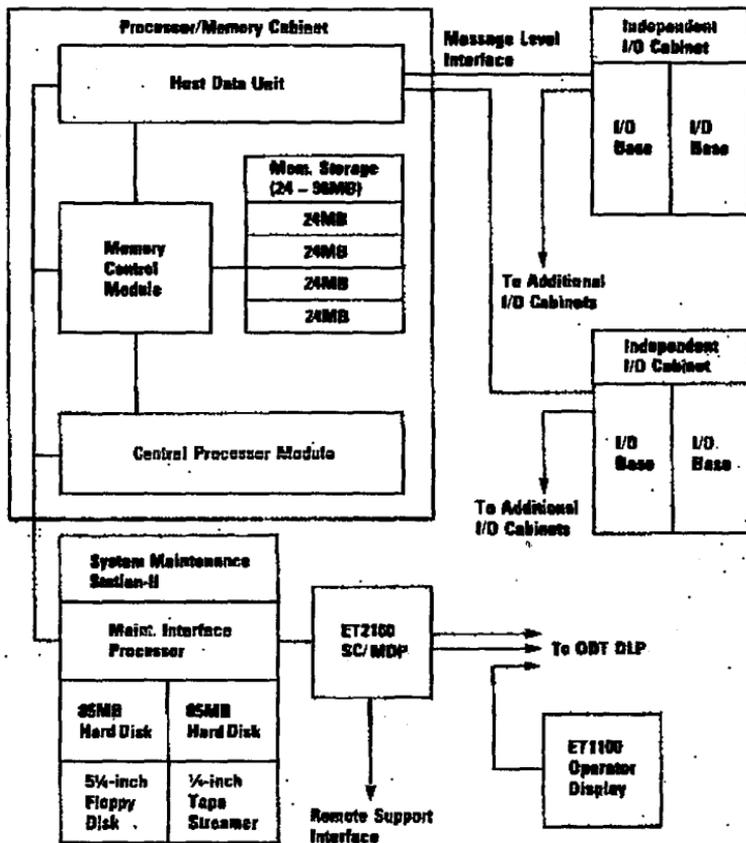
¹ Para una mayor información favor de referirse al Capítulo I, tema 1 "Antecedentes".

² Como explicamos en el capítulo anterior Unisys surge de la fusión de Burroughs y Univac, y la serie A es la heredera de las series mencionadas anteriormente, las cuales pertenecían a Burroughs.



Los sistemas de la Serie A ofrecen una amplia gama de computadoras centrales, terminales de propósito general o específico, impresoras, procesadores de documentos, procesadores de comunicaciones y dispositivos de almacenamiento de datos. A continuación se presenta el diagrama de bloques de un sistema Serie A.

A12 System



Entre las características más sobresalientes de los equipos de la Serie A cabe mencionar:

- Tienen una alta eficiencia en la ejecución de programas escritos en lenguajes de alto nivel, comparable a la de ensambladores y lenguaje de máquina en otros equipos.

- Permiten un fácil incremento de productividad mediante el uso de herramientas de tercera y cuarta generación sin afectar el rendimiento.
- Requieren de un mínimo esfuerzo para la implementación de sofisticadas redes de comunicación en línea bajo el esquema de manejo de datos transaccional.
- Su hardware es totalmente modular, lo que proporciona una gran facilidad y libertad de expansión.
- El hardware y el software están acoplados para ajustar la operación del sistema de la manera más eficiente.
- Independencia entre la configuración de hardware y los programas de aplicación y el sistema operativo.

Cada sistema de la Serie A requiere para su funcionamiento de una configuración mínima de módulos de hardware, misma que dependerá del modelo del equipo. Estos módulos deberán integrarse para formar los siguientes subsistemas:

- Subsistema de Procesamiento.
- Subsistema de Memoria.
- Subsistema de Entrada/Salida.
- Subsistema de Mantenimiento.
- Subsistema de Comunicaciones.

A la configuración básica se le pueden adicionar nuevos módulos y dispositivos periféricos para incrementar la capacidad y rendimiento. El número total y tipo de módulos a ser conectados dependerá del modelo del sistema y de las necesidades particulares del usuario.

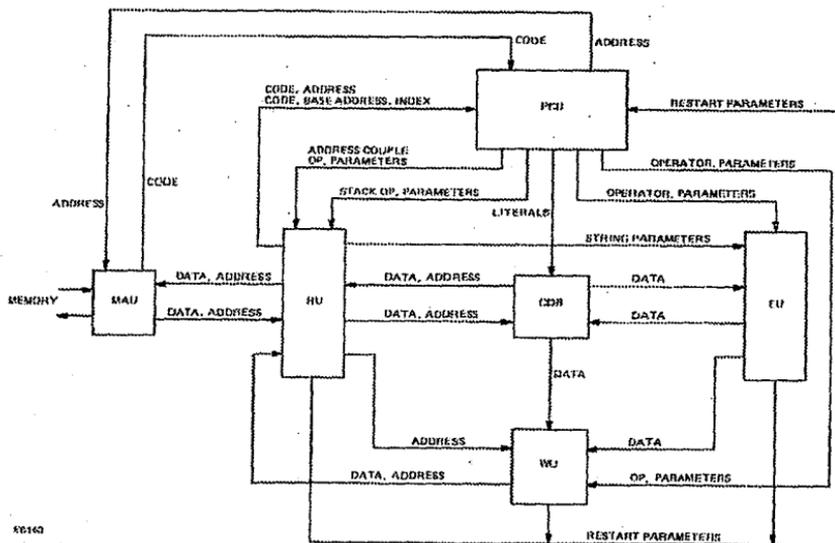


SUBSISTEMA DE PROCESAMIENTO

El Subsistema de Procesamiento o Unidad Central de Proceso esta constituido por una serie de procesadores microprogramados y dispositivos de almacenamiento y manejo de información que operan en conjunto para ejecutar los programas del sistema y de aplicación. Para que un programa sea ejecutado debe ser primero convertido a un conjunto de instrucciones comprensibles para el procesador. Este conjunto de instrucciones forman un programa escrito en lenguaje de máquina llamado firmware (concepto característico de UNISYS), y a su código ejecutable se le denomina microcódigo. El microcódigo es la fuente de control para la ejecución de instrucciones. El firmware no puede ser programado ni modificado por el usuario. Para la implementación y ejecución del microcódigo

se utilizan lenguajes de alto nivel. La Unidad Central de Proceso también controla a otros módulos de alto nivel de procesamiento, cada uno de los cuales tiene una función específica, por medio de una estructura de MultiBus en paralelo. El concepto de actividades coordinadas de múltiples módulos de hardware junto con el microcódigo, constituyen la técnica de procesamiento distribuido implementada en la arquitectura de los sistemas de la Serie A.

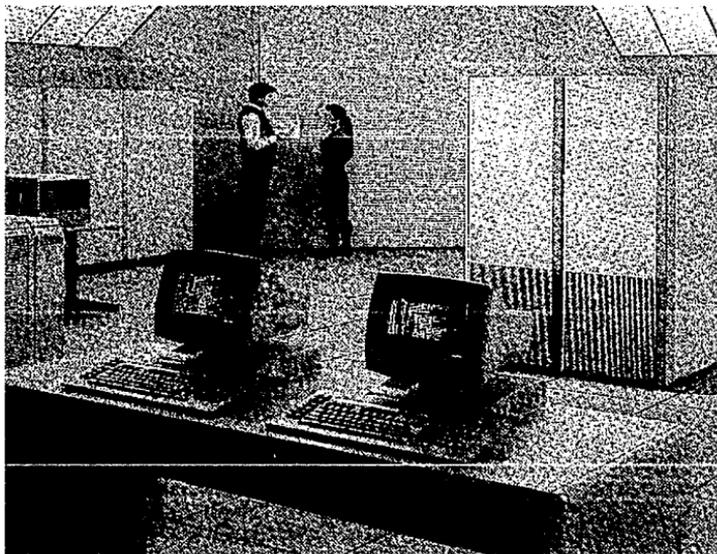
El Módulo del Procesador Central (CPM) utiliza unidades funcionales concurrentes. La comunicación entre estas unidades es llevada a cabo mediante cache de alta velocidad y colas de operaciones. La siguiente figura presenta el diagrama de bloques del CPM:



8E16Q

El procesador consiste de cinco áreas funcionales principales que son :

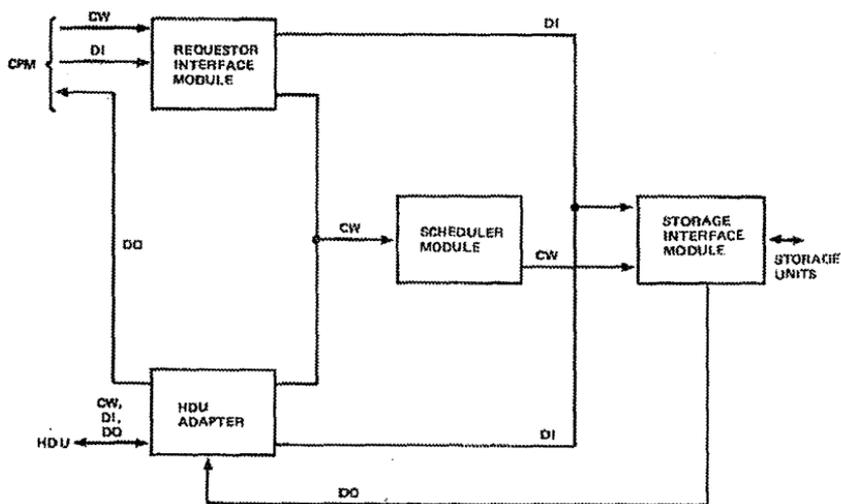
- Program Control Unit (PCU).
- Reference Unit (RU).
- Execution Unit (EU).
- Write Unit (WU).
- Memory Access Unit (MAU).



SUBSISTEMA DE MEMORIA

En la construcción del Subsistema de Memoria de los equipos de la Serie A se emplea la más reciente tecnología de memoria de alta densidad que permite incrementar considerablemente la confiabilidad, potencia, velocidad y capacidad

de almacenamiento. El Subsistema de Memoria es el encargado del control del almacenamiento y del manejo de la transferencia de datos entre el procesador y la memoria principal, para cumplir con estas funciones esta constituido por las unidades de almacenamiento y una o más unidades de control.



CW = CONTROL WORD
 DI = INPUT DATA
 DO = OUTPUT DATA

#5485

La Unidad de Control de Memoria o MCU (Memory Control Unit) es la encargada de establecer las interfases entre las unidades o módulos de memoria principal y el procesador central. En los equipos que cuentan con más de un procesador central se utiliza un módulo adicional llamado SAM (Shared Access Memory) que actúa como interfaz entre los procesadores y los módulos de almacenamiento de memoria. Adicionalmente el Subsistema de Memoria cuenta

con circuitería especial para detección y corrección de errores. Estos circuitos están capacitados para detectar errores en múltiples bits y poder corregir errores de un solo bit, sin degradar el rendimiento del sistema. También contiene una Unidad de Almacenamiento de Memoria o Memory Storage Unit (MSU) la cual tiene las tarjetas con la circuitería de memoria.

El A12 usa palabras de 60 bits que se dividen en 48 bits de datos, 4 bits de control y 8 bits de corrección de errores.



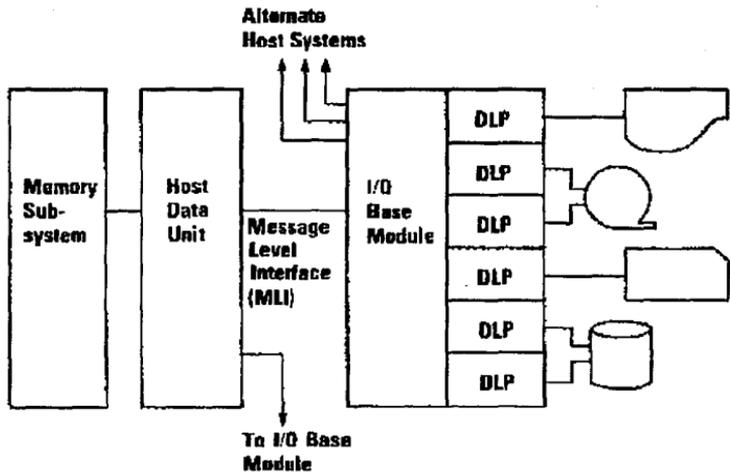
En los sistemas de la Serie A, el rendimiento del procesador es incrementado por medio del uso de almacenamiento de código y de datos en Memoria Cache. La Memoria Cache es un pequeño componente de la memoria principal usado para almacenar el contenido de las localidades de memoria referenciadas recientemente, hace uso de mecanismos que permiten que los tiempos de acceso sean sensiblemente menores que a memoria normal.

Unisys ha optimizado el diseño de memoria de los equipos de la Serie A de tal manera que actualmente la mayor parte de los accesos a memoria pueden ser atendidos por la Memoria Cache.

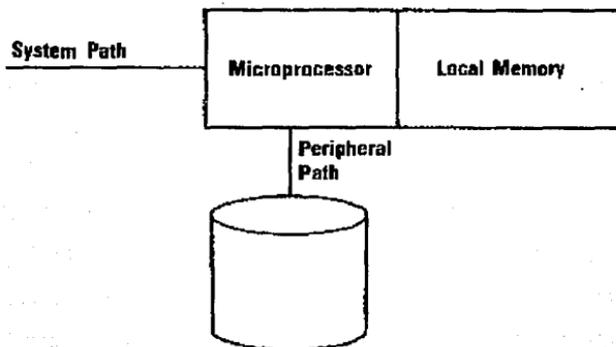
SUBSISTEMA DE ENTRADA/SALIDA

El Subsistema de Entrada/Salida es el encargado de manejar y controlar todas las transferencias de información entre la unidad central de proceso y todos sus dispositivos periféricos. El Subsistema de Entrada/Salida esta formado por una serie de Procesadores Especializados llamados DLP's (Data Link Processors), interfases, controladores y dispositivos periféricos. Las operaciones de I/O (Input/Output, Entrada/Salida) son iniciadas en la Unidad Central de Proceso e inmediatamente el control es transferido al DLP correspondiente, de tal manera que la carga de operaciones de I/O se distribuye permitiendo la atención de múltiples requerimientos de I/O en forma simultánea. En cada DLP corre un programa en microcódigo con las características únicas y funciones dependientes especiales para cada tipo de dispositivo periférico, por lo que existirá un DLP específico para cada tipo de dispositivo. El diagrama del subsistema de I/O y DLP es el siguiente:

A12 Input/Output Subsystem



Data Link Processor



El subsistema también contiene el Host Data Unit (HDU) el cual es una unidad mainframe microprogramable responsable del servicio de colas de I/O independientemente desde el CPM.

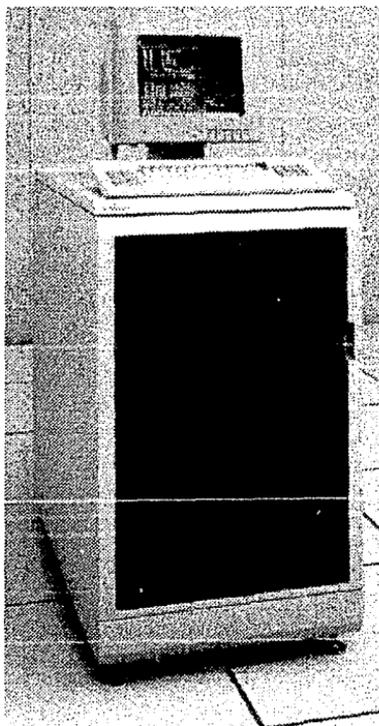


SUBSISTEMA DE MANTENIMIENTO.

El núcleo del Subsistema de Mantenimiento esta formado por un procesador especializado llamado UIP (User Interface Processor) y esta programado para cumplir con múltiples funciones, entre las que destacan:

- Establecer una interfaz que permita al operador controlar el hardware cuando el sistema va a ser inicializado, configurado, detenido o cuando el software va a ser cargado por primera vez.

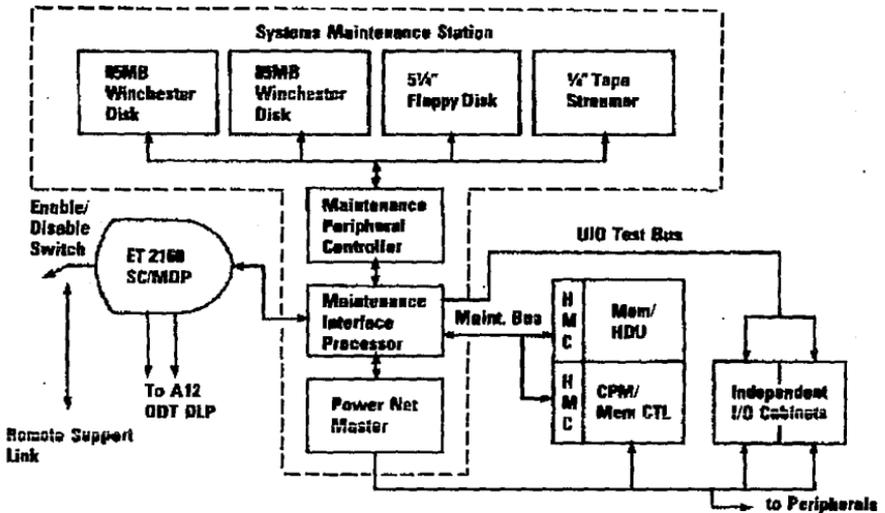
- Proporcionar y ejecutar pruebas de diagnostico para todos y/o cada uno de los componentes de los Subsistemas de Procesamiento, Memoria, Entrada/Salida y Comunicación de Datos.
- Proveer una interfaz para enlace de Diagnostico Remoto, esto es, que las rutinas de diagnostico serán ejecutadas y monitoreadas desde otro computador que funciona como una estación de servicio remoto.



El Subsistema de Mantenimiento cuenta, además, con un Microprocesador de Control de Encendido con reloj de tiempo y una batería de respaldo que mantiene

almacenada la información de fecha y hora. En algunos modelos se incluye la opción de programación automática de encendido y apagado, por tiempo o por fecha y hora. En modo de diagnóstico puede controlar la habilitación y deshabilitación de los márgenes de potencia de los módulos lógicos, esta función puede ser establecida por consola de operación o en forma remota. Su diagrama es el siguiente:

A12 Maintenance Subsystem

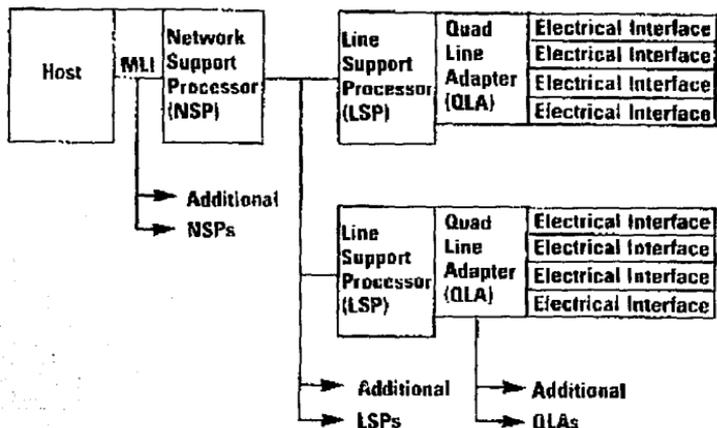


SUBSISTEMA DE COMUNICACIONES.

El Subsistema de Comunicaciones es la liga de comunicación entre los dispositivos remotos tales como terminales y otros hosts (computadoras con

procesamiento independiente), y el sistema central. Este Subsistema realiza sus funciones de comunicaciones desde y hacia el procesador central por medio de procesadores especializados tales como: el Network Support Processor (NSP), el LINE Support Processor (LSP), el Data Communications Data Link Processor (DCDLP), el Communications Processor 2000 (CP2000), Integrated Communication Processor (ICP) y los más recientes liberados EDC/DLP y CPDLP. A continuación presentamos el diagrama de bloques:

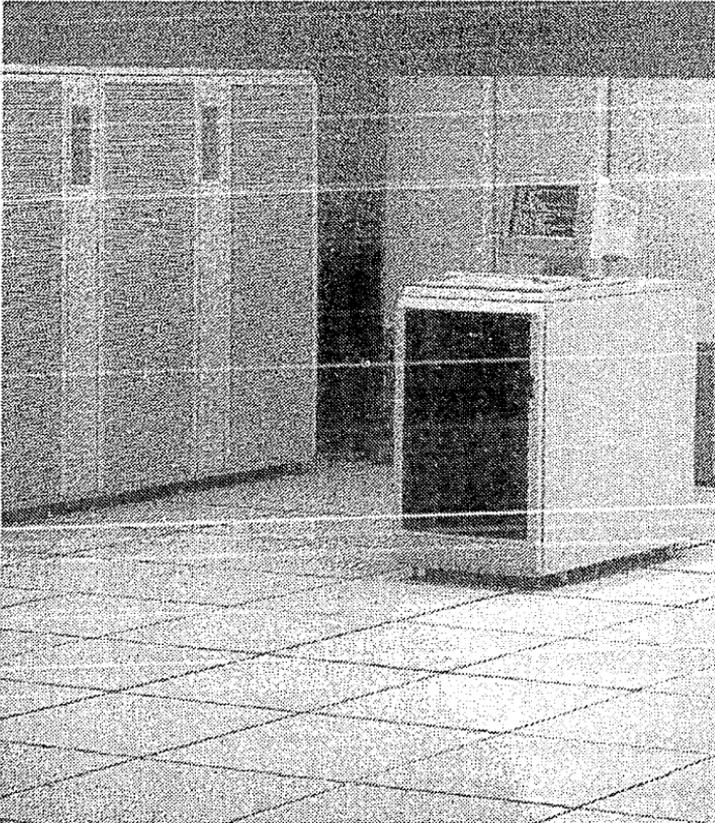
Data Communications Subsystem



El NSP, el LSP, el CP2000, el CPDLP y el EDC/DLP son totalmente programables y están capacitados para manejar la implementación de varios protocolos de comunicación diferentes, tanto propios como estándares tales como: SNA, X.25, TCP/IP y OSI.

El DCDLP no es programable por el usuario, viene previamente programado para trabajar solamente con los protocolos Poll-Select y TTY.

Los niveles más bajos de la interfaz de comunicaciones hacia las líneas son manejados por un dispositivo llamado Line Adaptor (LA).



Habiendo ya analizado en forma general a la familia Serie A de Unisys, la cual en cualquiera de sus diferentes modelos cumple con estas características, pasemos a ver los modelos con los que cuenta la UNAM.

La Universidad Nacional Autónoma de México cuenta actualmente para su administración escolar y algunos temas de investigación con dos mainframes de Unisys de la Serie A, el sistema A12. El sistema A12 esta disponible en tres modelos: el A12B, el A12, y el A12T. La UNAM cuenta con un sistema A12B y un A12; cuentan con 4 MW (MegaWord, 1 MW = 1*48 megabytes) en RAM cada una. En la figura presentamos el equipo en su conjunto



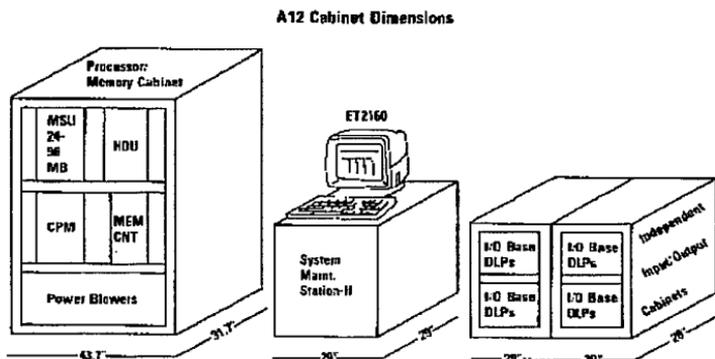
- | | |
|---|-----------------------------------|
| 1.- Unidad de mantenimiento | 5.- Consola de operación |
| 2.- Unidad de almacenamiento (discos) | 6.- Sistema de aire acondicionado |
| 3.- Unidad de almacenamiento (cintas) | |
| 4.- Módulo del procesador y de la memoria | |

El sistema operativo de la Serie A es el MCP/AS (Master Control Program/Advanced System), y este es un programa especial encargado de controlar el medio ambiente operacional del sistema mediante la asignación y distribución de recursos, el control de la ejecución de procesos, la asignación de prioridades, etc. Actualmente se tiene instalada la versión 4.0.1 del MCP/AS.

Como conclusión podemos decir que los sistemas A12 y A12B son sistemas robustos, seguros, que soportan cargas de trabajo pesadas sin bajar el performance (calidad) de su respuesta, maneja lenguajes de alto nivel como si fueran de bajo nivel, entre otras características. Como desventajas tenemos su alto costo, ser un sistema propietario y cerrado siendo la tendencia actual sistemas abiertos (como los sistemas Unix), además su implementación para redes publicas basadas en el modelo de protocolo TCP/IP no es fácil ni la más óptima aunque en su implementación de redes propietarias, entre los mismos equipos, es bastante aceptable por medio de su arquitectura de red como lo es BNAV2 (Burroughs Network Architecture versión 2).

La velocidad a la que trabaja el procesador central es de 16.25 MHz, aunque esta puede variar entre 16.00 y 16.50 MHz. Esta velocidad parecería bastante lenta a comparación de cualquier PC 386 o 486 las cuales llegan a tener velocidades de hasta 66 MHz; pero hay que tener en cuenta la cantidad de circuitería que maneja una PC y la que maneja un mainframe; podemos decir que la mother board (tarjeta principal) de una PC no es ni la mitad de una de las decenas de tarjetas que constituyen a un equipo de la serie A y que por lo tanto se necesita tener un control mucho mas preciso de las operaciones realizadas por cada una de estas tarjetas, sacrificando la velocidad a la que funcionan. Aun con esta 'limitante' no podemos comparar la velocidad con que se resuelven las

operaciones en un equipo serie A, que cuenta con muchos mas recursos, que el tiempo que tarda una PC o hasta una minicomputadora contemporánea de estos equipos.



Por su tamaño, consumo de energía y calor que generan estos equipos es necesario que estén instalados en lugares con características especiales como:

- Líneas trifásicas de electricidad; el equipo tiene un consumo de energía de 16.2 KVA (Kilo Volts Amper); cuenta con plantas "no break" para el caso en que el suministro normal de electricidad falle, entren en funcionamiento estos equipos para que la computadora en ningún momento tenga fallas por falta de flujo eléctrico; en dado caso que esto también falle y se llegue a apagar la computadora entonces al volver a levantar el sistema este ira a revisar el log (archivo donde se guarda cada una de las actividades que ha realizado el mainframe en su conjunto) y obtiene que es lo que estaba realizando antes de registrarse la falla, pasando entonces a reiniciar desde el principio los programas las impresiones y todas las actividades en general que estaba realizando; por esto ultimo se le considera un sistema semi-tolerante a fallas:

- Tiene una disipación de calor de 49,600 BTU/hr, por lo que es necesario contar con equipos enfriadores de aire para que el equipo, dependiendo de sus características se encuentren a: 19 °C para el equipo central y sus subsistemas, 21 °C para algunos periféricos como unidades de cinta y controladores de red (CP2000) y 23 °C para otros periféricos como impresoras. Además de enfriar el aire estos equipos tienen un sistema de filtrado y en general las instalaciones tienen características que impiden el paso de polvo en grandes cantidades hacia la sala de cómputo.
- Por las características del cableado es necesario que exista tanto un plafón falso como un piso falso para que por abajo de este estén todas las conexiones necesarias. El equipo ocupa un área de 27 pies cuadrados (2.51 metros cuadrados) esta área incluye los gabinetes del procesador, I/O y la estación de mantenimiento.
- Por último se tienen sistemas detectores y extintores de incendios a base de gas halón para no dañar el equipo en caso de ser necesario su uso. Por supuesto también existen sistemas de control de acceso a la sala de cómputo por medio de tarjetas controladas por código de barras y sistemas infrarrojos.

1.1.2 Lectura óptica

La introducción de las computadoras electrónicas en las organizaciones, ha significado la posibilidad de desarrollar sistemas de información automatizados, que apoyan las actividades de planeación, administración y control de una institución en forma más ágil.

Normalmente en un sistema, se realiza un gran esfuerzo en lograr que la programación de los procesos que realiza sea óptima y utilizar los recursos de cómputo de la manera adecuada.

Sin embargo estos esfuerzos no siempre dan los resultados que se esperaban de ellos, ya que la problemática que se presenta en los sistemas de información, por lo general radica en la captura de datos.

Este problema se empezó a resolver con el uso de tarjetas perforadas, pero esta técnica, frecuentemente es crítica en la veracidad de la información, debido a las diferentes etapas que se requieren para poder acceder los datos al computador y por consecuencia el alto costo asociado a su manejo.

Otra alternativa de solución es la introducción de terminales remotas. Esta opción a dado buenos resultados para cierto tipo de información y en cantidades no muy grandes, pero resulta ser inoperante, cuando el volumen de datos se incrementa drásticamente y se requiere de un tiempo de respuesta considerablemente corto. Esto dificulta la actualización de las bases de datos oportunamente, provocando un atraso en los procesos subsecuentes y originando que el sistema resulte ser inoperante por razones prácticas y económicas debido al tiempo asignado.

Esta situación que constantemente se presenta, ha requerido de un análisis e investigación sobre técnicas que la resuelvan. Una solución a este problema ha sido la aplicación de la técnica de **lectura óptica de marcas**.

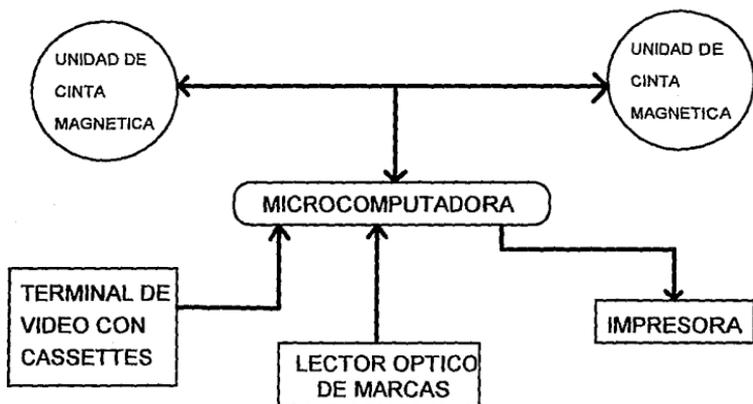
1.1.2.1 Descripción general

La UNAM a través de la DCAA, tiene en sus instalaciones dos equipos de lectura óptica de marcas. Se optó por la adquisición de estos equipos, debido a su versatilidad y confiabilidad para el tipo de aplicaciones a implantarse en el ámbito universitario.

El sistema de lectura óptica capta la información por medio de marcas plasmadas en el documento fuente (hoja óptica), su velocidad de lectura es de 85 hojas por minuto y tiene una capacidad máxima de 7,076 caracteres por hoja.

Como se muestra en la siguiente figura, el sistema está constituido por un lector óptico de marcas comunicado a un minicomputadora con los siguientes dispositivos periféricos:

- Dos unidades de cinta magnética.
- Una terminal de video con 2 unidades de cassette integradas.
- Una impresora.



Tipos de marcas captables

El equipo de lectura óptica capta las marcas hechas con el grafito de cualquier lápiz y la tinta negra de las cintas usadas en las impresoras de computadora.

Internamente el lector óptico cataloga cada marca plasmada en el documento fuente, procediendo a clasificarla por su intensidad en los cuatro tipos siguientes:

- Marca de tono fuerte.
- Marca de tono mediano.
- Marca de tono suave.
- No existe marca.

Características de la hoja óptica

El lector óptico lee y procesa hojas con un tamaño que puede variar desde 9" x 14" hasta 2.5" x 14", el gramaje de la forma oscila entre los 75 y 120 gramos y puede ser impresa en varios colores. El recomendado es de 90 g. sin embargo este varía de acuerdo a la aplicación, por ejemplo si la hoja va a ser impresa por ambos lados se recomienda el papel de 120 g.

El diseño del formato de la hoja es completamente libre, esto es: se pueden definir áreas muy pequeñas o tan grandes como la zona de datos. Cada área está constituida por uno o varios alvéolos que son fácilmente llenados con lápiz, permitiendo a la vez, la inmediata corrección de algún dato erróneo, ya que sólo se necesita borrar la marca del alvéolo incorrecto.

El espaciado entre alvéolos en forma horizontal es fijo, 6 alvéolos por pulgada, mientras que en forma vertical puede variar a 5,6 y 8 alvéolos por pulgada, lo cual significa que podemos variar la cantidad de alvéolos en una hoja, siendo el menos denso, el formato 6x5 y el mas denso el formato 6x8. También se puede diseñar la forma utilizando una combinación de distintas densidades.

La versatilidad del lector óptico no sólo permite diseñar una hoja óptica por aplicación, sino que se puede elaborar un conjunto de hojas, las cuales son presentadas en cuadernillos para facilitar su llenado y ampliar la cantidad de información a capturar.

Zonas de la hoja de lectura óptica

La hoja de lectura óptica se encuentra dividida en las siguientes cuatro zonas:

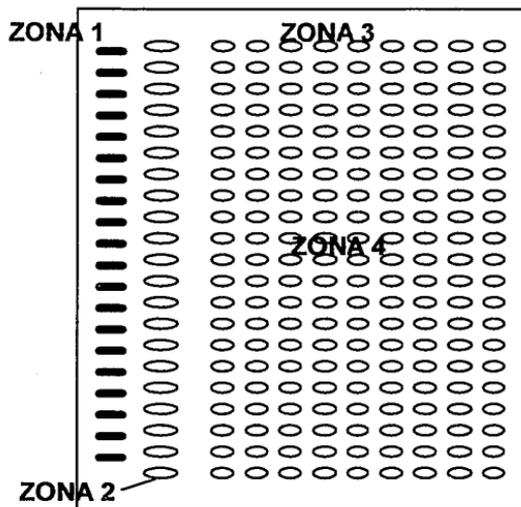
ZONA 1: *Zona de control de lectura.* Situada en el lado izquierdo del anverso y en el lado derecho del reverso de la hoja de lectura, se utiliza para indicar el número de renglones a leerse (5,6 u 8 alvéolos por pulgada). Las marcas en esta zona son conocidas como las barras de tiempo.

ZONA 2: *Zona de código.* Se utiliza para asignar y diferenciar el código de los tipos de hojas de lectura, de cada aplicación.

ZONA 3: *Zona libre de marcas.* Situada en la parte superior de la hoja y es utilizada para permitir al lector óptico reconocer el blanco relativo (color de la hoja), para así, poder detectar la intensidad de las marcas.

ZONA 4: *Zona de datos.* Permite contener hasta 7,076 posiciones de lectura, 3,538 posiciones en forma de alvéolo por cada lado de la hoja.

A continuación se muestran las zonas en una hoja para lectura óptica:

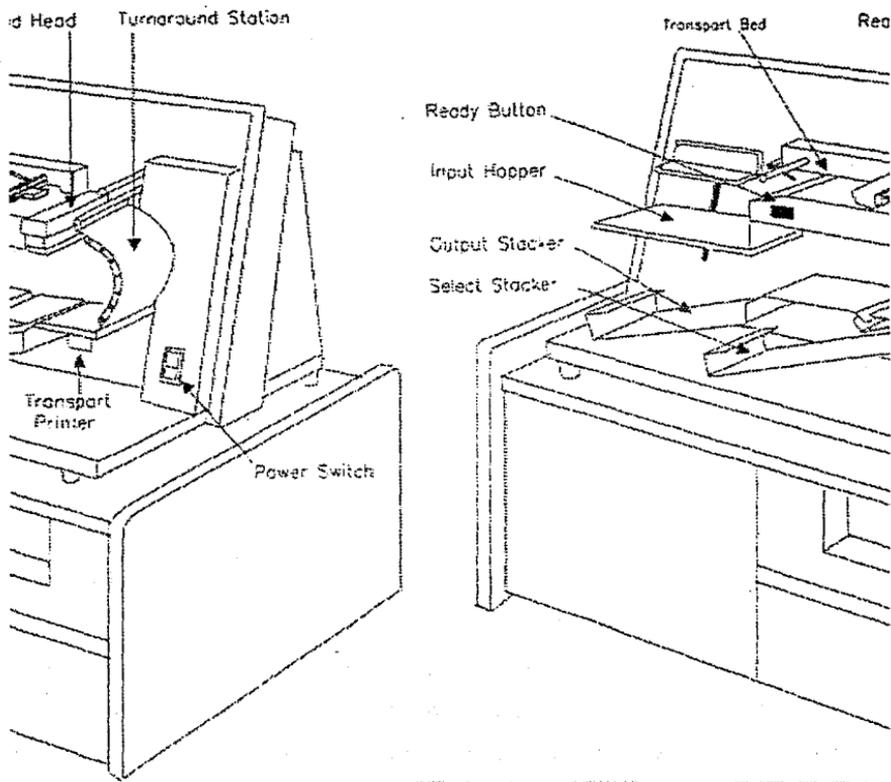


Descripción de los procesos de lectura óptica.

Como se muestra en la siguiente figura, se depositan las hojas de lectura óptica en el almacenamiento de entrada, cada hoja es alimentada a la zona de lectura mediante una banda de tracción, en donde se leen sus dos caras (anverso y reverso) a través de una cámara que reconoce las zonas de control y capta por reflexión de la luz, exclusivamente las marcas plasmadas en los alvéolos de las áreas de lectura previamente definidas, a una velocidad de 85 hojas por minuto.

Una vez leído el documento se va apilando en el almacenamiento de salida, o lo deposita automáticamente en el almacenamiento de error, si es que se detectó alguna discrepancia en la hoja óptica, pero sin captar su información.

De las hoja leídas correctamente, la información captada es transmitida al computador para posteriormente ser grabada en cinta magnética.



Sistemas de lectura óptica OPSCAN 10

Actualmente la Dirección de Computo para la Administración Académica, cuenta con dos lectores ópticos OPSCAN 10, cada uno con diferente configuración:

1. Un equipo con cabeza lectora de lápiz

Este lector puede leer solamente aplicaciones cuyos alvéolos hayan sido llenados con lápiz del número 2. La información preimpresa puede ser de cualquier color (azul, verde, café, rojo, etc.) excepto negro y tonalidades en gris.

2. Un equipo con cabeza lectora de tinta.

Este lector puede leer aplicaciones cuyos alvéolos hayan sido llenados con lápiz del número 2 y/o con tinta o bolígrafo color azul o negro. La información preimpresa de la hoja sólo podrá ser naranja, café o roja ya que de otra forma no se garantiza la calidad de la lectura.

Los lectores OPSCAN 10 están diseñados para trabajo pesado y velocidad media de lectura de formas (promedio máximo de trabajo 5,500 hojas por hora). Trabajan con la técnica de reflexión de luz, lo que significa que, cuando encuentran una marca de acuerdo a las posiciones que son programadas, dicha marca es reflejada y tomada como información. Es así como se convierten las marcas de las formas en caracteres almacenados en medios magnéticos. Puede discriminar hasta 16 niveles de grises.

Ambos lectores cuentan con una impresora interna que permite que, al pasar la forma leída, se imprima en ella la fecha, el número de forma leída, el número de corrida, etc., o un mensaje de error en caso de haber sido rechazada.

Aplicaciones en la UNAM

Las etapas que se siguen para aplicar la técnica de lectura óptica en la captura masiva de información, son las siguientes:

- Análisis para implantar este método de captura.
- Diseño del formato de la hoja óptica.
- Elaboración del programa de captura de datos.
- Pruebas de control de calidad en la producción de las hojas ópticas.
- Pruebas de verificación en la captura de la información.
- Proceso de lectura de hojas ópticas y captura de la información, la cual se graba en cinta magnética.

En la actualidad se han implantado cerca de 70 diferentes aplicaciones de esta técnica, en lo concerniente a:

- Exámenes de opción múltiple.
- Evaluaciones departamentales y de profesores.
- Pruebas psicológicas.
- Procesos administrativos de control escolar.
- Censos.

Ventajas

De acuerdo a los resultados obtenidos por cada una de las aplicaciones que se tienen en el lector óptico

- Confiabilidad de los datos capturados.
- Ahorro en tiempo de captura.
- Ahorro en costo rendimiento hasta un 80% en comparación con otros métodos de captura.
- Mayor densidad de información por documento leído, lo que representa una capacidad máxima incrementada en 80 veces el equivalente a una tarjeta perforada.

1.1.3 Unidad de microfilmación.

Otro de los recursos con que cuenta la Dirección de Cómputo para la Administración Académica (DCAA) es la **microfilmación**, como un medio de almacenamiento y consulta de los grandes volúmenes de información que se manejan. Para el caso de las historias académicas no es la excepción. La capacidad de esta unidad es la de almacenar en una sola microficha 208 hojas de tamaño carta de información a una velocidad de 70 microfichas por hora. Sin embargo esta tecnología es muy vieja: el equipo que se tiene data de 1978 y en lo que respecta a su mantenimiento y desempeño es superado, y por mucho, por las nuevas tecnologías de almacenamiento masivo de información (ya sea magnético, óptico o una tecnología híbrida).

Sin embargo, esta unidad cumplió con su tarea durante mucho tiempo pese a su obsolescencia y difícil mantenimiento (además de costoso). A continuación se dará una descripción del equipo.

1.1.3.1 Descripción general.

Antes de pasar a la descripción de la microfilmación y de varios conceptos relacionados con el tema, se mostrará las características generales del equipo:

- Unidad microfilmadora marca *DATAGRAPHIX*.
 - Modelo *AUTOCOM* con dos unidades de disco flexible de 8".
 - Unidad de cinta de 1600 BPI de densidad.
- Impresora de papel térmico *TEXAS INSTRUMENTS*
 - Modelo *SILENT 700*.
 - Utiliza rollos de película fotográfica.
 - Utiliza químicos para el revelado de las microfichas.
- Cada microficha contiene la información de 208 hojas tamaño carta.
- Velocidad de 70 microfichas por hora (teórico).
- Además se cuenta con una lectora de microfichas para la consulta de la información

PROCESO DE MICROGRAFICACIÓN.

Micrograficación es el proceso mediante el cual se reduce el tamaño del medio en donde reside la información a imágenes fotográficas o fílmicas (**film**), comúnmente llamadas **microformas**, las cuales son demasiado pequeñas para

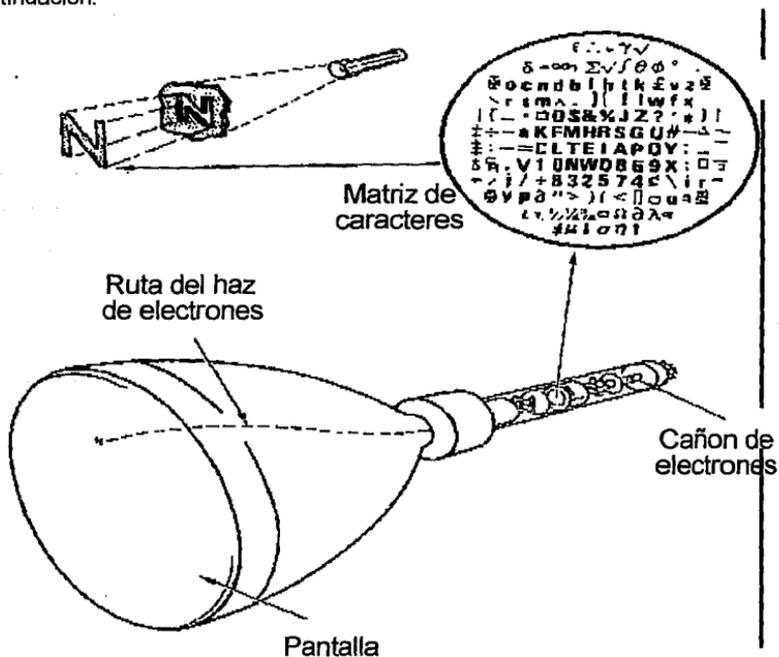
verse a simple vista, por lo que se tiene un aparato que permite amplificarlas para su consulta. La micrograficación significa "escritura pequeña", que es exactamente lo que ocurre cuando se genera un **microfilm**. Un microfilm es una película fotográfica de muy alta resolución que contiene la imagen muy reducida en tamaño comparada con la forma original. Existen dos formas de microfilmación disponibles para emplear la técnica de micrograficación: rollo de película y la microficha.

Un **rollo de película** no es más que un rollo continuo de material fotográfico tal y como lo conocemos, almacenado en un cartucho comúnmente de un tamaño de 16 mm. de ancho. **Microficha** o ficha (derivado del francés que significa "carta pequeña"), es una unidad de material fotográfico en los que las fotografías o imágenes están grabadas en un arreglo matricial, esto es en renglones y columnas. Las microfichas miden generalmente 105 mm. de ancho contra 148 mm. de largo. El estándar de reducción es de 24X, 42X y 48X, es decir, que los caracteres que se encuentran en la microficha son 24, 42 y 48 veces respectivamente más pequeños que los originales. Cada una de ellas contiene de 72 a 288 veces más información que una hoja de listado típico de una computadora.

El **COM (Computer Output Microfilm)** es un sistema de manipulación de la información basado en el proceso de grabar información generada en una computadora a un microfilm a altas velocidades. Este proceso permite el control de cantidades masivas de información en un espacio muy pequeño; fue inventado por DATAGRAPHICS hace **50 años**.

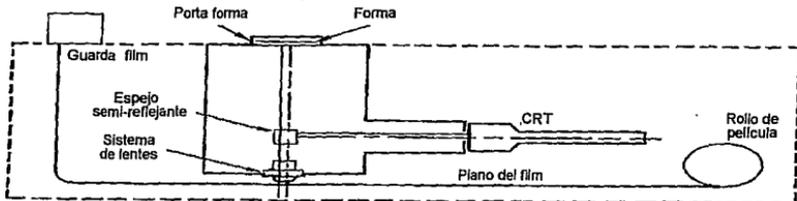
Producción del microfilm

La producción de un microfilm es muy similar al de una fotografía. La información es alimentada al COM, vía cinta magnética o disco flexible, incluso puede ser introducida *en línea*, es decir, desde una terminal de computadora. Entonces la información es transcrita al microfilm a través de un **Charactron™** el cual es un tipo de tubo de rayos catódicos (CRT). Este se muestra a continuación.



Un rayo de electrones es introducido y pasa por una matriz ubicada en el extremo del CRT, esta matriz tiene todos los caracteres que se pueden escribir, a

su vez son reflejados en la pantalla. La imagen del caracter es pasada a través de un espejo semi-reflectivo y se reduce su tamaño gracias a unos lentes y se enfoca hacia el rollo de película o film. Esto lo podemos ver en la siguiente figura:

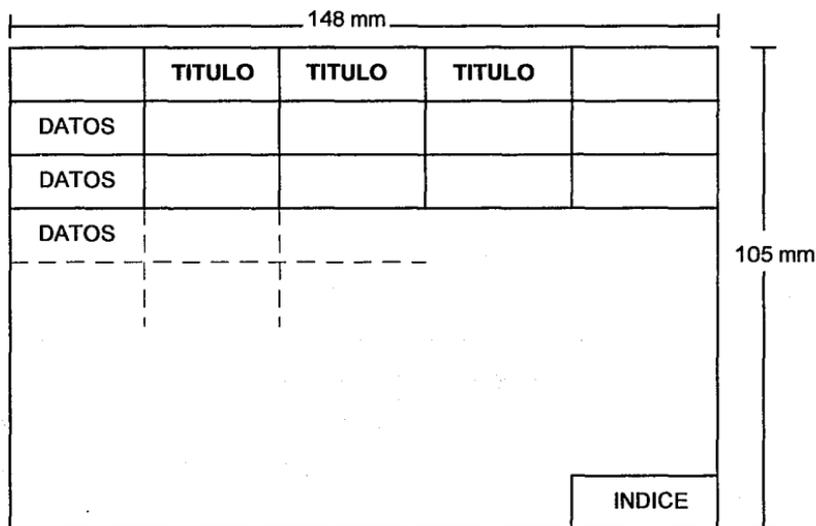


Después de que la hoja que contiene la información es expuesta en el CRT, se avanza tanto la hoja como la película para procesar la siguiente. Cada una de estas hojas de datos se conocen como **frames**. Una vez que todos los datos son procesados y transcritos a un microfilm, este es removido de la cámara y es procesado. El proceso que se sigue es exactamente el mismo que se sigue en un revelado normal de película. Una vez que es revelado, el microfilm está listo para ser cortado en microfichas (o dejada en su forma de rollo) y ser leído a través de un lector de microfichas, el cual no es más que un aparato que ilumina con un haz de luz la microficha y la imagen es reflejada a una pantalla.

La forma en que se tratan a los microfilms y a las microfichas y su forma de organización de la información es muy diferente, a continuación se describirá esto.

Formato de la microficha.

El formato que se le da a una microficha permite el alojamiento de la información así como una organización de la misma. Dentro de la microficha existen tres diferentes áreas, como se puede ver a continuación:



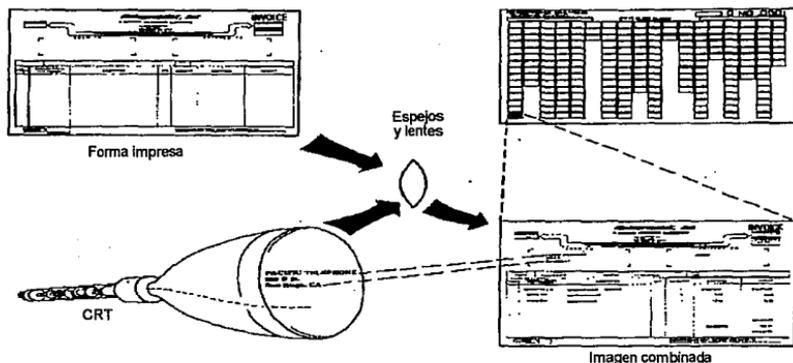
La primera y más importante área de la microfichas es la de datos, que contiene los **frames** de los datos de usuario. Cada **frame** contiene la información que generalmente es dada en un reporte impreso por computadora o forma, típicamente de 64 líneas de 132 caracteres. Estos **frames** están organizados en columnas y renglones.

La segunda área de la ficha es la de títulos, la cual se encuentra en la parte superior de la misma y ocupa uno o dos renglones. Estos dan una referencia de que guarda la microficha y son legibles a simple vista con objeto de identificación.

La tercera área es la de índice. Típicamente una microficha contiene dos cuadros de índice ubicados en la parte inferior derecha. El contenido de este índice varía, pero es una costumbre el poner la primera y última referencia de cada ficha para una rápida identificación y localización de datos. Asimismo se ofrece dentro del índice un código de renglones y columnas para saber la localización exacta de una ficha.

El formato de los rollos microfilmados no difieren mucho de las microfichas, de hecho contienen la misma información, pero en lugares diferentes. Comúnmente el ancho del film es de 16 mm. y la información grabada de izquierda a derecha es: títulos, área de datos, índices, otra área de títulos y un separador.

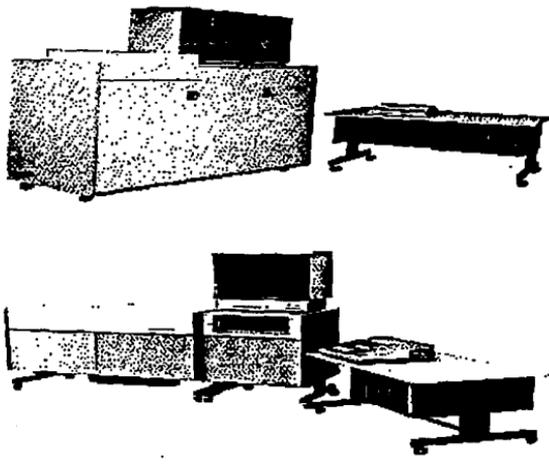
Ahora bien, en una microficha además de guardar datos aislados puede guardar formas impresas como pueden ser facturas, recibos y en nuestro caso **historias académicas**. Una **forma** es un film que contiene la imagen de una forma pre-impresa y que será superpuesta a la información de la microficha, el grabado de las formas se muestra en la siguiente figura:



Esta forma es insertada por el operador del sistema dentro del grabador. Para poder grabar la imagen de la forma junto con los datos de la misma, la forma es iluminada cada vez que se genera una ficha. Esto permite el uso de encabezados, logos, etc. dentro de la microficha.

Para terminar con la descripción de esta unidad de microfilmación veremos a gran detalle el proceso de grabación de una microficha.

La compañía **Datagraphix** tiene una gran diversidad de equipo **COM** que incluyen dos grabadores de microfilms controlador por dos computadoras, el **MiniAuto COM** y el **Sistema 4500**, ambos grabadores, estos se muestran en la siguiente figura:

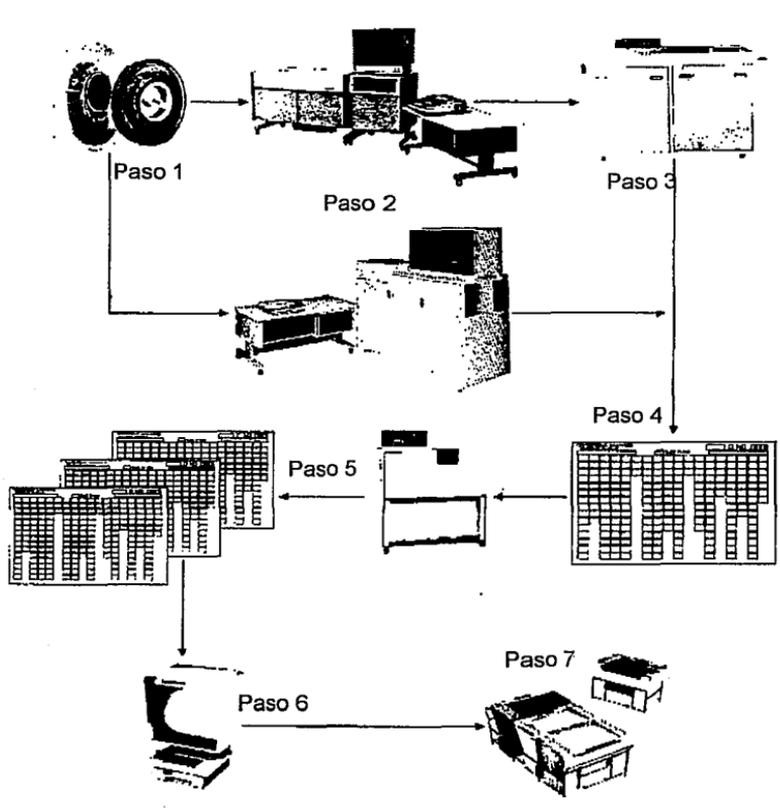


Estos dispositivos convierten los datos que se quieren grabar en microfilm (comúnmente archivos de impresión) en un flujo continuo de datos. El sistema MiniAuto COM produce microfichas, mientras que el System 4500 produce microfichas y microfilms. Estos sistemas son un completo laboratorio fotográfico unido con la electrónica necesaria para el manejo de la información, ya que en ellos se procesa la información a grabar, toman la foto y la revelan, todo esto en una sola operación. Ambos vienen equipados con una cámara especializada para la microfilmación, además de lentes para tal propósito, los cuales son móviles.

El control de estas unidades está regido por una minicomputadora. La comunicación a través de la unidad de grabación y el operador está dada por un panel luminoso, el cual despliega mensajes de su status, además de un teclado en donde se insertan las ordenes. Las unidades de cinta de las unidades son

capaces de leer diferentes formatos de cintas como pueden ser 800 bpi, 9 tracks, 1600 bpi, etc.

Ahora veamos la secuencia de pasos para la filmación, estos están representados a continuación:



Paso 1. Una cinta magnética que contiene la información que se quiere grabar es montada en la unidad del COM. Esta cinta contiene todos los caracteres que quieren ser impresos en la microficha. Este archivo es generalmente de impresión, es decir, contiene caracteres especiales que una impresora puede entender. El **mainframe A-12** es el que procesa toda la información de las historias académicas y genera estos archivos como resguardo.

Paso 2. El grabador COM, que contiene la minicomputadora, el tubo Charactron™, los lentes necesarios, cámara y sistemas relacionados, es conectado a una terminal electrónica. Una vez que la cinta fue montada, el operador carga el programa y subtablas necesarias para el correcto formateo de los datos, comienza el **job** (trabajo) y la producción comienza.

Para la producción de una microficha, la información de la cinta es desplegada en el Charactron™, un cuadro a la vez, es decir, la información necesaria que llena a la ficha. La información desplegada es reducida por un sistema de lentes y es expuesta al material fotográfico. El film avanza conforme la foto haya sido tomada y un sistema de transportación de film es el encargado de moverlo para la siguiente operación. Hasta este punto el film está listo para su proceso.

Paso 3. Se revela la foto tomada en el microfilm para hacerla visible. Este proceso es acompañado de todo el proceso que involucra el revelado de una foto normal, esto es el lavado, enjuague, secado, etc.

Paso 4. En el caso del sistema MiniAuto COM la microficha resultante está lista para ser leída. En el caso del System 4500 el film puede ser cortado en microfichas o dejado en forma de rollo.

Paso 5. Si se desean copias adicionales, estas se deben hacer en un dispositivo por separado, cuya única función es esa.

Paso 6. Una vez procesadas se procede a su lectura en el lector de microfichas. El lector amplifica la imagen y la proyecta a una pantalla donde es legible para el usuario.

Paso 7. La microficha es almacenada. Esta requiere de muy poco espacio de almacenamiento.

1.1.3.2. Formato de la información.

La información al terminar de ser procesada por el sistema A12 genera un Printer Backup Tape (Cinta de Respaldo de Impresión), para poder ser transportada y utilizada en la unidad de microfilmación; el formato general de la información que contiene la cinta es: código EBCDIC (nativo de la Serie A) y su densidad de grabación es de 1600 bpi's (bits per inch, bits por pulgada). Pero veamos un poco más a detalle este formato de la cinta.

Los datos en la cinta magnética están divididos en registros físicos. Estos registros físicos están divididos por áreas "blancas" (sin datos) llamadas **Inter Record Gaps** (IRG's, vacío entre registro). Cuando cada registro físico consiste de un registro lógico o una colección de datos relacionados, los datos son considerados **unblocked** (desbloqueados). Los datos son considerados bloqueados cuando registros lógicos son combinados y escritos como un solo registro físico. Cada registro lógico consiste de caracteres conocidos como bytes,

un campo de caracter de control de salto de hoja es el que especifica la posición de una línea de impresión en la hoja y otros bytes indican el tamaño del registro, los encabezados del bloque, número de serie del bloque, etc., que no son parte de los datos que finalmente aparecen en el film. Cuando el registro es analizado por la impresora ellos son adaptados en cadenas de impresión potenciales incluyendo los datos imprimibles y un campo con un caracter de control de salto de hoja.

La colección de registros lógicos con características comunes tal como los registros de la nomina de los empleados en un departamento, es un archivo. El fin de los datos en un archivo específico es hecho por una marca de archivo que es detectada por el hardware o software. El uso de las marcas de archivo permite que múltiples archivos sean grabados en un solo carrete de cinta.

Los registros lógicos pueden estar en formato de "fixed length" o "variable length" (tamaño fijo o variable). En un archivo compuesto de registros de tamaño fijo, cada registro lógico tiene el mismo tamaño. Por el contrario en un archivo compuesto de registros de tamaño variable, cada registro lógico puede tener diferente tamaño.

Los archivos de datos pueden contener etiquetas de encabezado o pie de archivo. Las cintas magnéticas están grabadas en 7 o 9 tracks dependiendo del tamaño de la cinta.

1.1.4 Equipo SparcStation SUN y el sistema operativo UNIX

Desde finales de 1992 se comenzó a realizar un estudio de los sistemas de administración escolar en la UNAM, sobre la evolución que han mostrado y sus carencias actuales. Como consecuencia de dicho estudio la Universidad a través de su Dirección General de Servicios de Cómputo Académico, ha apoyado un plan de modernización en la administración escolar que consiste en el mejoramiento de los procedimientos administrativos con el apoyo de los beneficios que ofrece la tecnología de punta.

Uno de estos cambios consistió en el diseño de una nueva arquitectura de cómputo. Como se comentó en el capítulo de introducción, el mainframe de la Serie A de Unisys modelo A12 es el que realiza varios procesos para la administración escolar desde el año de 1987. Análisis previos han revelado que el mantener a las computadoras Unisys como única plataforma para el desarrollo y producción de los sistemas de administración no es conveniente debido a varios factores, entre ellos tenemos, que no es un sistema abierto y todos los altos costos que se derivan de su actualización, mantenimiento y apertura.

El cambio deseado es el tener un sistema abierto que permita en un momento dado la descentralización de ciertos procesos administrativos y la implementación de un esquema cliente/servidor en varias aplicaciones administrativas. Debido a sus características de apertura y portabilidad el sistema operativo UNIX fue elegido como solución.

La primera alternativa de solución fue el tener bajo un esquema cliente/servidor, a los equipos Unisys como servidores y a una computadora SUN

como front-end. Este front-end es temporal pues forma parte de la primera etapa de la migración que se tiene pensada. El equipo elegido fue el SparcStation 10 de SUN.

Se tomó la opción de utilizar equipos de SUN Microsystems debido a que sus procesadores son de aplicación general y que han nacido y se mantienen bajo un ambiente UNIX. Además sus equipos no requieren mantenimiento muy costoso y actualmente, cuenta ya con una buena base instalada en la Universidad por lo que existen acuerdos ventajosos para la adquisición de equipos.

El equipo SUN esta basado en la arquitectura RISC por lo que antes de comenzar con la descripción de las características del equipo, daremos una descripción de esta.

Principios de la arquitectura RISC

La familia de microprocesadores conocida como **CISC (Complex Instruction Set Computer)** se caracteriza por su amplio conjunto de instrucciones, múltiples modos de direccionamiento, formatos de instrucciones y diferentes tamaños. Sus unidades de control son muy complejas, ya que deben ser capaces de distinguir entre un gran número de códigos, formatos y modos de direccionamientos. En contraparte a esta arquitectura surgió en la década de los 80's la llamada **RISC (Reduced Instruction Set Computer)**. La filosofía de la arquitectura RISC evolucionó a partir de un proyecto en la Universidad de Stanford y busca el incremento de la velocidad de procesamiento a partir de una simplificación en el diseño del CPU. Uno de los principios fundamentales de diseño es el manejo **continuo** y **uniforme** de todas las instrucciones, donde los ciclos de **fetch** y

ejecución tomaran el mismo tiempo para cualquier instrucción, preferentemente en un solo ciclo de máquina.

Para lograr este flujo continuo de operaciones es necesario establecer una serie de condiciones, que explicaremos a continuación:

- Formato fijo de instrucciones:

El tener un tamaño fijo en las instrucciones del microprocesador, permite una mejor implementación del **pipelining** o entubamiento de operaciones. Consideremos que tenemos un pipeline de dos etapas: 1) fetch (F) y, 2) ejecución (E); si tomamos un modelo sencillo con operaciones de tamaño fijo y que se ejecutan en solo ciclo de máquina tendríamos el siguiente plan de instrucciones contra ciclos de CPU:

CICLO /	1	2	3	4
INSTRUCCIÓN				
i	F	E		
i + 1		F	E	
i + 2			F	E

Ahora mostramos el mismo esquema pero utilizando un conjunto de instrucciones con diferente longitud y por lo tanto con diferentes ciclos de ejecución:

CICLO /	1	2	3	4	5	6	7
INSTRUCCIÓN							
i	F	E	E	E			
i + 1		F			E		
i + 2			F	F		E	E

Como se puede observar la diferencia de tamaño de las instrucciones provocan que algunas instrucciones tengan que esperar algunos ciclos dentro del entubamiento.

El tamaño estándar de las instrucciones debe de ser igual a la longitud de la palabra de la computadora, la cual es generalmente igual al número de líneas en el bus de datos. Esto se debe a la facilidad de implementar que las instrucciones se ejecuten en un solo ciclo de máquina.

- Menor número de instrucciones:

Se realizaron estudios que demostraron que la gran mayoría de compiladores usaban sólo el 30% del conjunto de instrucciones de computadoras CISC y que de ese porcentaje, solo el 20% se utilizaba el 80% del tiempo. La tendencia de los procesadores RISC es implementar solamente ese 30% de instrucciones permitiendo que el chip sea más pequeño, barato y rápido. Para una arquitectura RISC el numero de instrucciones debe ser menor o igual a 128.

- Ejecución en un sólo ciclo de máquina.

Al menos el 75% de instrucciones deben ser lo suficientemente sencillas para ser ejecutadas en un sólo ciclo de máquina. Operaciones complejas como carga de datos, almacenamiento en registros, multiplicaciones y divisiones sí requieren múltiples ciclos.

- Registros "muertos"

La gran mayoría de RISCs cuentan con un registro muerto (llamado g0 en los Sparc's) que contiene el valor 0 cuando es leído y descarta cualquier valor escrito en él. Usando ese registro como un operador fuente provee una manera

conveniente de convertir una instrucción con dos fuentes en una instrucción sencilla con una fuente simple simplificando el conjunto de instrucciones.

- **Arquitectura de Carga/Almacenamiento**

La manipulación de la memoria externa solo se debe realizar a través de las operaciones de **load** y **store**; todas las demás instrucciones afectan sólo los registros internos. Este tipo de implementación facilita el diseño de sistemas multitarea y multiprocesamiento.

- **Modos de direccionamiento limitado:**

Hablando estrictamente, un procesador RISC realiza sus cálculos de direcciones en registros y tiene un sólo modo de direccionamiento: indirecto a través de un registro. Sin embargo, las especificaciones de RISC ponen como máximo 4 modos de direccionamiento por lo que la gran mayoría de implementaciones utilizan un modo más. Inclusive el Sparc utiliza un modo más el cual forma una dirección de memoria al sumar el contenido de dos registros internos.

- **Unidad de control implementada en hardware (hardwired):**

La gran mayoría de sistemas CISC utilizan la microprogramación debido a la flexibilidad que ofrece y a que una instrucción CISC al realizar muchas funciones, permite que el código utilice menos espacio en memoria y ancho de banda del bus. La filosofía RISC se defiende diciendo que esto es un artificio utilizado en los primeros días de la computación cuando la RAM era más cara. La filosofía RISC propone que la relación costo/beneficio puede ser incrementada al eliminar el microcódigo e implementar en hardware la decodificación de instrucciones y la secuenciación.

- Mayor número de registros internos.

Se ampliaron los registros internos para almacenamiento intermedio lo que minimiza las operaciones de load /store en memoria. A nivel industrial se ha especificado que el mínimo número de registros sea de 32.

Arquitectura Sparc

En 1987, SUN Microsystems lanzó su primera computadora basada en una nueva arquitectura de CPU denominada **SPARC (Scalable Processor Architecture)**. A diferencia de los microprocesadores existentes, este nuevo CPU tan solo constaba de 20,000 compuertas. Además se presentó con el esquema de arquitectura abierta.

Desde ese entonces, SPARC ha llegado a ser más que una nueva arquitectura. Es una familia de estándares, un conjunto de circuitos integrados de al menos un docena de vendedores, un vehículo para la estandarización y una base en la cual docenas de fabricantes están implementando sus nuevas workstations.

La arquitectura SUN SPARC no esta relacionada específicamente a ninguna arquitectura de hardware en especial, se han concedido licencias a diferentes fabricantes, por lo que podemos encontrar en el mercado diferentes microprocesadores, todos ellos bajo el esquema SPARC. El concepto de escalamiento para los creadores de SPARC, es el amplio espectro de sus posibles implementaciones en cuanto al precio y desempeño, el cual va desde una microcomputadora hasta una supercomputadora. El escalamiento también

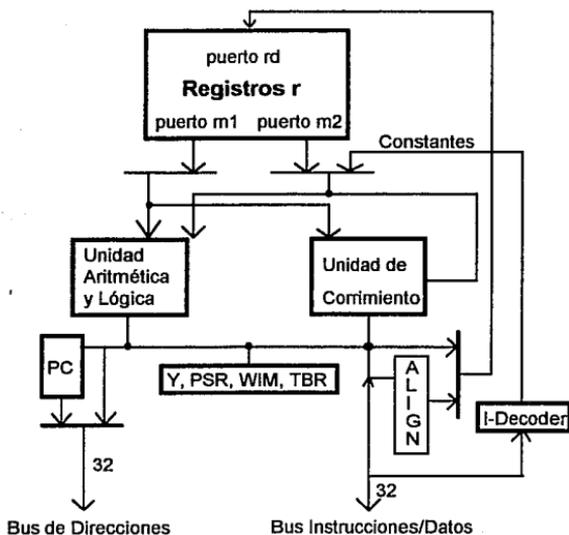
puede ser interpretado como el número de CPU que pueden ser utilizados en varios productos, pero todos ellos utilizando la arquitectura SPARC.

SPARC en lo que respecta al diseño sigue la filosofía de Berkeley, en la importancia que da a los registros del CPU y a la similar implementación de sus registros de ventanas.

El procesador SPARC esta subdividido en dos unidades básicas:

- **Integer Unit (IU) :** la unidad entera realiza el procesamiento básico y las operaciones aritméticas enteras.
- **Floating-Point Unit (FPU):** la unidad de punto flotante realiza las operaciones de punto flotante en forma concurrente a la IU.

En la siguiente figura se muestra un diagrama de bloques de la IU. Existen dos buses internos de 32 bits cada uno: el de direcciones y el de instrucciones/datos.



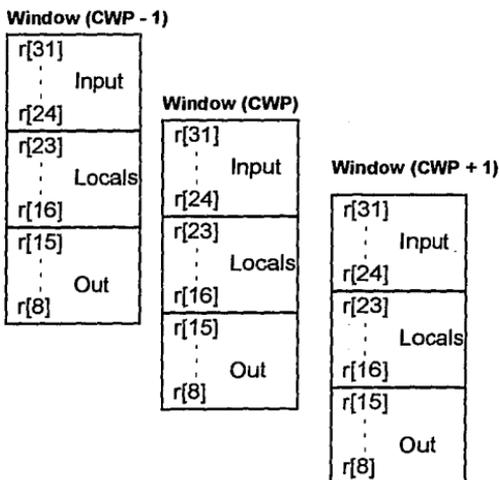
Los registros de control y status de la IU son los siguientes:

1. El **Program Counter (PC)** que contiene la dirección de la instrucción actualmente ejecutada por la IU.
2. El **siguiente PC (nPC)** que contiene la dirección de la siguiente instrucción a ser ejecutada.
3. El **Processor State Register (PSR)** que contiene el status del procesador, incluyendo la condición de las banderas de código y los 5 bits del apuntador actual a la ventana.
4. **Window Invalid Mask (WIM)** compuesta de 32 bits w_i ($i = 0, 1, 2, \dots, 31$) cada uno corresponde a una ventana. Si $w_i = 1$, la ventana i es considerada invalida y ocurre una condición trap.
5. El **Trap Base Register (TBR)** que sirve como apuntador a un trap handler.

También incluye el **type trap encoding**.

6. El **registro Y** el cual es usado para crear los productos de 64 bits en instrucciones con múltiples pasos.

Un concepto importante en la arquitectura SPARC es **register windowing**. En cualquier momento, un programa corriendo tiene acceso a 32 registros de procesamiento de 32 bits cada uno.

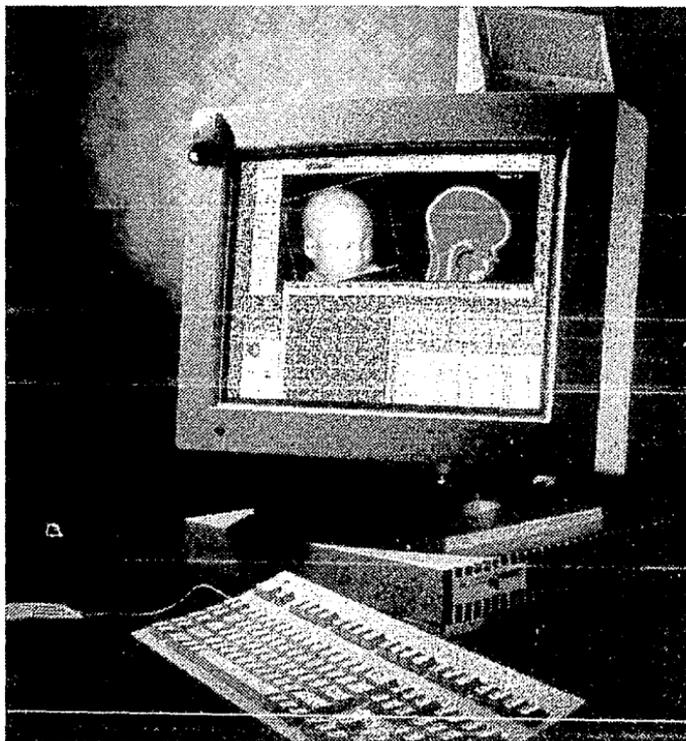


Los primeros 8 registros de la ventana se denominan los registros de entrada (i0 - i7). Cuando una función, o programa principal es llamado, estos registros pueden contener argumentos para que esa función los use. Los siguientes 8 registros son llamados los registros locales; los cuales pueden ser utilizados para cualquier propósito mientras la función se ejecute. Los últimos 8 registros son los

registros de salida; la función los usa para pasar argumentos a las funciones que el llame.

Cuando una función llama a otra, la llamada puede escoger ejecutar una instrucción **SAVE**. Esta instrucción decrementa un contador interno, llamado **CWP** (Current Workspace Pointer), corriendo el registro de ventana hacia abajo. Los registros de salida de la función que llama se convierten en los registros de entrada de la función llamada, y la llamada obtiene un nuevo conjunto de registros locales y de salida para su propio uso. Solamente cambia un apuntador, los registros no necesitan ser salvados en el stack.

- **Características del SparcStation 10**



El SparcStation 10 está diseñado para correr aplicaciones con buen nivel transaccional; posee además la capacidad de crecer o actualizarse fácilmente, lo que protege la inversión realizada y alarga la vida del sistema. La configuración básica para el modelo 40 es la siguiente:

Procesador: SuperSparc

Velocidad del reloj:	40 MHz
MIPS:	96.2
Memoria principal:	32 MB
Memoria caché:	1 MB
Disco magnético:	424 MB

El SparcStation 10 soporta hasta 512 megabytes de memoria principal si utiliza los módulos de memoria llamados DSIMMs de 64 MB o hasta 128 si utiliza los de 16 MB. DSIMM significa "Dynamic random access memory Single In-line Memory Module". La memoria principal tiene un bus de 128 bits y tiene 16 bits para la corrección y detección de errores; estas características hacen que un mayor número de datos viajen entre el CPU y la memoria en un ciclo.

Los módulos MBus conectan a la memoria principal y al procesador(es) SuperSparc. Este módulo contiene los siguientes circuitos centrales de procesamiento:

- IU (Unidad entera).
- FPU (unidad de punto flotante).
- Controlador de memoria caché.

Estos módulos ofrecen la ventaja de permitir la actualización del equipo (lo que comúnmente se denomina *upgrade*). Esta actualización va desde la instalación de otro procesador o la instalación de un nuevo modelo de módulo MBus. Además elimina la necesidad de utilizar las tarjetas SBus para entrada/salida y para la memoria principal.

Las tarjetas SBus permiten la conexión de ciertos dispositivos como impresoras y monitores. Existe una gran variedad, incluyendo tarjetas Ethernet, adaptadores SCSI para dispositivos SCSI, la tarjeta SparcPrinter para impresora y tarjetas para video.

La SparcStation 10 puede tener hasta 2 discos duros internos con capacidad de 424 MB cada uno. La unidad de disco es de 3.5 pulgadas y es compatible con el formato Unix y DOS. Utiliza discos de los tipos HD, 2HD y 2DD y soporta densidades de grabación de 720 KB, 1.2 MB Y 1.44 MB.

Junto con el equipo se adquirió un dispositivo externo de almacenamiento que se denomina **Mass Storage**. Este módulo contiene 2 discos de 2.3 GB cada uno, una unidad de cinta de 8 mm con una capacidad de 5 GB y una unidad de CD-ROM.

EL SISTEMA OPERATIVO UNIX

En el punto anterior discutimos algunos aspectos del hardware empleado, sin embargo, no podríamos dejar de mencionar una parte indispensable en el funcionamiento del equipo: **el sistema operativo**.

De más es decir que el más sofisticado equipo de cómputo con un gran poder de procesamiento, discos de alta capacidad de almacenamiento y rapidez de acceso, además de gran cantidad de memoria serían inútiles sino se tuviera una adecuada administración y control de todos ellos. Esta es precisamente una de las funciones del sistema operativo quien además se encarga de proveer al

usuario una serie de herramientas que le permitan llevar a cabo actividades comunes tal y como el copiado de archivos.

El origen de UNIX se remonta a mediados de 1960 cuando las compañías American Telephone & Telegraph (AT & T), Honeywell, General Electric y el Instituto de Tecnología de Massachusetts trabajaron en un sistema de información denominado Multics (*Multiplexed Information and Computing Service*). El objetivo era tener un servicio de cómputo las 24 horas del día y los 365 días del año diseñado de manera modular para que fallas en una parte no afectarían al resto del sistema.

Sin embargo en 1969 el proyecto fue abandonado debido a que no se terminó en el tiempo establecido dada su complejidad. A pesar de ello, en el mismo año uno de los desarrolladores de Multics en AT & T decidió rescatar parte del proyecto y realizar un diseño más sencillo en una PDP-7. Se le unieron en el desarrollo Ken Thompson y Brian Kernighan, este último fue quien sugirió el nombre de Unix, un juego de palabras entre Multics y el objetivo del nuevo proyecto: mientras Multics trató de hacer muchas cosas, Unix solo iba a hacer una: correr programas.

El sistema está basado en programas compactos, llamados **herramientas**, las cuales realizan una sola función. Al colocarlas juntas, los programadores pueden realizar cosas más complicadas. En 1973 Thompson reescribió el sistema en el nuevo lenguaje por él inventado: **el lenguaje C**. Esta nueva característica hizo al **sistema portable**, es decir, un sistema operativo que es capaz de funcionar en cualquier plataforma con sólo realizar modificaciones en las funciones primitivas que están íntimamente relacionadas con el hardware de la máquina.

Muy pronto se hizo sumamente popular en varias compañías y universidades, siendo una de ellas la Universidad de Berkeley, que hizo una serie de aportaciones importantes al sistema, surgiendo así en 1977 Berkeley Software Distribution. Una de las más importantes fue el desarrollo de software de red que permitió de manera fácil la conexión de computadoras Unix a LANs.

Hoy en día se estima que en los E.U.A. 1.5 millones de computadoras utilizan el sistema operativo Unix. Las diferentes versiones corren en casi todo tipo de computadoras existentes, desde una PC hasta una supercomputadora y con la ventaja que a nivel usuario podemos asegurar una gran compatibilidad. Además, debido a que es fácilmente adaptable a nuevos tipos de computadoras, es el sistema operativo que muchos de los procesadores de alto rendimiento escogen.

CONCEPTOS DEL SISTEMA

El sistema operativo lo podemos dividir en tres partes:

El Kernel: es el corazón de Unix. Es un gran programa que se carga en la computadora al ser inicializada. Controla todas las I/O, el multiprocesamiento y la administración de los recursos de procesador y memoria.

Utilerías estándar: programas ejecutados por el usuario y el sistema. Algunos son pequeños y específicos otros son más complicados y llevan a cabo múltiples funciones, como el intérprete de comandos shell.

Archivos de bases de datos: utilizados por el sistema para guardar y llevar control de actividades del sistema.

La razón de la creciente popularidad de Unix reside en la combinación que logra entre facilidad de uso y eficiencia, además de la gran cantidad de ayudas y utilerías para la programación que tiene incluidas. Con Unix es sencillo lograr comunicación y sincronización entre procesos, lo que requiere de programación dedicada y exclusiva en los lenguajes de control de otros sistemas operativos.

Entre las características del sistema operativo Unix están las siguientes:

- Es un sistema operativo multiusuario, con capacidad de simular multiprocesamiento y procesamiento no interactivo.
- Esta escrito en un lenguaje de alto nivel: C
- Dispone de un lenguaje de control programable, llamado Shell.
- Ofrece facilidades para la creación de programas y sistemas, y un ambiente muy propio para las tareas de diseño de software.
- Emplea manejo dinámico de memoria (por intercambio o por paginación).
- Tiene capacidad de interconexión de procesos.
- Permite comunicación entre procesos.
- Emplea un sistema jerárquico de archivos, con facilidades de protección de archivos, cuentas y procesos.
- Usa un manejo consistente de archivos de diversos tipos.
- Tiene facilidades para redireccionamiento de entradas/salidas.
- Incluye mas de un centenar de subsistemas y varios lenguajes de programación.
- Garantiza un alto grado de portabilidad.

- Puede adaptarse a requerimientos de configuración particulares; esto es, puede reconfigurarse en cada instalación.
- Es un buen sistema operativo para programadores y actualmente también para la ejecución de aplicaciones de todo tipo.

El sistema se basa en un núcleo conocido como kernel que reside permanentemente en la memoria y que atiende todas las llamadas del sistema, administra el acceso a los archivos y el inicio o suspensión de las tareas de los usuarios.

Este núcleo es un programa de aproximadamente 10 000 renglones, escrito casi en su totalidad en C con excepción de una parte del manejo de interrupciones, que esta escrita en el lenguaje ensamblador del procesador en el que opera.

Las funciones del núcleo son permitir la existencia de un ambiente en el que sea posible atender a varios usuarios y múltiples tareas en forma concurrente, repartiendo al procesador entre todos ellos, e intentando mantener en grado optimo la atención individual.

El kernel opera como asignador de recursos para cualquier proceso que necesite hacer uso de las facilidades de computo. Tiene las siguientes funciones:

- Creación de procesos, asignación de tiempos de atención y sincronización.
- Asignación de la atención del procesador a los procesos que lo requieren
- Administración de espacio en el sistema de archivos, que incluye:
- acceso, protección y administración de usuarios,

- comunicación entre usuarios y entre procesos, y
- manipulación de E/S y administración de periféricos.
- Supervisión de la transmisión de datos entre la memoria principal y los periféricos.

1.2 Red de Servicios Integrados de la UNAM.

Actualmente, si nosotros hacemos una retrospectiva de todos los avances que el hombre ha tenido a lo largo del tiempo, nos percataremos que cada siglo es identificado plenamente por uno en particular, así por ejemplo en el siglo XVIII los avances en las máquinas y engranes sentaron las bases para que en el siglo XIX con la invención de la máquina de vapor se pudiera dar la revolución industrial.

Durante el siglo XX la tecnología de la información ha sido la dominante. Además debemos considerar lo siguiente: en siglos pasados las invenciones tomaban mucho tiempo para su desarrollo, actualmente en menos de 40 años, se han dado pasos gigantescos en lo que se refiere a la tecnología, sobretodo con el advenimiento de la herramienta característica de este siglo; **la computadora**, entre otros muchos otros avances como son las comunicaciones, los satélites, radio, televisión, etc.

Conforme se ha avanzado se ha dado una rápida convergencia de estos avances, es decir, se ha tratado de englobar la captura, transporte, almacenamiento y procesamiento de la información. Actualmente, gracias a los avances de las comunicaciones, es factible que, una empresa que tiene varias

sucursales disgregadas a lo largo de una geografía, pueda comunicarse con ellas para obtener sus estados actuales con simplemente oprimir algunas teclas.

Aunque la industria de las computadoras es una industria todavía muy joven, ésta ha demostrado un progreso espectacular, sino es que sorprendente, en muy poco tiempo. Durante las primeras décadas de su introducción, las computadoras estaban muy centralizadas, usualmente en un cuarto muy grande, con sistemas muy especiales para poder mantenerla trabajando; se encontraban de una manera aislada y la gente se maravillaba de las cosas que podía hacer ese gigante electrónico. Solamente grandes instituciones o universidades se podían dar el lujo de adquirir algunas de esas maravillas. Sin embargo al pasar el tiempo dicha concepción cambió radicalmente; se comenzaron a construir computadoras tan pequeñas como un cuaderno y de una potencia de cálculo es algunas ocasiones superiores a los grandes ordenadores de hace 30 años. Para aquel entonces dichos avances rayaban dentro del terreno de la ciencia ficción.

Uno de los avances más significativos que tuvieron las computadoras fue su unión con las comunicaciones. Pongamos un ejemplo: el concepto de que el usuario de un "centro de cálculo" donde residía un gran computador, debía llevar su trabajo a procesar, ha quedado ya obsoleto. Este modelo tiene dos serias deficiencias, la primera es que el ordenador central debe hacer todo el trabajo, y el segundo es que el usuario debe llevar su trabajo a la computadora, en vez de que esta vaya a su lugar de trabajo. Este ejemplo ha sido rápidamente modificado por el concepto de un número grande de computadoras pequeñas interconectadas entre sí; a esta clase de sistemas se le denomina **redes de computadoras**.

La conexión de estas computadoras no necesariamente debe de ser a través de un hilo de cobre, este puede ser a través de fibra óptica, microondas o satélites de comunicación por mencionar los más comunes.

El hecho de referirse a una serie de computadoras pequeñas conectadas entre si no debemos confundir con lo que se denomina un **sistema distribuido**. Estas computadoras deben de considerarse como *autónomas* es decir, que una computadora pueda arrancar por sí misma. La diferencia que existe entre un sistema distribuido y una red es que en un sistema distribuido la existencia de múltiples computadoras autónomas es transparente para el usuario, este simplemente teclea un comando y la computadora ejecuta su aplicación, el hecho de seleccionar el mejor procesador, el transportar y poner los archivos a la entrada del mismo y poner los resultados en un lugar apropiado, depende exclusivamente del sistema operativo de la máquina. El usuario ve a su computadora como una máquina virtual que solamente lo esta atendiendo a él en ese instante; todas las demás actividades relacionadas con el sistema se realizan de forma automática.

Con una red de computadoras no sucede así, ya que el usuario debe *explícitamente* entrar a una máquina, debe enviar sus trabajos de forma remota, debe mover sus archivos y generalmente administrar personalmente la red.

Podemos decir entonces que un sistema distribuido es un caso especial de una red de computadoras y que por lo tanto la diferencia existe en el software que se emplea más que en el hardware.

Antes de entrar en materia consideramos que es necesario aclarar algunos puntos acerca de por que son tan importantes las redes de computadoras actualmente, además de saber en que nos pueden servir en un momento dado.

En la actualidad son muchas las organizaciones que cuentan con una red de computadoras y en la mayoría de los casos en donde se tienen estas máquinas se encuentran muy alejadas unas de otras. Una vez interconectadas éstas pueden seguir funcionando de manera independiente, sin embargo, en algún momento cuando se necesite información de alguna de ellas en específico, puede compartir su información con las demás, podemos extraer y correlacionar información referente a toda la organización.

Como primera conclusión podemos decir que en una red, su principal objetivo es el de **compartir recursos** ya que se debe tener datos, programas, aplicaciones y equipos disponibles para cualquiera en cualquier lugar, sin importar la clase de usuario o en donde se haya solicitado. Es decir, si un usuario se encuentra a 1000 km de distancia esto no debe de ser problema para acceder datos como si lo hiciera localmente.

Otro objetivo es el de una **alta fiabilidad** de los datos, al contar con fuentes alternas de suministro. Por ejemplo duplicación de información en máquinas para que en dado caso que alguna de ellas dejara de funcionar por alguna falla de hardware los datos se encuentren disponibles. Además de contar con diversos procesadores para que otras computadoras puedan hacerse cargo si existiese alguna falla. En organizaciones como las bancarias, ejército y otras es de vital importancia la capacidad de que los sistemas siempre estén funcionando a pesar de que exista alguna falla de hardware.

Otro de los objetivos es el económico. Las computadoras pequeñas tienen un mayor rendimiento en este aspecto que las grandes; no podemos negar el hecho del poder de procesamiento de un equipo grande, sin embargo el costo por ello es muy elevado en relación a tener muchas máquinas pequeñas. Actualmente la tendencia es tener una máquina de gran almacenamiento comúnmente llamada **servidor** y máquinas pequeñas interconectadas entre si con un aceptable poder de procesamiento.

Este objetivo conduce al concepto de varias máquinas interconectadas en un solo edificio, a este tipo de red se le conoce como **LAN** o *red de área local*, si nos referimos a un dominio más extenso podemos denominarla **WAN** o *red de área extendida*.

Comúnmente se cree que la implantación de una red tiene que verse relacionada con cuestiones tecnológicas, esto es falso. Una red de computadoras puede proporcionar un poderoso medio de comunicación entre personas alejadas entre sí.

Con el empleo de una red es relativamente fácil para dos o más personas, que viven en lugares distantes escribir, por ejemplo, un informe juntos. Por ejemplo si alguien realiza un cambio, las demás personas si se encuentran en línea pueden ver la modificación de manera inmediata. Este objetivo sobrepasa los objetivos técnicos como puede ser la fiabilidad por el factor de comunicación entre personas.

Queda claro entonces que otros de los objetivos, y se puede decir que el más importante es el de proporcionar un **medio de comunicación**, mucho más

versátil que cualquier otro, ya que podemos compartir datos, información, imágenes e incluso sonidos de una manera instantánea sin la necesidad de esperar tanto tiempo para recibir dicha información.

Respondiendo a la apremiante necesidad de modernizar las comunicaciones en la Universidad, a finales de 1989 se estableció un ambicioso proyecto para renovar completamente la infraestructura y los sistemas de telecomunicaciones.

Para este proyecto, que constituye la parte fundamental del programa institucional en informática, en la Dirección General de Servicios de Cómputo Académico se creó la Dirección de Telecomunicaciones Digitales cuyo objetivo sería la creación de la Red Integral de Telecomunicaciones de la UNAM.

Esta red debería ser capaz de transmitir indistintamente voz, datos e imágenes entre las dependencias universitarias independientemente de su ubicación geográfica.

Tres años después, a finales de 1992, la Red Integral de Telecomunicaciones contaba ya con 31 nodos de cómputo y telecomunicaciones enlazados entre sí a través de fibra óptica, vía satélite o vía microondas. En la parte de transmisión de voz, la Red contaba ya con capacidad instalada para 13,000 servicios digitales alimentados por 2,400 troncales digitales conectadas vía fibra óptica con las centrales telefónicas públicas. En la parte de transmisión de datos e imágenes, se tenía ya infraestructura instalada para la conexión de más de 110 redes locales de cómputo en 8 regiones del país.

Actualmente, las principales instalaciones de la Universidad están integradas a la Red. Esto significa que, a nivel bachillerato, licenciatura, posgrado e investigación, alrededor del 90% de sus miembros se encuentran en instalaciones cubiertas por la Red, en varias regiones del país desde Ensenada hasta Cancún.

Es importante mencionar que la Red Integral de Telecomunicaciones es completamente privada y propiedad de la UNAM y es operada en su totalidad por personal de la Dirección de Telecomunicaciones Digitales.

Para la comunicación entre las dependencias universitarias, la Red de Telecomunicaciones emplea los más modernos medios y sistemas de comunicación aprovechando las características principales de cada uno, en función de la capacidad de transmisión requerida en cada caso y de las características geográficas y físicas de cada dependencia.

Para la comunicación entre diferentes campus en regiones alejadas del país se utilizan **enlaces vía satélite** cuya capacidad de transmisión es razonable y permite la comunicación en lugares donde, por su ubicación geográfica, no existen otros medios de comunicación.

Para la comunicación entre diferentes campus dentro de la misma región se utilizan **enlaces vía microondas** punto a punto cuya capacidad de transmisión es alta y cuya instalación, operación y mantenimiento es relativamente sencilla y mucho más económica que el establecimiento de medios de comunicación por la vía pública.

Para la comunicación entre edificios dentro de cada campus se utiliza **fibra óptica** cuya capacidad de transmisión es verdaderamente asombrosa (equivalente a más de 10,000 llamadas telefónicas simultáneas por cada fibra). Adicionalmente, la fibra óptica es inmune a interferencias electromagnéticas y a descargas atmosféricas, no requiere de mantenimiento y en caso de ruptura se puede reemplazar en cuestión de horas.

Para la comunicación entre equipos y sistemas dentro de un mismo edificio se utilizan principalmente las tecnologías de **par trenzado, cable coaxial delgado y grueso** por su bajo costo y facilidad de instalación y mantenimiento. Este último se utiliza principalmente en las verticales de los edificios. Adicionalmente, ya se tienen conectados algunos equipos y sistemas con fibra óptica hasta el escritorio para aplicaciones que requieren grandes capacidades de transmisión; sin embargo, esta tecnología no se ha generalizado dentro de los edificios debido al alto costo de los equipos terminadores.

En lo que toca a los protocolos de comunicación, la Red Integral de Telecomunicaciones es completamente ruteada y se han seleccionado aquellos protocolos que se han convertido en estándares internacionales y que, previsiblemente, seguirán siendo vigentes por mucho tiempo (en el entendido que, en tecnología, la frase "mucho tiempo" es verdaderamente indefinible).

Para la transmisión de voz y datos de baja velocidad, el protocolo utilizado es el Sistema de Señalización #7 (SS7). Este protocolo es uno de los pocos estándares en los que se ha llegado a un acuerdo internacional para el establecimiento de las ya contempladas Redes Digitales de Servicios Integrados (ISDN).

Para la transmisión de imágenes y de datos de alta velocidad se utiliza el protocolo TCP/IP sobre las redes Ethernet locales y sobre el "backbone" FDDI. Otros protocolos utilizados, aunque en mucho menor medida, son IPX y BNA.

El diseño, la administración y la operación de la Red se realiza bajo la supervisión de personal altamente calificado que se encarga de:

- Monitoreo permanente del funcionamiento de la Red tanto a nivel físico como lógico.
- Planteamiento de acciones para evitar la saturación o fallas en la Red.
- Diagnóstico y seguimiento de solución de fallas.
- Implantación de tarifas y costeo de tráfico.
- Asignación de direcciones y números telefónicos y programación de facilidades.
- Coordinación para la integración de *hosts* (nuevas computadoras) a la Red.
- Configuración y programación de los conmutadores y ruteadores.
- Implementación de esquemas de seguridad en la Red.
- Instalación de nuevos equipos y sistemas.
- Información a los usuarios.
- Capacitación, etc.

Es importante indicar que la Red tiene un crecimiento, en términos de tráfico, equipos, enlaces, sistemas y usuarios, mayor a un 15% mensual.

Si bien la Red Integral de Telecomunicaciones de la UNAM se ha convertido en una herramienta invaluable para la comunicación de la comunidad

universitaria consigo misma, las conexiones de la Red con otras redes nacionales e internacionales han permitido el intercambio y la cooperación académica entre comunidades, enriqueciendo infinitamente las posibilidades de acceso a la actividad y a la información mundial.

Conscientes de lo que ello representaba, desde un principio se planteó como objetivo indispensable la conexión de la Red Integral de Telecomunicaciones a otras redes académicas de reconocido prestigio internacional. La más importante de ellas es la red **Internet** que es la red académica y comercial **más grande del mundo**.

La red Internet es, en realidad, una **red de redes** que ha debido su gran éxito a la conexión a ella de todas las instituciones académicas importantes a nivel mundial; a través de la Red y con la misma facilidad se puede tener acceso tanto a la información como a la actividad académica en los centros de educación y de investigación de los cinco continentes. Por tanto, desde 1989 se estableció un enlace vía satélite al nodo **Internet** en el Centro Nacional de Investigaciones Atmosféricas (**NCAR**), en Boulder, Co. E.U.A.

En 1993, debido al inusitado crecimiento del tráfico internacional, se instaló un segundo enlace terrestre, vía fibra óptica, con la empresa **ANS** que opera la red Internet, en Houston, TX., E.U.A.

Otras conexiones externas incluyen enlaces a las redes **RUTyC** (red mexicana de instituciones educativas de la Secretaría de Educación Pública), **MEXNET** (red mexicana de instituciones educativas privadas) y **SIRACyT** (red mexicana de centros de investigación del CONACYT).

Adicionalmente, varias instituciones, públicas y privadas, se conectan directamente a la Red Integral de Telecomunicaciones de la UNAM. Estos enlaces se realizan a través de diversos medios: fibra óptica, radio enlaces, microondas, vía satélite, etc.

En lo que respecta a la transmisión de voz, la Red Integral de Telecomunicaciones se conecta a la red telefónica pública mediante tres enlaces de fibra óptica que transportan 2,400 troncales digitales. Todo el servicio hacia la Red desde el exterior es de entrada directa, es decir, con solo marcar 62+extensión la llamada entra directamente al teléfono del usuario sin intervención humana.

Los servicios que ofrece la Red Universitaria, son administrados por la Coordinación de Servicios de Cómputo de la Dirección de Cómputo para la Investigación de la DGSCA.

El servicio más importante que ofrece la Red es posibilitar el acceso y el intercambio de información académica y científica nacional e internacional a través de tres protocolos: **TELNET** (establecimiento de sesiones remotas en otras computadoras), **FTP** (transferencia de archivos entre computadoras) y **correo electrónico**. Con estos tres protocolos se puede conseguir una infinidad de información como imágenes satelitales para el estado del tiempo, software de dominio público, correo electrónico, etc. Otros servicios disponibles son:

- INFO (sistema integrado de información).
- ARCHIE (localización exacta sin costo de software de dominio público).
- Consulta al catálogo de bibliotecas de la UNAM

- Consulta de bases de datos del Grupo de Interés de Ciencias y Humanidades.
- TELUNAM (directorio telefónico de la UNAM).
- TEXTUNAM (texto completo de publicaciones periódicas nacionales y de Gaceta UNAM).
- Consulta del calendario de cursos de la Dirección General de Servicios de Cómputo Académico.
- Imágenes del Herbario Nacional.
- Imágenes digitalizadas de los diarios nacionales.
- Acceso al sistema de correo electrónico del Programa de Desarrollo de la Organización de las Naciones Unidas, y muchos otros más.

Así, el servicio es la expresión concreta y por ende el beneficio social que la tecnología informática ha proporcionado y convertido en una realidad latente que pretende una mayor respuesta a futuro.

2.

Análisis del problema.

En este tema se plantearán las condiciones actuales en que se encuentra el sistema de generación de historias académicas en microfichas. Se presenta además las propuestas de solución que hemos planteado para la solución del problema y la elección de la más viable dados los recursos con los que actualmente contamos.

2.1 Planeación del sistema.

La planeación del sistema se desarrollará en parte, como resultado de un análisis de las condiciones actuales en las que se realiza el proceso de generación de historias. A continuación, con base en una entrevista a los usuarios finales del sistema actual se pudo analizar los problemas a los que se enfrentan y de esta manera podremos plantear diversas soluciones.

2.1.1 Condiciones actuales del sistema.

En este tema daremos una visión general de cuales son los procesos que se llevan a cabo actualmente para la generación de las historias académicas así como para el almacenamiento de su información, en este caso en microfichas. Pasaremos después a una serie de entrevistas que se enviaron, a través de un

cuestionario a dos de los principales usuarios de los equipos de cómputo y de microfilmación de la DGSCA¹. Pasemos entonces a explicar el como se realiza actualmente el proceso de historias académicas.

Podemos decir que aunque DGAE² es la encargada de toda la administración escolar³ de la UNAM, delega parte de estas responsabilidades a las Secretarías de Servicios Estudiantiles de cada Facultad y Escuela de la UNAM. Siendo así que ellas están responsabilizadas de su propia administración escolar y DGAE será solo la entidad coordinadora y rectora de todos los procesos.

Es así que cada Escuela o Facultad se llevan a cabo los tramites correspondientes a las inscripciones y reinscripciones de sus alumnos del periodo lectivo actual. Al concluir el periodo para inscripciones ordinarias se genera una lista provisional para los maestros, en lo que es posible entregárseles la lista definitiva de alumnos cuando ya ha pasado el periodo de altas, bajas y cambios de grupo.

Con base en esa lista definitiva se generaran al final del semestre las actas correspondientes para que en ellas los profesores asienten las calificaciones de la materia. Pero también durante el transcurso del semestre se generarán actas de examen extraordinario para cada una de las materias que se pueden presentar en alguno de los 3 periodos. Por último se generan actas de alumnos oyentes para cada uno de los grupos en donde existieran este tipo de alumnos, las cuales se consideran como examen extraordinario.

¹ Dirección General de Servicios de Computo Académico.

² Dirección General de Administración Escolar

³ Enténdase por administración escolar todo lo referente a altas, bajas, inscripciones, reinscripciones y control de calificaciones de estudiantes, entre muchas otras cosas.

Casi al terminar el semestre deben estar ya generadas e impresas estas actas para que sean oportunamente repartidas a las diferentes Facultades y Escuelas de la Universidad y de ahí se encarguen de repartirlas a sus profesores para que puedan asentar las calificaciones de examen ordinario sobre la materia que impartieron.

Los profesores las entregan en un período máximo establecido por cada una de las Facultades y Escuelas; al final y cuando ya se han entregado todas las actas, se envía el original de las actas a la DGAE para que ella se encargue del proceso de lectura óptica de la información. Hasta aquí es donde termina la parte de responsabilidad por parte de las Facultades y Escuelas.

Conforme DGAE va recopilando la totalidad de las actas de cada una de las Escuelas y Facultades, estas son enviadas a la DCAA⁴ que se encarga del proceso de lectura óptica.

Ya que han sido capturados los datos, éstos son trasladados hacia el equipo A-12 vía red. Cuando ya se encuentran en el equipo de cómputo A-12 los datos, los cuales hasta el momento son solamente 1's y 0's, son procesados para convertirlos en información utilizable.

Debido a que los programas para este proceso son muy viejos, la mayoría están escritos en lenguajes como Algol y Cobol, los cuales sólo manejan archivos planos y no bases de datos⁵.

⁴ Dirección de Cómputo para la Administración Académica.

⁵ Actualmente existen interfaces para programar a partir de Algol y Cobol aplicaciones para el manejador de datos DMSII (Data Manager System II) de Unisys, pero en el proceso que nos concierne no son utilizadas.

2.1.2 Entrevista.

Para poder tener idea de las necesidades que tiene que cubrir el sistema se hicieron sendas entrevistas a dos de los principales usuarios del servicio de microfilmado de la DCAA: DGAE y DGPA (Dirección General de Personal).

A continuación presentamos la transcripción de la entrevista que fue entregada por Mat. Facundo Ruiz Doncel, Subdirector de Diseño de Proyectos de la Dirección de Administración Escolar dependiente de la Secretaria de Servicios Académicos el día 1 de junio de 1993.

DIRECCION GENERAL DE ADMINISTRACION ESCOLAR
SUBDIRECCION DE SISTEMAS DE REGISTRO ESCOLAR
SUBDIRECCION DE DISEÑO DE PROYECTOS
MAYO, 1993
RESPUESTA AL CUESTIONARIO

1. SEÑALAR LAS FUENTES DE LA INFORMACION QUE SE REQUIERE MICROFILMAR.
BACKUP DE A-12
2. ¿DE QUE MANERA HACE SU DEPENDENCIA LA DISTRIBUCION DE LA INFORMACION MICROFILMANDA AL USUARIO FINAL?
POR MENSAJERIA
3. ¿QUIENES SON LOS USUARIOS FINALES DE ESTAS MICROFICHAS?
- CADA ESCUELA Y FACULTAD DE LA UNAM
- ARCHIVO GENERAL DE LA UNAM
- LA MISMA D.G.A.E.
4. ¿EN CUANTO TIEMPO SE LLEVA A CABO LA DISTRIBUCION?
APROXIMADAMENTE EN UNA SEMANA, PUESTO QUE SE HACE UNA REVISION DE LAS MICROFICHAS Y DESPUES SE ENVIAN
5. SEÑALAR LA CONFIDENCIALIDAD DE LA INFORMACION.
RESPONSABLES DE SERVICIOS ESCOLARES EN LAS ESCUELAS Y FACULTADES Y PERSONAL OPERATIVO EN LA D.G.A.E.
6. ¿UTILIZA SU DEPENDENCIA SERVICIOS EXTERNOS DE

MICROFILMADO O GRABADO DE INFORMACION?

SI; MICROFILMADO DE ACTAS DE EXAMEN CON
CALIFICACIONES Y LAS FIRMAS CORRESPONDIENTES

7. ¿REQUIERE SU DEPENDENCIA DE LA MICROFILMACION DE IMAGENES ADEMAS DE LA INFORMACION DE CARACTERES?

ACTUALMENTE SE REALIZA CON LAS ACTAS DE EXAMEN

8. ¿LA IMAGEN QUE SE MICROFILMA SE USA MAS BIEN DE RESGUARDO O PARA CONSULTA?

PARA AMBAS COSAS

9. ¿CUAL ES LA NECESIDAD DE SU DURABILIDAD?

SI ENTENDEMOS BIEN; COMO RESGUARDO ES INDEFINIDO
COMO CONSULTA POR LO GENERAL ES DE UN SEMESTRE A
UN AÑO ESCOLAR.

10. ¿CON QUE PERIODICIDAD O FRECUENCIA SE CONSULTA ESTA INFORMACION?

DEPENDE DE CADA USUARIO; EN LA MAYORIA DE LOS
CASOS ES MUY FRECUENTE EN LOS PERIODOS QUE SE
PROGRAMAN COMO "AJUSTES" A LAS HISTORIAS
ACADEMICAS, QUE SE REALIZAN A LA CONCLUSION DE
CADA SEMESTRE EN LAPROS DE 60 DIAS.

11. ¿QUE TIPO DE INFORMACION SE CONSULTA CON MAS FRECUENCIA: LA HISTORICA, LA MAS RECIENTE O INDISTINTAMENTE?

INDISTINTAMENTE

12. SEÑALAR LA CONCURRENCIA O CONSULTAS SIMULTANEAS POR PARTE DE LOS USUARIOS FINALES A ESTE TIPO DE INFORMACION.

SI ENTENDEMOS BIEN, NO SE LLEVAN A CABO CONSULTAS
SIMULTANEAS, DADO QUE SOLO SE GENERAN 3
MICROFICHAS POR APLICACION, UNA DE RESGUARDO (LA
ORIGINAL) UNA AL USUARIO (ESCUELAS Y FACULTADES)
Y UNA PARA USO DE LA D.G.A.E.

13. ¿CUAL ES EL TIEMPO MAXIMO DE ACCESO TOLERADO EN LA CONSULTA DE LA INFORMACION UNA VEZ QUE SURGE LA NECESIDAD DE SU CONSULTA?.

ACTUALMENTE, EL TIEMPO QUE SE LLEVA A CABO LA
LOCALIZACION DE LA MICROFICHA Y LA CONSULTA DE LA
INFORMACION A TRAVES DEL LECTOR. SIN EMBARGO, POR
LOS TRAMITES PARA LOS QUE UTILIZAN LAS MICROFICHAS
SERIA RECOMENDABLE DISMINUIR ESOS TIEMPOS.

14. SEÑALAR LOS MEDIOS ALTERNOS A LAS MICROFICHAS QUE UTILIZA SU DEPENDENCIA PARA LA DISTRIBUCION DE SU INFORMACION.

- LISTADOS Y REPORTES
- ARCHIVOS DE DATOS ENTRE EL EQUIPO A-12 Y EQUIPOS
PC, POR LINEA O POR DISKETTE, CINTA O CARTUCHO

15. SUGERENCIAS Y COMENTARIOS.

HEMOS ANALIZADO NUESTRAS APLICACIONES Y CONSIDERAMOS QUE
PODEMOS PRESCINDIR DE LAS MICROFICHAS (DE DATOS) SI
DISPONEMOS DE OTRO TIPO DE RECURSO QUE NOS PERMITA

RESOLVER LAS NECESIDADES DE INFORMACION, COMO SON:

1. DIRECTORIO DE ALUMNOS; LA MICROFICHAS CON LOS DIRECTORIOS DE ALUMNOS QUE SE GENERAN PARA EL ARCHIVO GENERAL Y LAS OFICINAS DE LA D.G.A.E. (EDIFICIO PROXIMO AL METRO C.U.) PUEDEN CANCELARSE SI UTILIZAMOS CD-ROM. EN UN CD-ROM SE GENERARIA TODO EL DIRECTORIO, APROXIMADAMENTE 1'500,000 NUMEROS DE CUENTA CON DATOS DE LOS ALUMNOS Y SE REQUERIRIAN UNOS 5 ORIGINALES POR SEMESTRE ESCOLAR.

REQUERIMIENTOS: 5 CONTROLADORES DE CD-ROM PARA INSTALARSE EN EQUIPOS PC, GENERACION DEL DIRECTORIO TOTAL (INCLUIR ALGUNOS DATOS NO CONSIDERADOS ACTUALMENTE) Y PRODUCIR 5 CD-ROM POR SEMESTRE ESCOLAR.

2. HISTORIAS ACADEMICAS; LAS MICROFICHAS DE HISTORIAS ACADEMICAS SE PUEDEN OMITIR SI CADA ESCUELA Y FACULTAD TIENE ACCESO AL EQUIPO CENTRAL PARA LA CONSULTA EN LINEA O TRANSFERENCIAS DE SU INFORMACION.

ACTUALMENTE SE GENERAN MICROFICHAS DE HISTORIAS ACADEMICAS DEBIDO A QUE LA MAYORIA DE LOS PLANTELES NO TIENEN ACCESO DIRECTO A LA CONSULTA EN EL EQUIPO CENTRAL, ADEMAS SE HA INICIADO EL ENVIO DE HISTORIAS ACADEMICAS A LOS EQUIPOS LOCALES A TRAVES DE DISKETTES.

REQUERIMIENTOS: COMUNICACION ENTRE EL EQUIPO A-12 Y LOS EQUIPOS (PC'S O REDES) DE LAS AREAS DE ADMINISTRACION ESCOLAR DE LOS PLANTELES, CON POSIBILIDADES DE CONSULTA EN LINEA Y TRANSFERENCIAS DE DATOS DE EQUIPO A EQUIPO.

3. MICROFICHAS DE ACTAS DE EXAMEN (QUE SE GENERAN EN FORMA EXTERNA); SERIA CONVENIENTE CONSIDERAR ALGUNA ALTERNATIVA PARA CAMBIAR EL MICROFILMADO DE ACTAS DE EXAMEN, PUESTO QUE, LA MICROFILMACION QUE SE HACE ES PARA RESGUARDO Y PARA QUE CADA PLANTEL TENGA UNA COPIA CON LA IMAGEN EXCLUSIVA DE SUS ACTAS.

Pasemos ahora a la transcripción de la entrevista que fue entregada por el Lic. Ángel Trujillo Negrete Director General de la Dirección General de Personal entregado el día 11 de junio de 1993.

CUESTIONARIO DE DGSCAC SOBRE MICROFILMACIÓN 02/06/93

1. SEÑALAR LAS FUENTES DE LA INFORMACIÓN QUE SE REQUIERE MICROFILMAR. (BACKUP DE A-12, BASES DE DATOS).

ACTUALMENTE EN DGSCA SE MICROFILMAN LOS ARCHIVOS DE NÓMINA EN TANTO QUE, EN LA DGP SE REALIZA LA

MICROFILMACIÓN DE LOS EXPEDIENTES Y KÁRDEX DEL PERSONAL.

LOS ARCHIVOS DE NÓMINA MICROFILMADOS SE PRESENTAN DE LA SIGUIENTE FORMA:

- A) RELACIÓN ALFABÉTICA DE PAGOS A NIVEL UNAM.
- B) RELACIÓN ALFABÉTICA DE PAGOS POR DEPENDENCIA.
- C) ARCHIVO DE CARGA (MODIFICACIONES A LA NÓMINA -V.G. CAMBIO DE LUGAR DE PAGO-).

2. ¿DE QUÉ MANERA HACE SU DEPENDENCIA LA DISTRIBUCIÓN DE INFORMACIÓN MICROFILMADA AL USUARIO FINAL? (POR MENSAJERIA, CORREO, PASAN ELLOS A RECOGERLAS, ETC.)

LA INFORMACIÓN DE NÓMINA SE ENTREGA MEDIANTE PROPIOS, CON EXCEPCIÓN DE LA DGAPA, QUIENES VIENEN POR ELLA.

LA INFORMACIÓN DE LOS EXPEDIENTES ES PARA USO INTERNO Y RESPALDO. LAS CONSULTAS SE REALIZAN DIRECTAMENTE EN LOS EXPEDIENTES Y SÓLO DE MANERA EXCEPCIONAL SE CONSULTAN LAS MICROFICHAS.

3. ¿QUIENES SON LOS USUARIOS FINALES DE ESTAS MICROFICHAS? LOS USUARIOS DE LA INFORMACIÓN DE NÓMINA SON LOS QUE A CONTINUACIÓN SE RELACIONAN.

3.1 DIRECCIÓN GENERAL DE PERSONAL.

A) ORDENADAS ALFABÉTICAMENTE.

- DIRECCIÓN GENERAL
- DEPARTAMENTO DE MOVIMIENTOS ACADÉMICOS
- DEPARTAMENTO DE PRESTACIONES
- MÓDULO ACATLÁN
- SUBDIRECCIÓN DE ESTUDIOS ADMINISTRATIVOS
- MÓDULO IZTACALA
- DEPARTAMENTO DE SERVICIOS
- DEPARTAMENTO DE ACLARACIONES A NÓMINA
- MÓDULO CUERNAVACA

B) ORDENADAS POR RFC.

- DEPARTAMENTO DE ACLARACIONES A NÓMINA
- MÓDULO ACATLÁN
- MÓDULO CUERNAVACA

3.2 OTROS USUARIOS.

A) ORDENADAS ALFABÉTICAMENTE.

- DIRECCIÓN GENERAL DE CONTROL E INFORMÁTICA
- DIRECCIÓN GENERAL DE ASUNTOS DEL PERSONAL ACADÉMICO
- COORDINACIÓN GENERAL DE ASUNTOS LABORALES
- DIRECCIÓN GENERAL DE PROGRAMACIÓN Y PRESUPUESTACIÓN

B) ORDENADAS POR RFC.

- DIRECCIÓN GENERAL DE PROGRAMACIÓN Y PRESUPUESTACIÓN

Capítulo II Planteamiento del problema.

LA INFORMACIÓN MICROFILMADA EN EL ARCHIVO DE EXPEDIENTES DEL PERSONAL NO SE DISTRIBUYE DEBIDO A QUE CONSTITUYE SÓLO UN RESPALDO QUE SE CONSULTA POR EXCEPCIÓN EN LAS PROPIAS OFICINAS DEL ARCHIVO.

4. ¿EN CUÁNTO TIEMPO SE LLEVA A CABO LA DISTRIBUCIÓN? LA INFORMACIÓN DE NÓMINA SE DISTRIBUYE EN TRES DIAS MÁXIMO. LA INFORMACIÓN DE EXPEDIENTES NO SE DISTRIBUYE.
5. SEÑALAR LA CONFIDENCIALIDAD DE LA INFORMACIÓN (JERARQUÍAS DE ACCESO).

LA INFORMACIÓN DE LA MICROFILMACIÓN SE MANEJA CON UN ALTO NIVEL DE CONFIDENCIALIDAD EN LAS ÁREAS QUE REQUIEREN DE ESA INFORMACIÓN PARA CONSULTA O LA REALIZACIÓN DE TRÁMITES. ATENDIENDO A LAS OPORTUNIDADES GENERADAS POR LAS NUEVAS TECNOLOGÍAS SERÍA CONVENIENTE QUE ÉSTA INFORMACIÓN ESTUVIERA A DISPOSICIÓN DE CADA DEPENDENCIA UNIVERSITARIA EN TRES NIVELES DE CONSULTA. EN EL PRIMER NIVEL SE TENDRÍA INFORMACIÓN COMPLETA DE LA PROPIA DEPENDENCIA PARA CONSULTA DE LA MISMA; UN SEGUNDO NIVEL DE CONSULTA, EN DONDE LAS DEPENDENCIAS PUEDAN OBSERVAR SI SU PERSONAL TIENE COMPROMISOS LABORALES CON OTRAS DEPENDENCIAS EN ASPECTOS TALES COMO: NOMBRAMIENTOS, CATEGORÍAS, COMPROMISOS DE HORAS, ETC. ESTE NIVEL DE CONSULTA NO DARÍA ACCESO A INFORMACIÓN DE SUELDOS. EN EL TERCER NIVEL, LAS DEPENDENCIAS SOLO PODRÍAN ACCESAR INFORMACIÓN DE RFC'S Y ADSCRIPCIONES EN DEPENDENCIAS DE TODO EL PERSONAL UNIVERSITARIO.

6. ¿UTILIZA SU DEPENDENCIA SERVICIOS EXTERNOS DE MICROFILMADO O GRABADO DE INFORMACIÓN? (Y EN SU CASO, PARA QUÉ TIPO DE INFORMACIÓN).

NO SE UTILIZAN SERVICIOS EXTERNOS. PARA LA MICROFILMACIÓN DE NÓMINA SE CUENTA CON EL APOYO DE LA DGSCA Y SE CUENTA CON UN EQUIPO PROPIO PARA LA MICROFILMACIÓN DE EXPEDIENTES.

7. ¿REQUIERE SU DEPENDENCIA DE LA MICROFILMACIÓN DE IMÁGENES, ADEMÁS DE INFORMACIÓN EN CARACTERES?

SÍ, PARA EL EXPEDIENTE DE CADA TRABAJADOR, EN TANTAS IMAGENES COMO TENGA EN DOCUMENTOS EL ARCHIVO FÍSICO DE LA DGP (V.G. ACTA DE NACIMIENTO, TÍTULOS, ETC.). HAY APROXIMADAMENTE 55,000 TRABAJADORES VIGENTES EN NÓMINA.

8. ¿LA INFORMACIÓN QUE SE MICROFILMA, SE USA MÁS BIEN RESGUARDO O PARA SU CONSULTA?

PARA AMBAS SITUACIONES: LA INFORMACIÓN DE NÓMINA SE UTILIZA PRIMORDIALMENTE PARA CONSULTA, EN TANTO QUE LA INFORMACIÓN DE LOS EXPEDIENTES CONSTITUYE UN MECANISMO DE RESGUARDO Y EN MENOR MEDIDA DE CONSULTA.

9. ¿CUÁL ES LA NECESIDAD DE SU DURABILIDAD?

CADA MICROFICHA TIENE SENTIDO DE EXISTIR EN LA MEDIDA DE QUE LOS TRABAJADORES DE LA UNAM ESTÁN ADSCRITOS A LA INSTITUCIÓN.

POR CITAR UN EJEMPLO, CADA VEZ QUE UN TRABAJADOR UNIVERSITARIO SE JUBILA Y REQUIERE DE INFORMACIÓN (TAL COMO LA HISTORIA SUS SUELDOS TABULARES) PARA TRÁMITES ANTE EL ISSSTE, HOY DÍA SE CONSULTAN AL EFECTO LAS NÓMINAS PAGADAS POR LA UNAM HISTÓRICAMENTE.

10. ¿CON QUÉ PERIODICIDAD O FRECUENCIA SE CONSULTA ESTA INFORMACIÓN?

SE CONSULTA A DIARIO, AUNQUE ACTUALMENTE LA CONSULTA DE EXPEDIENTE NO SE HACE EN LA MICROFICHA.

11. ¿QUÉ TIPO DE INFORMACIÓN SE CONSULTA CON MÁS FRECUENCIA: LA HISTÓRICA, LA RECIENTE O INDISTINTAMENTE?

AMBOS CASOS PRESENTAN UN ALTO NIVEL DE DEMANDA, AUNQUE SIEMPRE ES SUPERIOR LA CONSULTA RECIENTE. LOS REQUERIMIENTOS DE LA INSTITUCIÓN Y DE SU PERSONAL SON TAN VARIADOS, QUE JUSTIFICAN POR SI MISMOS EL ALMACENAMIENTO DE LA INFORMACIÓN EN CUESTIÓN.

12. SEÑALAR LA CONCURRENCIA O CONSULTAS SIMULTÁNEAS POR PARTE DE LOS USUARIOS FINALES A ESTE TIPO DE INFORMACIÓN.

A NIVEL DE TRÁMITACIÓN LA CONSULTA ES SIMULTÁNEA EN VARIAS OCASIONES AL DÍA. A NIVEL EJECUTIVO LA CONSULTA ES ESPORÁDICA.

13. ¿CUÁL ES EL TIEMPO MÁXIMO DE ACCESO TOLERADO EN LA CONSULTA DE LA INFORMACIÓN UNA VEZ QUE SURGE LA NECESIDAD DE SU CONSULTA?

SE REQUIERE ACCESO INMEDIATO POR EL CARÁCTER DE LA OPERACIÓN (SERVICIO AL PERSONAL). EN EL CASO DE QUE SE LLEGASE A ESTABLECER UN SERVICIO DE CONSULTA PARA LAS UNIDADES ADMINISTRATIVAS DE LAS DEPENDENCIAS, EL ACCESO TENDRÍA QUE SER INMEDIATO TAMBIÉN.

14. SEÑALAR LOS MEDIOS ALTERNOS A LAS MICROFICHAS QUE UTILIZA SU DEPENDENCIA PARA LA DISTRIBUCIÓN DE SU INFORMACIÓN.

PROPONEMOS LA CONSULTA A LA NÓMINA MEDIANTE EL PROGRAMA NOM DESARROLLADO EN LA DGP (PREVIA ASIGNACIÓN DE CLAVES DE ACCESO), VIA RED UNAM A TODAS LAS DEPENDENCIAS. POR LO QUE CONCIERNE A LA INFORMACIÓN DE LOS EXPEDIENTES NO SE CUENTA CON UNA VALORACIÓN AL RESPECTO.

15. SUGERENCIAS Y COMENTARIOS.

SITUACIÓN PROPUESTA :

I. PARA RESPALDO Y ARCHIVO HISTÓRICO DE LA INFORMACIÓN DE NÓMINA, SUBSTITUIR LAS MICROFILMACIONES ACTUALES POR LA GRABACIÓN DEL ARCHIVO ORIGINAL DE NÓMINA (CADA QUINCENA) EN CD-ROM. EN ESTE CASO SE PROPONE ALMACENAR EXCLUSIVAMENTE LOS DATOS, EN FORMATO DE REGISTRO.

II. PARA FINES DE CONSULTA PARA TRÁMITE LA DGP CUENTA

YA CON UN PROGRAMA DE CONSULTA A LA NÓMINA, EN FORMATO DE BASE DE DATOS, QUE PERMITE ACCESAR LA INFORMACIÓN VÍA RED LOCAL EN MODO MULTIUSUARIO, AL PERSONAL QUE TRAMITA DIARIAMENTE MOVIMIENTOS A NÓMINA. SE PROPONE PERMITIR EL ACCESO A LAS DEPENDENCIAS VÍA RED UNAM.

III. PARA LA MICROFILMACIÓN DE EXPEDIENTES SUBSTITUIR MICROFILMACIÓN POR SISTEMAS DE DIGITALIZACIÓN.

ACTUALMENTE EL NIVEL DE CONFIABILIDAD DE LA TRANSFERENCIA DEL ARCHIVO DE NÓMINA DEL EQUIPO A-9 DE DGSCAD A LA RED LOCAL DE DGP ES DE UN 97%. A FIN DE HACER TRASCENDER EL SISTEMA DE CONSULTA AL NIVEL QUE SE REQUIERE, SE NECESITA LA CONFIABILIDAD DE TRANSMISIÓN A UN NIVEL DEL 100%.

Con base a la información recabada gracias a estas entrevistas así como la información con la que ya contábamos, fue posible que pudiéramos crear algunas propuestas de solución utilizando diversas tecnologías las cuales se explican a detalle en el siguiente tema.

2.2 Propuestas de solución.

De acuerdo con la descripción anterior de los recursos que se tienen en la **Dirección de Cómputo para la Administración Académica (DCAA)** y las entrevistas anteriores, el objetivo es el de buscar las mejores opciones para el almacenamiento masivo de la información de las historias académicas y un mecanismo de consulta que sea ágil y de fácil uso.

Debido a las condiciones actuales del sistema, estas propuestas deben de ser lo suficientemente abiertas para sustituir de manera definitiva el antiguo sistema de microfilmación, además deben de ser congruentes con las plataformas de **hardware** que la dependencia tiene. En cuanto a estas se tienen tres opciones:

- **Mainframe A12 Unisys.**

Actualmente es en este equipo en donde se lleva todo el proceso de la información académica de la UNAM. Es decir, en este mainframe se lleva el proceso de listas de alumnos, impresión de actas, impresión de historias académicas, etc. Debido a su poder de procesamiento, esta tarea la realiza con relativa facilidad, sin embargo la carga de trabajo es colosal. Además de prestar este servicio, da atención a muchos usuarios de diversas dependencias para el proceso de sus datos, análisis estadísticos, etc. En cuanto al almacenamiento de la información se cuentan con dos alternativas: el uso de cintas magnéticas y espacio en disco duro, aunque cabe resaltar que este es un punto crítico del equipo, debido a todas las funciones que tiene a su cargo.

Ahora, en cuanto al ambiente que presenta al usuario, es un poco complicado, no es tan amigable como se espera. Además presenta un inconveniente: no da una fácil interconectividad con las nuevas tecnologías de almacenamiento masivo de la información. Podemos añadir que este equipo es un poco viejo, más no obsoleto, además de que su mantenimiento es costoso.

Sin embargo presenta varias ventajas: la más obvia es que es en este equipo donde se procesa la información de las historias académicas, no tenemos que migrar la información a algún otro sistema para su proceso y conversión; una segunda ventaja es el lenguaje nativo de programación: **ALGOL**, el cual es muy sencillo de aprender y da la ventaja de construir aplicaciones robustas en poco tiempo.

- **Redes de computadoras personales.**

En la actualidad el uso de redes de computadoras personales es lo más común. Comparten la información de una forma muy fácil y sencilla, además de que se tienen al alcance muchos de los paquetes comerciales que se utilizan, como pueden ser procesadores de texto, hojas electrónicas, lenguajes de programación, bases de datos, etc.

Presentan un gran facilidad en cuanto a la interconectividad de dispositivos de almacenamiento masivo, con la ventaja de que se pueden tener más de uno a la vez.

Tienen una gran interfaz para el usuario, la cual es la más conocida a nivel mundial, y si se tiene corriendo **MSDOS** como **sistema operativo**, la ventaja es mayor. Sin embargo presenta un inconveniente: el poder de procesamiento.

Como sabemos, las computadoras PC fueron diseñadas originalmente para uso monousuario, es decir, una sola persona puede estar haciendo uso de ella en un momento dado, y solamente se puede tener un proceso ejecutándose a la vez.

Si consideramos el hecho de que el tamaño de la información de las historias académicas es de más de 600 Mb, las computadoras personales serían un poco lentas para manejar dicho volumen. Sin embargo, existen paquetes de **indexamiento**, los cuales son muy eficientes en lo que respecta a la consulta de la información. Para la consulta de la información, se tendría una sola máquina destinada a tal propósito, que tendría la función de **servidor**.

- **Sun SparcStation.**

Como hemos visto anteriormente, el equipo SparcStation es una estación de trabajo que presenta varias ventajas: la primera es su poder de procesamiento (arquitectura SPARC), gran interconectividad entre dispositivos de almacenamiento masivo, que van desde la cintas magnéticas hasta CD-ROM's; una interfaz amigable al usuario (similar al que ofrece una PC), con la ventaja de tener un sistema operativo multiusuario como lo es UNIX.

En cuanto a los medios de almacenamiento masivo de la información, actualmente se tiene una gran diversidad de dispositivos: medios de grabación ópticas, magnéticas o una tecnología híbrida, es decir, una tecnología que utiliza ambos medios. Sin embargo el mercado de cada uno de estos dispositivos puede abarcar está bien delimitado.

2.2.1 Sistemas de almacenamiento masivo de información.

DISCOS FLÓPTICOS.

Los *discos flópticos* utilizan una tecnología muy similar a la de los discos magnéticos de 3.25", sin embargo el *drive* (lector) contiene cabezas sensibles a marcas ópticas grabadas en el diskette desde la fabricación del mismo, por lo que permite un control más fino de la lectura/escritura de la información. Gracias a esto, se puede grabar hasta un total de 20.8Mb en un solo disco. Una cosa interesante es que varios de los *drives* que leen estos discos, también pueden leer discos magnéticos del mismo formato, es decir, de 3.25 pulgadas.

La principal desventaja de los **discos flópticos** radica en su lentitud, dado que tiene la misma velocidad que los discos magnéticos comunes. De aquí se desprende su principal utilidad: servir como un medio de respaldo de archivos de frecuente uso.

El costo de un **drive flóptico** es de alrededor de **\$590⁶** dólares, con un costo por unidad (discos) de **\$45** dólares.

DISCOS MAGNETO-ÓPTICOS.

Los discos magneto-ópticos utilizan una tecnología híbrida, grabando **información óptica** en un **medio magnético**. Su capacidad va de los **128MB** a los **650MB**, por lo que son comparables con un dispositivo CD-ROM. Para grabar información en un disco magnetoóptico, el drive deberá calentar por medio de un **láser** el área a utilizar, después por medio de campos magnéticos las partículas del disco deberán ser orientadas en una dirección específica. La lectura de un dispositivo magneto-óptico se basa en el **efecto Kerr** o sea en el cambio del plano de polarización de un haz de luz cuando éste es reflejado por una superficie magnetizada.

Una de las desventajas de los dispositivos magneto-ópticos es la velocidad de grabación ya que se deberán grabar primero los 0s y después los 1s. Otra desventaja es el calor disipado por los drives, debido a que el disco deberá calentarse para su grabación.

⁶ A partir de este momento, todos los precios a los que hacemos mención son del año 1993.

El costo de un drive magnetoóptico va de **\$1000** a **\$4000** dólares, con un costo por disco de **\$64** a **\$90** por los de **128MB** o de **\$200** dólares aproximadamente por los de **650MB**.

Dado que para la lectura los discos magnetoópticos tienen una velocidad comparable a la de un disco duro (**40 ms.** de tiempo de acceso), son buena elección cuando lo que se quiere es expandir la capacidad de almacenamiento primario. Además su gran robustez (no son susceptibles a campos magnéticos) y durabilidad (no hay posibilidad de roce entre las cabezas del drive y el disco) los hace adecuados para ambientes no condicionados.

CD-ROM.

Los discos ópticos CD-ROM han conformado un estándar ampliamente aceptado para la grabación masiva de información. El estándar ISO-9660 es el más difundido; y gran cantidad de información es distribuida en él, desde software de sistema, manuales, hasta enciclopedias o información propia de usuarios.

Los drives que tradicionalmente se han utilizado para CD-ROM solo pueden leer, basándose en la reflexión o no de un haz láser. Recientemente han surgido drives que pueden además de leer, escribir en un disco CD-R (como se llaman los discos CD susceptibles de ser grabados), con la gran ventaja de que estos discos pueden ser leídos en un drive CD-ROM normal por medio del estándar ISO-9660.

Una de las desventajas de un drive CD-ROM es la velocidad, ya que tienen velocidades de acceso de aproximadamente **450ms.** y una velocidad de

transferencia de aproximadamente **150kbps**, aunque existen drives de doble velocidad (apropiados para aplicaciones de multimedia) con velocidades de acceso de **280ms**. y transferencia de **300kbps**. Como comparación una unidad de disco duro tiene velocidades de acceso de aproximadamente 20ms, aunque han surgido unidades de 9ms y velocidades de transferencia de hasta 10MBps.

La escritura de un drive **CD-R** es lenta, va de 1/2 hora a 1 hora, y dado que deberá hacerse a una velocidad constante, es necesario contar con una imagen previa. Lo anterior es resuelto por los proveedores de unidades CD-R de dos maneras distintas, ya sea contando con un disco duro en donde guardar la imagen, o simplemente simulando la imagen (lo que hace el proceso más lento y riesgoso), algunos vendedores incluyen el disco duro necesario dentro de la misma unidad de grabación.

El costo de un drive CD-R con capacidad de escritura oscila entre los **\$9,000** dólares y los **\$13,000** dólares, con un costo por disco que oscila entre los **\$25** dólares y los **\$40** dólares. Un drive con capacidad de solo lectura no deberá exceder un costo de **\$1,000** dólares.

OTROS DISCOS.

Existen otras opciones en cuanto a lo que toca al almacenamiento masivo de información, sin embargo su costo o baja capacidad no los hacen adecuados para la aplicación que se proyecta.

Discos Bernoulli.

Estos discos, aunque siguen una tecnología magnética, son extremadamente seguros en la prevención de aterrizajes de cabezas ya que gracias al efecto Bernoulli es el disco el que se acerca a la cabeza y no la cabeza al disco. Tienen una capacidad de **90MB** y un drive cuesta aproximadamente **\$1000** dólares con un costo por disco de **\$229** dólares.

Drives de cartucho.

Siguen el mismo esquema de grabación escritura que un disco duro normal, solo que al ser removibles, la posibilidad de aterrizaje de cabezas aumenta y son más susceptibles a la intrusión de partículas de polvo. Lo anterior no permite que las cabezas estén tan cercanas a la unidad de grabación como en un disco duro normal, lo que disminuye la densidad de grabación.

Un disco de 5.25" proporciona una capacidad de **88 MB** y un drive cuesta de **\$1000** a **\$1300** dólares, con un costo por disco que oscila entre los **\$200** y los **\$300** dólares. La velocidad que éstas unidades proporcionan es excelente y en nada diferenciable de la de un disco duro fijo.

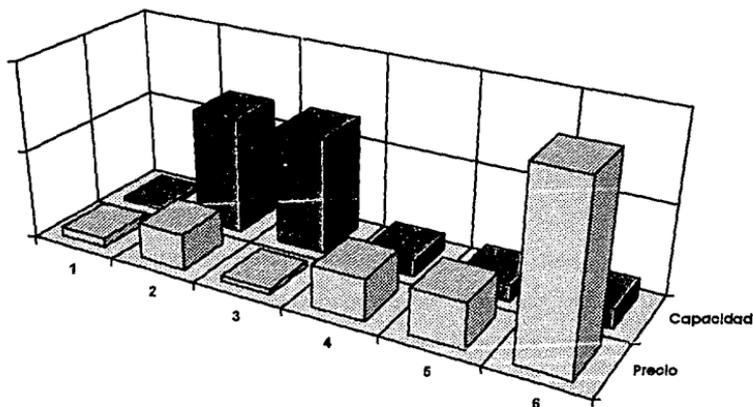
Drives SCSI removibles.

El rendimiento de estas unidades es igual a la de un disco **SCSI** fijo, con un capacidad de almacenamiento de **105MB** un drive cuesta aproximadamente **\$1200** dólares con un costo por disco que oscila entre los **\$700** y los **\$1100** dólares.

Drives Worm.

La utilización de éstos drives ha bajado en gran medida, se espera que para 1997 esta tecnología deje de ser utilizada, además de que no existen estándares de aceptación general.

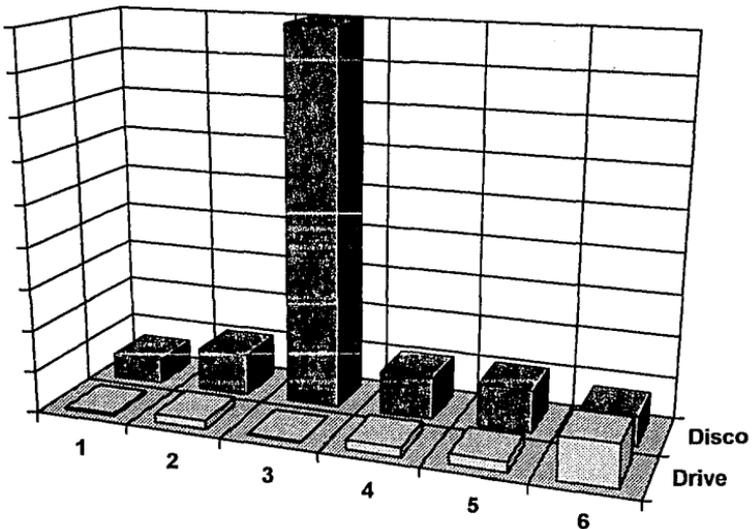
A continuación se presenta una gráfica que muestra las relaciones capacidad y costo de cada una de las tecnologías anteriores, así podemos tener un punto de comparación de cada una de ellas para decidir cual es la más conveniente.



La descripción de cada una de las barras es la siguiente:

Tecnología	Precio por unidad [US\$]	Capacidad [MB]
1. Disco Flóptico	50	20
2. Disco Magnetoóptico	200	650
3. CD-ROM	30	650
4. Bernoulli	229	90
5. Cartuchos	250	80
6. SCSI	1000	105

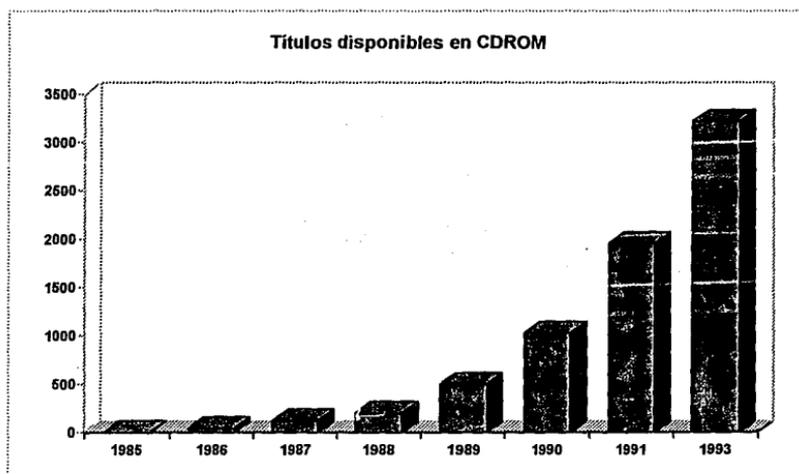
Ahora bien, podemos tener otra gráfica comparativa de los precios de las unidades lectoras (y grabadoras en su caso) contra el precio por unidad de almacenamiento:



Tecnología	Precio por drive [US\$]	Precio por unidad [US\$]
1. Disco Flóptico	590	20
2. Disco Magnetoóptico	1000	200
3. CD-ROM	9000	30
4. Bernoulli	1000	229
5. Cartuchos	1000	250
6. SCSI	1000	1000

Para esta tabla se tomo el mejor de los caos, es decir, se tomo el precio más barato que se encuentra en el mercado.

De las gráficas podemos decir que el CD-ROM es la opción más viable, sin embargo es un poco más costosa que las otras, sin embargo, esta tecnología se va desarrollando rápidamente y estos costos se van reduciendo, además el formato CD-ROM se está convirtiendo en un estándar en lo que respecta a almacenamiento masivo de la información, tal y como lo muestra la siguiente gráfica:



SITUACIÓN ACTUAL.

De acuerdo a un reporte anual del estado del sistema de microfilmación se observa que existen principalmente cuatro dependencias que utilizan dicho servicio. Por una parte está la **DGAE**. que 2 veces por año graba información de **historias académicas**, esto lo hace en lotes aproximadamente como sigue:

#páginas	#caracteres por página	Megabytes
88,440	2,131	179.73MB
55,744	2,131	113.28MB
75,296	2,131	153.02MB
77,376	2,131	157.24MB

TOTAL 603.27MB

Los anteriores parciales corresponden a **Facultades, E.N.E.P., C.C.H. y E.N.P.**

La nómina de la UNAM utiliza más frecuentemente el servicio (2 veces por mes):

#páginas	#caracteres por página	Megabytes
3,224	2,663	8MB
2,250	2,569	5.51MB
3,232	2,610	8.04MB
3,224	2,610	8.02MB

TOTAL 26MB

Usuarios que no tienen una periodicidad para la utilización del servicio son:

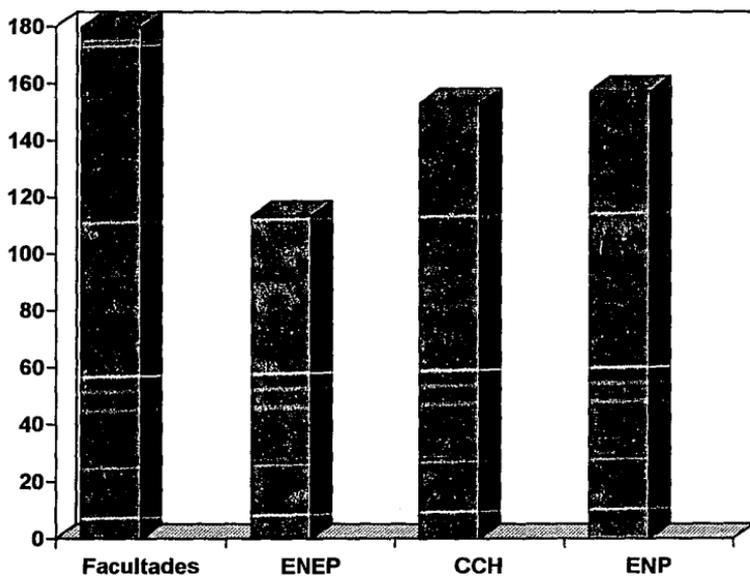
TIENDA UNAM

13.78MB

PATRONATO UNAM

19.466MB

En el caso de las historias académicas podemos ver estas relaciones en la siguiente gráfica:



Cabe señalar que la medición de Megabytes antes realizada se llevó a cabo considerando solo los caracteres leídos. Actualmente la técnica de microfilmación

llena un formato con los caracteres leídos de una cinta, después se reproduce una fotografía de dicho formato. El grabar la imagen completa del documento consumiría mucho más espacio, sin embargo no se considera necesario, ya que de cualquier manera un equipo con la capacidad de leer un CD-ROM también tendrá la capacidad de correr un sistema que reproduzca en su pantalla el formato deseado lleno con los datos leídos.

Los factores a considerar en la elección de un dispositivo de almacenamiento como el propuesto son los siguientes:

- Costo de compra de unidad lectora/escritora
- Costo de unidad lectora y de equipo en donde se deberá conectar dicha unidad.
- Costo de medios de grabación
- Reusabilidad de las unidades lectoras en otras aplicaciones
- Tiempo de grabación
- Longevidad y duración del medio de grabación

Las dos principales dependencias usuarias tienen requerimientos diferentes en cuanto a la cantidad de información grabada. Por una parte la **DGAE** demanda mayor cantidad de fichas por lote, sin embargo requiere de los servicios de microfilmación menos frecuentemente que la **DGAPA**.

Debido a los requerimientos de **DGAE**, y si pensamos que no se utilicen medios distintos de respaldo para cada dependencia, las opciones se restringen a discos **CD-R** o **discos magnetoópticos**. Si, por otra parte, se utilizan medios distintos, la elección sería de **discos flópticos** para las aplicaciones de Nómina,

Tienda UNAM y Patronato, utilizándose discos CD-R o discos magneto-ópticos para DGAE.

En todas las opciones habría que implementar o adquirir un sistema de software que permitiera al usuario final desplegar y opcionalmente imprimir el documento llenando una forma predeterminada con los datos leídos.

Cualquiera de las opciones tiene una velocidad de grabación mayor a la de la unidad de microfichas (70mf/hora x208 pag. por microficha x 5120 caract./pag = 71MB x hora).

El costo de los discos es semejante en todas las opciones, llegando a ser más barata la de CD-R, por otra parte, aunque su longevidad no está comprobada, se espera sea mayor que la de los discos CD-ROM debido a que la capa reflectora de oro que contienen es más durable que la de aluminio de los CD-ROM. Sin embargo los fabricantes de discos CD-R especifican que deben manejarse con cuidado ya que son más frágiles que los CD-ROM. La longevidad de los discos flópticos es mucho menor, ya que existe rozamiento entre la cabeza del drive y el disco.

La gran ventaja que tienen los discos CD-R es el que las unidades lectoras pueden ser utilizadas no solo para leer los datos que los usuarios requieren, y que antes se encontraban en microfichas, sino que a través de ellas el usuario tendrá acceso a toda la gama de información que actualmente comienza a ser distribuida en CD-ROM. Por otra parte, el uso de discos magneto-ópticos solo permitiría al usuario leer los mismos datos que hubiera tenido a través de las fichas, y opcionalmente de nuevas aplicaciones que él generara.

El costo de la unidad lectora es semejante para el caso de los discos CD-R y magneto-ópticos, siendo menor en un 50% para los discos flópticos.

El costo de la unidad de grabación es mucho mayor para el caso de los discos CD-R, sin embargo se debe considerar que adquiriendo una de estas unidades se podrán reproducir el número de discos que se deseen. Por otra parte, si comparamos el costo de una unidad de microfilmación (\$262,365 dólares) con el de una unidad de CD-R existe una diferencia muy considerable (costo 20 veces menor) aún sin considerar el costo de mantenimiento.

2.2.2 Sistemas de Consulta de Información.

Una vez que hemos analizado a los sistemas masivos de información pasaremos a analizar a los sistemas de consulta para esa información.

Para este caso nos encontramos con varias opciones las cuales iremos describiendo a continuación, cada una con sus características específicas.

SYBASE (Manejador de bases de datos).



SYBASE es el nombre genérico para un grupo de productos de la Sybase Inc. Los principales productos de esta compañía son: Servidor SQL⁷ (SQL Server) y Utilerías de SQL (SQL Toolset). El SQL

⁷ SQL significa Structured Query Language, lenguaje estándar para el manejo y diseño de aplicaciones con bases de datos.

Capítulo II Planteamiento del problema.

Server es un *sistema manejador de bases de datos* (database management system, DBMS) que corre bajo una gran variedad de plataformas y sistemas operativos, incluyendo:

DEC VAX, Micro VAX, etc.	-	VMS, UNIX, ULTRIX
AT&T 3B series	-	UNIX
Stratus	-	UNIX
Sequent	-	UNIX
Pyramid	-	UNIX
Sun Workstations	-	UNIX
IBM PC/XT, PC/AT, etc	-	PC/DOS, OS/2
IBM PS/2	-	OS/2

A continuación se presenta el ambiente que presenta SYBASE al usuario bajo Solaris 2.2:

```

Sybase Data Workbench
Data Workbench Login
L
User name: root
Password:

Sybase Data Workbench Serial #: 112549
Data Workbench/5.2/P/Solaris/SunOS 5.1/1/Fri Jan 08 12:37:10 1993

Confidential property of Sybase, Inc.
(c) Copyright Sybase, Inc. 1987, 1993
All rights reserved.

Use, duplication or disclosure by the U.S. Government is subject to
restrictions set forth in FAR subparagraphs 52.227-19(a)-(d) for
civilian agency contracts and DFARS 252.227-7013(c)(1)(ii) for
Department of Defense contracts. Sybase reserves all unpublished
rights under the copyright laws of the United States.

Sybase, Inc. 6475 Christie Avenue, Emeryville, CA 94608, USA.
```

Más específicamente, este es un DBMS relacional para esos ambientes (sistemas operativos); en otras palabras, este es un sistema que permite cualquier número de usuarios (usuarios finales o programadores de aplicaciones o ambos) para acceder cualquier número de bases de datos relacionales⁸ por medio del lenguaje de SYBASE. El SQL Toolset es un conjunto de interfases para los usuarios finales y programadores de aplicaciones en el SQL Server. Ellos pueden comunicarse con cualquier número de SQL Servers y pueden manipular datos vía el lenguaje relacional de SYBASE.

El lenguaje relacional de SYBASE es conocido como Transact-SQL; esta es una versión de lo que conocemos como lenguaje relacional SQL, con numerosas extensiones destinadas a soportar el procesamiento transaccional. SQL es el lenguaje de bases de datos que soportan la mayoría de las DBMS relacionales hoy en día. Un dialecto de SQL fue adoptado por la **American National Standards Institute (ANSI)** en 1986 y por la **International Standard Organization** para la estandarización en 1987 como un estándar oficial para los sistemas relacionales.

Como podemos ver el uso del DBMS relacional de SYBASE nos serviría para poder manejar la información de nuestras historias académicas como una base de datos que se podría manejar fácilmente con el SQL y sus utilerías propias, con la ventaja de ser estándar (el SQL), mientras que el DBMS es bastante robusto y conocido en el mercado, nos evitaría el manejar un indexamiento de la información para después buscar un presentador de la misma; ahora bien su desventaja es que el DBMS no soporta tener dispositivos de almacenamiento de solo lectura, como lo sería el CD-ROM.

⁸ Una base de datos relacional es una base de datos que es percibida por sus usuarios como una colección de tablas.

WAIS (Wide Area Information Server)

La distribución del conocimiento fue primero hecha a través de la memoria humana y la tradición oral, después por manuscritos y, posteriormente por libros. Actualmente los medios impresos son la forma más común de difundir información, sin embargo los medios electrónicos han ganado terreno en lo que respecta a la distribución de la misma. WAIS es proyecto que intenta hacer, como técnica de distribución, a las redes de computadoras, explotando nuevas tecnologías y estándares.

Un protocolo abierto para conectar interfaces de usuarios en workstations y servidores es crítico para la expansión de servidores de información disponibles. El éxito de este sistema radica en una cantidad crítica de usuarios y servidores. Este protocolo, entonces, podrá ser usado en cualquier red electrónica desde redes digitales hasta redes por línea telefónica.

Para los dueños de la información el poner sus datos disponibles en una serie de servidores será fácil de iniciar, a bajo costo de operación, y provechoso. Una posibilidad será el de proveer software a bajo costo que pueda ayudar a esos poseedores de información a poner sus datos en redes electrónicas. El poder de las **workstations** es bastante para disponer de sofisticadas capacidades de servicios de información. En esta vía, es fácil de configurar, operar y hacer cargos por servicios de información.

Las ideas clave del sistema WAIS son que los servicios de información deberán ser fácil y gratuitamente distribuidos, que el poder de las **workstations** actuales puedan proveer de sofisticadas herramientas como servidores y clientes,

y que las redes electrónicas deberán ser explotadas para distribuir información. La meta del proyecto WAIS es crear una arquitectura abierta de servidores de información y clientes por derivación y estandarización de un protocolo computadora-a-computadora que permita a los usuarios encontrar y cuestionar servidores.

Lo anterior son las ideas generales y los alcances totales de los Servidores de Información de Area Amplia (WAIS), para nosotros sería útil una pequeña parte: el indexador, es decir, lo utilizaríamos como una herramienta de indexamiento de la información para poderla grabar entonces en el CD-ROM y después con otra herramienta poder presentar esa información al usuario final.

GOPHER

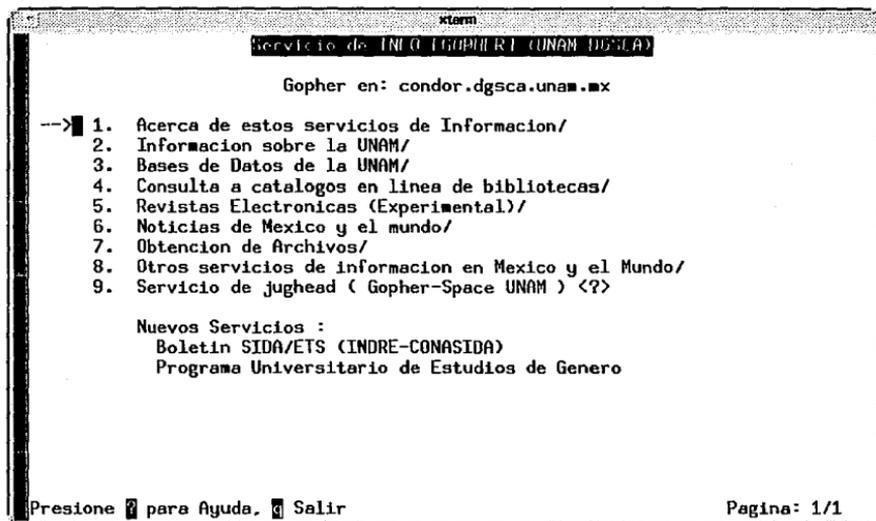
Alrededor del mundo, los datos son almacenados en computadoras, muchas de las cuales están conectadas por la Internet. Con el Internet Gopher podemos fácilmente acceder a información de acceso público almacenada en muchas de esas computadoras.

Gopher combina características del servicio de boletín electrónico y bases de datos, dejándonos una jerarquía de documentos, o búsqueda por documentos que contengan ciertas palabras o frases.

El software del Internet Gopher fue concebido en el departamento de Cómputo y Servicios de Información de la Universidad de Minnesota. El software es propiedad de la Universidad y es distribuido gratuitamente para fines no comerciales.

Gopher soporta un rango muy diverso de datos, todos los cuales pueden ser accedidos por un simple teclado o un click del mouse.

A continuación se presenta una pantalla del Gopher del servidor ubicado en DGSCA condor.dgsca.unam.mx



Es decir, Gopher nos proporciona una forma de encontrar y visualizar información por medio de menús, para lo cual se auxilia de otras herramientas como WAIS, ya que necesita que para la búsqueda de la información esta se encuentre indexada.

Otra ventaja es que Gopher no distingue entre dispositivos de lectura/escritura y los de solo lectura (CD-ROM) para poder acceder a la información.

MOSAIC

El Centro Nacional para aplicaciones de supercómputo de la Universidad de Illinois está desarrollando un ambiente para habilitar redes de área amplia basadas en el descubrimiento y recuperación de información y un trabajo en conjunto asíncrono. El ambiente, llamado NCSA Mosaic (National Center for Supercomputing Applications Mosaic), provee un portal dentro de un amplio universo basado en redes e información separada, y cualquier información accesible para el Mosaic -incluyendo información que no esta bajo el control directo de Mosaic o en servidores tan simples como para soportar actividades basadas en colaboraciones complejas- puede ser una parte de un ambiente de colaboración asíncrona de un usuario.

Mosaic usa una mezcla de hipertexto distribuida, organización jerárquica y búsquedas funcionales para el descubrimiento y recuperación de información. En adición, Mosaic esta diseñado para ser completamente capaz de relacionarse con datos científicos como componente estándar dentro de ese espacio de información. Mosaic esta diseñado para ser completamente distribuido a través de redes de área amplia, en parte soportando una amplia variedad de protocolos y formatos de información comunes y flexibles y proveyendo simples y convenientes métodos para expandir el dominio de información sobre el cual Mosaic puede operar.

Mosaic provee una interfaz unificada de varios protocolos, formato de datos y archivos de información usados en la Internet y habilita nuevos y poderosos métodos de descubrimiento uso y contribución de información⁹.

⁹ Mosaic esta basado en la tecnología WWW (World Wide Web) de la CERN en Suiza y usa la librería común del cliente de WWW para

Mosaic usa un modelo cliente/servidor para la distribución de la información - un servidor es una maquina en la Internet que atiende las peticiones enviadas por los clientes de Mosaic localizados en cualquier parte de la Internet.

Las unidades de información enviadas desde los servidores hacia los clientes son llamadas simplemente documentos. Los documentos pueden contener texto plano, texto formateado, gráficas, sonidos y otro tipo de datos multimedia, así mismo hiperligas a otros documentos que pueden estar localizados en cualquier parte de la Internet.

Mosaic nos serviría, como hemos visto, como un recuperador y visualizador de información (como Gopher) pero con la gran ventaja de ser mas completo, ofreciendo mas servicios, y sobre todo mas amigable al ser totalmente gráfico, ya sea en workstations o por medio de Windows en cualquier PC/DOS.

En seguida se presenta la pantalla de MOSAIC bajo Solaris 2.2

NCSA Mosaic: Document View

File Options Navigate Annotate Help

Document Title: NCSA Mosaic Home Page

Document URL: <http://www.ncsa.uiuc.edu/SDG/Software/Mosaic/>



Welcome to NCSA Mosaic, an Internet information browser and World Wide Web client. NCSA Mosaic was developed at the National Center for Supercomputing Applications at the University of Illinois, Urbana-Champaign.

Each highlighted phrase (in color or underlined) is a hyperlink to another document or information resource somewhere on the Internet. *Single click* on any highlighted phrase to follow the link.

The following resources are available to help introduce you to cyberspace and keep track of its growth:

- The NCSA Mosaic Demo Document
- Suggested Starting Points for Internet Exploration
- What's New with NCSA Mosaic and the Internet
- Internet Resources Meta-Index at NCSA

Special Notices

Back Forward Home Reload Open... Save As... Clone New Window Close Window

DESARROLLO DE UN SOFTWARE DE INDEXAMIENTO.

La compañía CROMASOFT la cual ofrece un soporte técnico a la DCAA en lo que respecta a los equipos SUN, está desarrollando un software de indexamiento de información, este software permite hacer búsquedas booleanas de algún ítem específico sobre un ambiente UNIX.

Sin embargo, el software no tiene la documentación necesaria y debido a que esta en una etapa de desarrollo tiene algunos detalles que no permiten su implementación a corto plazo.

2.3 Selección de la propuesta.

De acuerdo a los análisis y estadísticas efectuadas en capítulos anteriores, el medio de almacenamiento que reemplazará a la microfilmación será el CD-ROM. Los factores que se consideraron para la selección de esta alternativa son los siguientes:

- Existencia de estándares en el formato de grabación y reproducción.
- Plataforma instalada de unidades de reproducción.
- Información a almacenar de sólo consulta.
- Facilidad de distribución.
- Durabilidad del medio de almacenamiento.

EXISTENCIA DE ESTÁNDARES EN EL FORMATO DE GRABACIÓN Y REPRODUCCIÓN.

Como se ha venido mencionando en el capítulo anterior la industria del disco compacto ha sido una de las más exitosas en términos comerciales en la última década. Su éxito se debió más que a la innovación de la tecnología, a que desde un principio se definieron estándares en los formatos de grabación y unidades de reproducción. Todas estas especificaciones definidas por Phillips y Sony constituyeron lo que se denominó como **Red Book**. Esta especificación fue definida de una manera brillante, robusta y completa y fue lo que despertó el interés de fabricantes de hardware y de la industria de la música.

Como una aplicación orientada hacia el campo de la computación, surge en 1985 el CD - ROM, el cual permitió tener a un bajo costo una gran capacidad de almacenamiento. El CD - ROM no sólo heredó del CD algunas de sus características físicas, tales como la composición del disco, métodos de grabación y reproducción. También desde el principio se realizaron esfuerzos por generar estándares en cuanto a las características de hardware y a la información. De esta manera surgieron los estándares denominados **Yellow Book** e **ISO-9660**. La especificación Yellow Book es semejante a la del CD pero con algunas nuevas cosas como la corrección de errores. El estándar **ISO-9660** se refiere al formato de los archivos para que pueda ser recuperados desde diferentes plataformas, sin embargo en su momento todavía no quedó perfectamente especificado y fue hasta el año de 1990 con el nacimiento de las propuestas denominadas **Rock Ridge**, que hubo absoluta compatibilidad con los sistemas operativos UNIX y DOS.

PLATAFORMA INSTALADA DE UNIDADES DE REPRODUCCIÓN.

En los últimos años se ha popularizado el uso del CD - ROM como medio de distribución de información, almacenamiento de grandes bases de datos referentes a todos los tópicos, catálogos de bibliotecas, manuales técnicos y como medio de distribución de software. Este gran auge de aplicaciones y títulos disponibles en CD - ROM aunado al decremento cada vez mayor en los precios de las unidades de reproducción han provocado que la plataforma instalada tanto en equipos pequeños como en estaciones de trabajo se incremente cada vez más. Siendo esta razón un factor importante por el que se eligió el CD - ROM como medio de almacenamiento de información referente a la administración escolar.

Se presentan dos alternativas de solución: por un lado se puede pensar en tener cierta información centralizada, la cual podrá ser accesada a través de RedUNAM mediante la utilización de una clave de acceso a un servidor, el cual tendrá conectado un *juke box* de CD - ROMs, que no es más que lo que conocemos como una "rockola" de discos que de acuerdo a la petición del usuario seleccionará el disco que contiene la información requerida.

La otra alternativa consiste en la distribución de información, es decir, proporcionar a las dependencias usuarias una copia de su información para que la puedan manejar de manera local y de esta manera descentralizar el manejo de la administración escolar.

INFORMACIÓN A ALMACENAR DE SÓLO CONSULTA.

Debido a la naturaleza confidencial de la información que estamos manejando, como es el caso de historias académicas, resulta bastante deseable que el dispositivo de almacenamiento a utilizar no permita de ninguna manera la alteración de la información al ser consultada.

FACILIDAD DE DISTRIBUCIÓN.

Uno de los problemas que presentaba el uso del CD - ROM era el elevado costo que representa el proceso de grabación y reproducción de copias. Este proceso en la actualidad sólo se realiza en los Estados Unidos a través de compañías especializadas en esta actividad. El proceso de realización del disco maestro es el que tiene el mayor costo, de aproximadamente \$1500 U.S. y cada copia cuesta aproximadamente \$2 U.S. Dado que la aplicación que estamos realizando no requiere de grandes volúmenes de copias el costo de hacer la grabación de un disco para generar cuando mucho 5 copias resultaba demasiado caro. Sin embargo a finales de 1992 se liberó en el mercado la tecnología CD - R que permite al usuario hacer la grabación de su CD - ROM en su mismo lugar de trabajo sin necesidad de todas las condiciones ambientales que se requieren para el proceso.

Realizando un análisis de todos los equipos disponibles en el mercado se llegó a la decisión de utilizar la solución ofrecida por Young Minds, debido a que era el único que trabajaba bajo las plataformas UNIX y DOS.

UNIDAD DE GRABACIÓN CD-ROM

Este dispositivo no forma parte del equipo SUN, sin embargo, dado que es un elemento de hardware fundamental en el desarrollo de este trabajo y estamos trabajando con él en ambiente UNIX, procederemos en esta sección a la descripción del equipo.

Como se mencionó anteriormente la tecnología del CD-R nos permite a un bajo costo la producción de bajos volúmenes de discos compactos en "nuestro escritorio". Actualmente se tienen en el mercado una serie de equipos de grabación y software para realizar premaster/master. En el capítulo "propuesta de solución" se expondrán en forma más detallada las razones por las que se eligió el equipo que a continuación describiremos.

El equipo con el que cuenta la DCAA es la solución denominada CD-Studio de la compañía Young Minds. Este paquete consta del grabador Phillips CDD 521, una controladora y el software de premastering/mastering MakeDisc.

Este sistema trabaja bajo los sistemas operativos UNIX y DOS y su principal característica es la inserción de una controladora cuya función es la de encargarse de la transmisión de la información hacia el grabador, liberando de esta carga al sistema de I/O de la máquina. Este controlador es en esencia una PC sin monitor ni teclado, con 2 adaptadores del tipo SCSI de alto rendimiento y un disco de 1 GB. Uno de los adaptadores se encarga de la comunicación con la máquina UNIX, el otro se comunica con el grabador.

El proceso de grabación es bastante sencillo y más con la ayuda de la interfase gráfica que ofrece MakeDisc. Ya sea a través de ella o de la línea de comandos se le indica la estructura de archivos de la que va a realizar la imagen y la unidad de destino. Esta unidad de destino es el disco duro de la controladora, la cual es vista por Unix como una unidad de cinta. Una vez que se realiza este proceso, el sistema nos permite hacer una prueba de la imagen antes de proceder a la grabación en el disco virgen.

En lo que respecta al software que se utilizará para dicho propósito (consulta masiva de la información) se tienen las siguientes opciones:

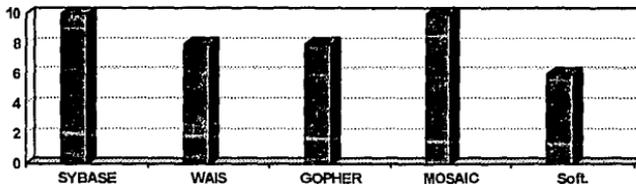
- SYBASE
- WAIS
- GOPHER
- MOSAIC
- Software de indexamiento desarrollado por CROMASOFT.

Los factores que se consideraron para la elección de alguno de ellos fueron los siguientes:

- Ambiente amigable, tanto para el usuario como para el administrador del sistema.
- Facilidad de realizar modificaciones para adaptarlas según las necesidades.
- Costo.
- Soporte y documentación.
- Facilidad en el manejo del medio de almacenamiento, en este caso CD-ROM.

A continuación se presentan gráficas comparativas basadas en el análisis de cada una de estas herramientas, según nuestra experiencia con ellas y según las necesidades del sistema a implementar. Las calificaciones están dadas del 1 al 10, siendo 1 la calificación más baja y 10 la más alta.

Ambiente amigable.



En el caso de un ambiente amigable SYBASE proporciona el mejor ambiente junto con MOSAIC, estas dos herramientas presentan la facilidad de una interfaz **GUI**. Una de las ventajas que de MOSAIC es que la interfase que da al usuario puede programarse de una manera muy simple. SYBASE también da esta posibilidad, sin embargo la da a través de clientes, es decir programas en Visual Basic, C++, etc. que accesan a la información y la presentan al usuario.

WAIS y GOPHER tienen una interfaz **CUI (Character User Interface)** que no es tan intuitiva como las anteriores pero no por ello son peores.

El software de indexamiento da la peor interfaz, debido a que las búsquedas de información y la recuperación de la misma las arroja a la pantalla; se tendría que programar alguna CUI para presentar la información.

FACILIDAD DE MODIFICACIONES SEGÚN NECESIDADES.

En este punto se considera la facilidad que da la herramienta de hacerle modificaciones para adaptarla a nuestras necesidades; se consideró que la herramienta debe de adaptarse al problema y no al contrario.

Considerando a SYBASE, este no presenta ninguna posibilidad de adaptación, sino al contrario, se tiene que modelar el sistema de consulta de historias académicas como una base de datos relacional, programar una interfaz para mostrar la información así como la petición de las consultas. A la vista resaltan entonces dos problemas: un modelado de bases de datos y programación extra.

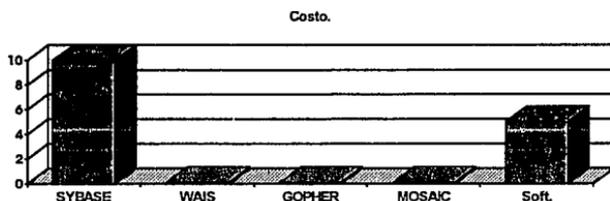
La ventaja que dan WAIS, Gopher y MOSAIC es la de ofrecer el código fuente de dichas herramientas, es decir, en la red Internet estas herramientas son de dominio público, y para poder obtener un programa ejecutable en los diferentes tipos de plataformas (Solaris 2.2 en nuestro caso) se necesitan los fuentes para poder compilarlos y así poder ejecutarlos. Debido a que se cuenta con el código, se pueden hacer modificaciones superficiales o de fondo según se requiera.

Además por su naturaleza, estos programas son los apropiados para el manejo masivo de información: el indexamiento de ésta así como su consulta la ofrece WAIS, Gopher da un **front-end** amigable al usuario además de poder acceder otros servicios de Gopher en la red y MOSAIC finalmente, es un ambiente gráfico, en el cual se pueden visualizar imágenes, gráficas, películas además de sonido.

El software de Cromasoft se puede considerar que es modificable según necesidades, sin embargo se tendría que negociar con la empresa la adaptación

de esta herramienta a nuestro problema y de esperar algún tiempo a que ésta se desarrollara.

COSTO.



Un punto crítico de la selección de la herramienta es su costo, sin embargo, como se puede observar en la gráfica este punto dentro del análisis resultó muy simple: SYBASE es la aplicación más costosa, aunque la DCAA ya contaba con esta herramienta. WAIS, Gopher y MOSAIC son herramientas de dominio público que son **totalmente gratuitas** y no se debe de pagar nada en absoluto por su uso. El software de Cromasoft debido a que tiene que ser desarrollado por esta empresa, se tenía que pagar un cierto precio por ella.

SOPORTE Y DOCUMENTACIÓN.

El soporte y la documentación de todas las herramientas es muy bueno, se puede obtener mucha bibliografía al respecto además de poder obtener muchos consejo de la comunidad de la Internet para cualquiera de éstas. La excepción fue

el software de Cromasoft, el cual no contaba con una buena documentación aunque si de un buen soporte por parte de la empresa.

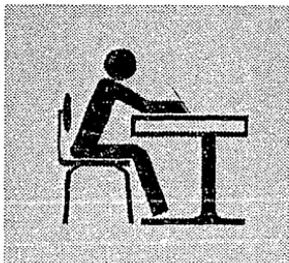
FACILIDAD DEL MANEJO DE LA INFORMACIÓN EN CD-ROM.

El punto decisivo para la elección de alguna de las herramientas fue la facilidad que da en el manejo de la información en CD-ROM. SYBASE *no permite el manejo de información grabada en CD-ROM*, mientras que las demás si lo pueden hacer.

En la siguiente tabla hacemos un resumen de las características a considerar:

	SYBASE	WAIS	GOPHER	MOSAIC	Software CROMASOFT
Ambiente agradable.	SI	SI	SI	SI	NO
Modificaciones según necesidades.	NO	SI	SI	SI	SI , se debe de negociar.
Costo.	ALTO	Ninguno	Ninguno	Ninguno	Sujeto a negociación.
Soporte y documentación.	BUENO	BUENO	BUENO	BUENO	REGULAR
Facilidad de manejo de información en CDROM.	No, SYBASE no puede manejar información en estos medios.	Sí puede manejar información en este medio.			

La elección de las herramientas es obvia: se seleccionó WAIS, GOPHER y MOSAIC para la implementación del sistema de consulta de las historias académicas en CD-ROM. Otro factor que influyó para la elección de estas herramientas fue que en la actualidad están definiendo un nuevo estándar en lo



Capítulo III.

Diseño de la propuesta.

Este es el capítulo central del presente trabajo, ya que en él se desarrollará el Sistema de Información Académica (SINAC).

Para esto se hace uso de recursos con los que ya se contaban, como el mainframe A12 Unisys y recursos recientes como el equipo SUN SparcCenter 10 y la unidad de grabación de CD-ROM Phillips CDD 521, así como de nuevas aplicaciones como lo son WAIS, Gopher y Mosaic.

Finalmente se presentan las perspectivas de desarrollo que este sistema tendrá dentro de la UNAM, no solo en la administración escolar sino en muchas otras áreas.

1.

Migración de la información al ambiente UNIX.

Debido a que la generación de la información de las historias académicas se lleva a cabo en el mainframe A-12 de Unisys, el cual lo deja en un formato nativo, es necesario primero: depurar esa información para después pasarla hacia el ambiente UNIX, donde se terminará de dar formato para que sea útil a las aplicaciones que se van a utilizar (WAIS principalmente).

1.1 Metodología.

En este punto se explicarán los dos programas que se utilizaron para migrar la información de un ambiente propietario como lo es el MCP de Unisys, hacia el ambiente UNIX, que es donde se desarrollará el Sistema de Información Académica (SINAC). En el ambiente MCP se desarrollará en el lenguaje de programación ALGOL y en el ambiente UNIX en la herramienta AWK.

1.1.1 El lenguaje de programación ALGOL en el equipo A12 de UNISYS.

ALGOL, que proviene de las palabras ALGOrithmic Lenguaje (Lenguaje Algorítmico), es exactamente lo que implica su denominación: un lenguaje para la

expresión de algoritmos, es decir, totalmente estructurado; es más podemos atrevernos a decir que fue el primer lenguaje estructurado. Fue desarrollado entre 1957 y 1960 por un grupo internacional de especialistas en computación. Aún se sugieren modificaciones y mejoras, conforme se van actualizando las versiones del sistema operativo MCP/AS, pero hoy en día el lenguaje está más o menos establecido como fue definido en el número de mayo, 1960 de *Communications* de la Asociación de Máquinas Computadoras, en el artículo "Report on the Algorithmic Language ALGOL 60", editado por Peter Naur. El número "60" distingue al lenguaje presente de uno preliminar, a veces llamado ALGOL 58 y de versiones posteriores que puedan, con el tiempo, ser lo bastante diferentes para merecer una denominación propia. Esto significa que desde que salió a la luz pública ese artículo no ha habido modificaciones sustanciales a la estructura del lenguaje.

ALGOL esta destinado como un lenguaje de propósito general, es decir tanto para aplicaciones científicas como comerciales. La importancia del lenguaje **ALGOL** en los equipos Serie A es que se puede considerar que este lenguaje es el **lenguaje ensamblador** del equipo, así es, la Serie A no tiene ningún tipo de lenguaje ensamblador como tal sino que el lenguaje de mas "bajo nivel" que existe es precisamente ALGOL el cual es un lenguaje estructurado y de alto nivel. El lenguaje ALGOL es sumamente poderoso en este tipo de equipo, basta mencionar que gran parte del sistema operativo MCP/AS esta escrito totalmente en ALGOL o en alguna de sus variantes¹ y la otra parte esta escrita en ESPOL el cual está basado en ALGOL.

¹ Existen algunas variantes de este lenguaje como son DCALGOL (Data Communications ALGOL) y DMALGOL (Data Manager ALGOL) los cuales son para el manejo de comunicaciones de teleproceso y red y el manejo e interfaz con DMS II (manejador de bases de datos del equipo).

Con ALGOL es posible hacer llamadas directas al sistema, interrupciones, etc. pero además, si al programa se le da el estatus de privilegiado se pueden hacer llamadas directas al hardware sin pasar por el sistema operativo en si, además de muchas de las utilerías del sistema así como otros lenguajes como lo son NEWP². En conclusión reiteramos que en ésta plataforma de equipo no hay lenguaje mas poderoso que ALGOL.

Solamente a manera de información podemos decir que ALGOL está en la actualidad confinado a los equipos UNISYS ya que casi ningún otro equipo lo soporta o al menos no con sus grandes ventajas, pero esto solo sucede en América ya que en Europa sigue siendo un lenguaje muy utilizado en aplicaciones científicas. Podemos mencionar además que ALGOL es el "padre" de un lenguaje mucho más popular que es PASCAL. Este lenguaje es una versión reducida y práctica, pero menos poderosa, de lo que es ALGOL, con la facilidad de implementarse en más equipos desde PC's hasta supercomputadoras.

1.1.2 Especificación del programa en ALGOL.

Las especificaciones del programa se basaron en las necesidades del formato que debería de tener nuestra información antes de poder pasarla vía red hacia el equipo SUN.

Antes ya se había mencionado el ciclo que se lleva a cabo para poder tener la información actualizada semestre por semestre de las calificaciones de los

² Lenguaje para la creación de otros lenguajes, compiladores y hasta sistemas operativos.

alumnos de las Facultades y Escuelas de la UNAM³. Ahora sólo nos falta decir cuales son las etapas de este ciclo que se seguirán utilizando con este nuevo sistema que propusimos y realizamos.

Las etapas que se siguen utilizando son, desde el momento de la inscripción de los alumnos hasta que se realiza la impresión de las historias y el respaldo en cinta del **backup** generado. Una vez que se ha respaldado este último se convierte en la fuente de información a utilizar por el nuevo sistema pero como es un archivo generado por el sistema operativo del equipo A12, el cual tiene caracteres de control bien definidos, que serían solo "basura" en el equipo SUN, es decir datos que no son utilizables; por lo anterior todos estos caracteres de control se eliminaron por medio de un programa en ALGOL, lenguaje que por las características que posee es el que mejor cumplía con los requerimientos.

Se decidió hacerlo antes de transportar el archivo hacia el equipo SUN debido a que el depuramiento no representaría una carga efectiva importante de trabajo en el mainframe de UNISYS además de que sería más fácil reconocer los caracteres de control en su medio natural, que los exportados y que podrían variar en el equipo SUN.

A continuación pasemos a describir como se encuentra la información en el archivo original. La información como ya dijimos se encuentra en un formato el cual no es entendible ni desplegable en pantalla por el número de caracteres de control que posee además del número total de caracteres por registro que tiene, el cual es de 1800 caracteres por registro. Para poder tener acceso a este archivo es necesario hacerlo por medio de una utilidad del sistema operativo MCP/AS de

³ El ciclo completo se explica en el tema 2.1.1 del capítulo II, Condiciones Actuales del Sistema.

la serie A llamada SYSTEM/DUMPALL. Esta utilería permite manejar archivos que no tienen los atributos por default para poder ser listados normalmente por CANDE⁴. Es importante señalar que no es necesario usar la utilería DUMPALL para poder utilizar el archivo dentro del programa, es decir, se pueden hacer llamadas y lecturas de registros directas ya que sólo se maneja la información siendo que lo que causa conflicto es el querer desplegar la información ya sea en pantalla, en archivo de trabajo local de la terminal o querer mandar a imprimirlo sin estar la impresora en el formato correcto para ello.

Después de correr esta utilería y de pedir que nos fueran impresos algunos registros para poder ver cual era la estrategia a utilizar para su depuramiento obtuvimos el listado que se presenta a continuación.

```

.....
.....dr.....m.....XEROX HIST ACAD 1 TA.....HAPROD.....
.....U-H-A-M.....HA-941-208-AC-
941.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....XEROX HIST ACAD 1 TANTO.....STA17002   CANDE.1.....
.....
08/ABR/94 ..... .. ... ..
VELAZQUEZ ISRAEL ..... 9362639-4   1 1 ..... ABUNDIZ
E.N.E.P. ACATLAN (ECONOMIA) ..... 208 93.....
LIC EN ECONOMIA ..... 21 94-1 .. ...
.. ... 72 300 24.00 ..... 13 .. ...
3 ..... 0 0 0.00 ..... 16 .....
72 300 24.00 ..... 7.53 .....
PRIMER SEMESTRE .. ... 208 0241 06 GEOGRAFIA ECONOMICA
B OR 93-1 1650718 1151 1 .. ... 208 0322 06 INTRODUCCION A LA

```

⁴ CANDE es un editor tanto de archivos como de comandos en el sistema MCP/AS. CANDE significa Command AND Edit.

Capítulo III Diseño de la propuesta.

```

ECONOMIA          B OR 93-1 1650745 1151 1    .. ... 208 0361 06
ECONOMIA POLITICA I          S OR 93-1 1650753 1151 1    ..
... 208 0368 04 CIES I
1650775 1151 1    ..... 208 0497 08 MATEMATICAS I
NA EX 94-1 1775914 EA51 2 2
.....
.. ... 208 0291 06 HISTORIA ECONOMICA GENERAL I          S OR 93-1
1650726 1151 1    ..... SEGUNDO SEMESTRE ..
... 208 0292 06 HIST ECONOMICA GENERAL II          NA EX 93-2
1718183 EA51 1 1 .. ... 208 0362 06 ECONOMIA POLITICA II
S OR 93-2 1710573 2251 1    .. ... 208 0369 04 CIES II
B OR 93-2 1710591 2251 1    .. ... 208 0500 06 METODOLOGIA DE LAS C
SOCIAL          B OR 93-2 1710615 2251 1    ..... 208 0755 06
SOCIOLOGIA Y POLITICA I          B OR 93-2 1710621 2251 1
..... TERCER SEMESTRE .. ... 208 0293 06
HIST ECON DE AMERICA LATINA          B OR 94-1 1765636 1352 1    ..
... 208 0363 06 ECONOMIA POLITICA III          S OR 94-1
1765664 1351 1    .. ... 208 0370 04 CIES III
B OR 94-1 1765691 1352 1    .. ... 208 0756 06 SOCIOLOGIA Y POLITICA
II          MB OR 94-1 1765717 1351 1    ..... 208 0814 06
TEORIA ECONOMICA I          NP EX 94-1 1775929 EA51 1 1
..... 20..... 20..... 20..... 20.....
20..... 20..... 20..... 20..... 20..... 20.....
20..... 20..... 20.....

```

Como se puede observar en primer lugar aparecen los atributos del archivo los cuales son importantes para poder hacer un manejo efectivo de él a la hora de programar. A continuación se presenta el contenido del archivo registro por registro; en este caso pedimos que nos fuera presentada la información tanto en EBCDIC (el cual es el código de la máquina A-12), en ASCII y en modo hexadecimal. Lo anterior debido a que EBCDIC nos permitirá saber que caracteres que son información útil, pero al momento de que aparezca un caracteres de control este será representado en la impresión con una interrogación (?), es ahí cuando entra la función de la representación en hexadecimal ya que nos permite saber que carácter de control es (por ejemplo retorno de carro, centrado, sangría, nulos, etc.). De los caracteres de control los que nos sirven para el depuramiento son: 06 en hex. que significa un retorno de carro y para nosotros representa que es una nueva línea; y el 80 en hex. es nulo y que nosotros interpretamos como un espacio en blanco.

Cabe señalar que en el primer registro los primeros 978 caracteres son de lo que se llama un *header* es decir información sobre la impresión como el formato, el usuario que lo mando imprimir, etc. el cual definitivamente no nos interesa. Por esto es que en la primer iteración, que es cuando se lee el primer registro, nos saltamos estos 978 caracteres que no nos son útiles y empezamos la depuración formal a partir del carácter 979; para los demás registros se hizo la depuración por completo de cada uno de ellos.

Explicemos pues la filosofía básica que inspiró el programa: se lee cada uno de los registros y se almacenan en un arreglo, a continuación se va leyendo carácter por carácter y se compara cada uno con un conjunto definido de caracteres alfanuméricos⁵; si pertenece a éste, entonces se inserta en otro arreglo de "información depurada" y si no lo es se salta y lee el siguiente. Si se encuentra el carácter de control que significa nulo el cual en hexadecimal es 80 se inserta como un blanco para que al final la información quede lo mas "alineada" posible; así también cuando se encuentra el carácter de control que indica el retorno de carro para la impresora nosotros lo interpretamos como el fin de línea y es cuando se escribe esta línea al nuevo archivo (archivo que contendrá la información depurada) después de esto se limpia el arreglo de la "información depurada" y se sigue leyendo el registro o se seguirá leyendo otro en caso de que el anterior registro se haya ya terminado de depurar. El programa fuente se muestra a continuación:

⁵ Alfanuméricos son el abecedario tanto en mayúsculas como en minúsculas y los dígitos del 0 al 9 así como algunos caracteres especiales como lo son /,., y el espacio en blanco.

Capitulo III Diseño de la propuesta.

```
BEGIN          %INICIO DEL PROGRAMA
COMMENT       %DECLARACION DE UN COMENTARIO
*****
*
*   EL PRESENTE PROGRAMA QUITA TODOS LOS CARACTERES DE CONTROL DEL
*
*   ARCHIVO UTILIZADO PARA IMPRIMIR LAS HISTORIAS ACADEMICAS DE LOS
*
*   ALUMNOS. ES DECIR DEPURA EL ARCHIVO DEJANDO SOLO LA INFORMACION
*
*   UTILIZABLE.
*
*****
;
#####
%
#####          DECLARACION DE ARCHIVOS          #####
%
#####
%   DECLARACION DE ARCHIVO QUE CONTENDRA LA INFORMACION YA DEPURADA
#####
FILE DEPURADO(KIND=DISK,MAXRECSIZE=85,UNITS=CHARACTERS,
              PROTECTION=SAVE);

#####
%   DECLARACION DE ARCHIVO QUE CONTIENE LA INFORMACION SIN DEPURAR
#####

FILE ORIGINAL(KIND=DISK,MAXRECSIZE=300,BLOCKSIZE=300,UNITS=WORDS);

#####
%   DECLARACION DE LA PANTALLA DE LA TERMINAL
#####

FILE VIDEO(KIND=REMOTE);

#####
%
#####   DECLARACION DE UN CONJUNTO DE CARACTERES Y NUMEROS   #####
%
#####

TRUTHSET CARACTERES(ALPHA OR "1234567890/-.\ " );

#####
%
#####   DECLARACION DE ARREGLOS DE TIPO EBCDIC (CHARACTER)   #####
%
#####

EBCDIC ARRAY HEADER[0:1799];
EBCDIC ARRAY INFORMACION[0:5000];
EBCDIC ARRAY INF_DEPURADA[0:1799];
```

```

*****
%
%*****      DECLARACION DE VARIABLES DE TIPO ENTERO      %*****
%
%*****
INTEGER CONT1,CONAUX,CONT2;

*****
%
%***** PRINCIPIA CUERPO DEL PROGRAMA *****
%
%*****

CONT2:=0;          %SE INICIALIZA LA VARIABLE

FOR CONT1:=0 STEP 1 UNTIL 1799 DO          %SE LIMPIA EL ARREGLO HEADER
REPLACE HEADER[CONT1] BY " " FOR 1;

FOR CONT1:= 0 STEP 1 UNTIL 120 DO %SE LIMPIA EL ARREGLO INF_DEPURADA
REPLACE INF_DEPURADA[CONT1] BY " " FOR 1;

*****
%*****
% SE LEE EL PRIMER REGISTRO DEL ARCHIVO SIN DEPURAR QUE ES DONDE SE %
% ENCUENTRA EL HEADER Y SE ASIGNA AL ARREGLO LLAMADO HEADER (CARACTER%
% A CARACTER SIENDO EN TOTAL 1800) %
%*****
READ(ORIGINAL,*,HEADER[0]);

*****
%*****
% EL SIGUIENTE "FOR" REALIZA LA FUNCION DE QUE APARTIR DEL CARACTER %
% 979 LEE LOS DATOS Y SOLO ACEPTA COMO VALIDOS LOS QUE SE ENCUENTRAN %
% EN LA DEFINICION DEL TRUTHSET CARACTERES. AL ENCONTRAR EL CARACTER%
% 48"06" QUE REPRESENTA EL RETORNO DE CARRO, ESCRIBE LA INFORMACION %
% AL ARCHIVO DEPURADO AL TERMINAR LIMPIA NUEVAMENTE EL ARREGLO %
% PARA PODER VOLVER A UTILIZARLO. %
%*****

FOR CONT1:= 978 STEP 1 UNTIL 1799 DO
BEGIN
IF (HEADER[CONT1] = 48"06" FOR 1) THEN
BEGIN
WRITE(DEPURADO,<A1,A84>," ",INF_DEPURADA[*]);
CONT2:=0;
FOR CONAUX := 0 STEP 1 UNTIL 120 DO
REPLACE INF_DEPURADA[CONAUX] BY " " FOR 1;
END
ELSE
IF (HEADER[CONT1] IN CARACTERES) THEN
BEGIN
REPLACE INF_DEPURADA[CONT2] BY HEADER[CONT1] FOR 1;

```

Capítulo III Diseño de la propuesta.

```
CONT2:=CONT2+1;
END;
END;

CONT2:=0;

*****
@ EL SIGUIENTE "WHILE" FUNCIONA MIENTRAS NO SE ACABE EL ARCHIVO SIN @
@ DEPURAR; SI NO SE ACABA ENTONCES LEE EL REGISTRO CORRESPONDIENTE Y @
@ REALIZA LA FUNCION DE CONFIRMAR QUE EL CARACTER ESTE DENTRO DEL @
@ TRUTHSET CARACTERES. SI NO ESTA CONFIRMA QUE SEA EL CARACTER @
@ 48"06", EL CUAL ES UN RETORNO DE CARRO, PARA ENTONCES ESCRIBIR UNA @
@ LINEA EN EL ARCHIVO YA DEPURADO Y REPETIR TODO EL PROCESO DESDE @
@ LIMPIAR EL ARREGLO DE LA INFORMACION. EN CASO QUE ENCUENTRE UN @
@ 48"80", EL CUAL ES UN NULO, LO CAMBIA POR UN 48"40" QUE ES UN @
@ ESPACIO EN BLANCO. @
@ AL FIN DEL WHILE ESCRIBE LA ULTIMA LINEA DE INFORMACION. @
*****

WHILE NOT(READ(ORIGINAL,*,INFORMACION[0])) DO
BEGIN
FOR CONT1:=0 STEP 1 UNTIL 1799 DO
BEGIN
IF (INFORMACION[CONT1] = 48"06" FOR 1) THEN
BEGIN
WRITE(DEPURADO,85,INF_DEPURADA[*]);
FOR CONAUX:=0 STEP 1 UNTIL 120 DO
REPLACE INF_DEPURADA[CONAUX] BY " " FOR 1;
CONT2:=0;
END
ELSE
BEGIN
IF (INFORMACION[CONT1] IN CARACTERES) THEN
BEGIN
REPLACE INF_DEPURADA[CONT2] BY INFORMACION[CONT1] FOR 1;
CONT2:=CONT2+1;
END
ELSE
IF (INFORMACION[CONT1] = 48"80" FOR 1) THEN
BEGIN
REPLACE INF_DEPURADA[CONT2] BY 48"40" FOR 1;
CONT2:=**+1;
END;
END;
END;
END;
WRITE(DEPURADO,85,INF_DEPURADA[*]);
END.
```

Al final se obtiene un archivo ya depurado de los caracteres de control y que solo posee la información útil, del cual se presenta una parte a continuación.

				08/ABR/94		
ABUNDIZ VELAZQUEZ ISRAEL				9362639-4	1	1
E.N.E.P. ACATLAN ECONOMIA				208	93	
LIC EN ECONOMIA				21		94-1
72	300	24.00	13			
			3			
0	0	0.00	16			
72	300	24.00	7.53			
PRIMER SEMESTRE						
208 0241 06	GEOGRAFIA ECONOMICA			B OR 93-1	1650718	1151 1
208 0291 06	HISTORIA ECONOMICA GENERAL I			S OR 93-1	1650726	1151 1
208 0322 06	INTRODUCCION A LA ECONOMIA			B OR 93-1	1650745	1151 1
208 0361 06	ECONOMIA POLITICA I			S OR 93-1	1650753	1151 1
208 0368 04	CIES I			B OR 93-1	1650775	1151 1
208 0497 08	MATEMATICAS I			NA EX 94-1	1775914	EA51 2 2
SEGUNDO SEMESTRE						
208 0292 06	HIST ECONOMICA GENERAL II			NA EX 93-2	1718183	EA51 1 1
208 0362 06	ECONOMIA POLITICA II			S OR 93-2	1710573	2251 1
208 0369 04	CIES II			B OR 93-2	1710591	2251 1
208 0500 06	METODOLOGIA DE LAS C SOCIAL			B OR 93-2	1710615	2251 1
208 0755 06	SOCIOLOGIA Y POLITICA I			B OR 93-2	1710621	2251 1
TERCER SEMESTRE						
208 0293 06	HIST ECON DE AMERICA LATINA			B OR 94-1	1765636	1352 1
208 0363 06	ECONOMIA POLITICA III			S OR 94-1	1765664	1351 1
208 0370 04	CIES III			B OR 94-1	1765691	1352 1
208 0756 06	SOCIOLOGIA Y POLITICA II			MB OR 94-1	1765717	1351 1
208 0814 06	TEORIA ECONOMICA I			NF EX 94-1	1775929	EA51 1 1
						1
				08/ABR/94		
ACEVEDO FERRER INGRID				8906263-4	1	1
E.N.E.P. ACATLAN ECONOMIA				208	92	
LIC EN ECONOMIA				21		94-1
174	300	58.00	30			
			0			
0	0	0.00	30			
174	300	58.00	8.80			
PRIMER SEMESTRE						
208 0241 06	GEOGRAFIA ECONOMICA			MB OR 92-1	1552721	1102 1
208 0291 06	HISTORIA ECONOMICA GENERAL I			MB OR 92-1	1552729	1102 1
208 0322 06	INTRODUCCION A LA ECONOMIA			MB OR 92-1	1552749	1102 1
208 0361 06	ECONOMIA POLITICA I			B OR 92-1	1552757	1102 1
208 0368 04	CIES I			MB OR 92-1	1552782	1102 1
208 0497 08	MATEMATICAS I			S OR 92-1	1552803	1102 1
SEGUNDO SEMESTRE						
208 0292 06	HIST ECONOMICA GENERAL II			B OR 92-2	1606372	2202 1
208 0362 06	ECONOMIA POLITICA II			MB OR 92-2	1606384	2202 1
208 0369 04	CIES II			B OR 92-2	1606398	2201 1
208 0498 08	MATEMATICAS II			B OR 92-2	1606420	2202 1
208 0500 06	METODOLOGIA DE LAS C SOCIAL			MB OR 92-2	1606426	2202 1
208 0755 06	SOCIOLOGIA Y POLITICA I			MB OR 92-2	1606431	2201 1
TERCER SEMESTRE						
208 0293 06	HIST ECON DE AMERICA LATINA			B OR 93-1	1650731	1302 1
208 0363 06	ECONOMIA POLITICA III			B OR 93-1	1650758	1302 1
208 0370 04	CIES III			MB OR 93-1	1650783	1352 1
208 0499 08	MATEMATICAS III			MB OR 93-1	1650800	1301 1
208 0756 06	SOCIOLOGIA Y POLITICA II			MB OR 93-1	1650806	1301 1
208 0814 06	TEORIA ECONOMICA I			MB OR 93-1	1650816	1352 1
CUARTO SEMESTRE						
208 0004 04	ANALISIS D EDOS FINANCIEROS			B OR 93-2	1710531	2452 1
208 0164 08	ESTADISTICA I			B OR 93-2	1710548	2451 1
208 0294 06	HIST ECONOMICA D HEXICO I			S OR 93-2	1710566	2451 1
208 0364 06	ECONOMIA POLITICA IV			B OR 93-2	1710580	2452 1
208 0371 04	CIES IV			B OR 93-2	1710599	2453 1
208 0815 06	TEORIA ECONOMICA II			B OR 93-2	1710627	2451 1

QUINTO SEMESTRE							
208	0099	04	CONTABILIDAD SOCIAL	B	OR	94-1	1765590 1551 1
208	0165	08	ESTADISTICA II	B	OR	94-1	1765602 1552 1
208	0295	04	HIST ECONOMICA D MEXICO II	MB	OR	94-1	1765642 1552 1
208	0365	06	ECONOMIA POLITICA V	MB	OR	94-1	1765669 1551 1
							2

Para finalizar cabe mencionar que por el tiempo que tarda en la depuración de cada uno de los archivos, dependiendo de su tamaño, es mejor mandar correr el programa a través de WFL (Work Flow Lenguaje) el cual es una herramienta para el control del flujo de trabajos, y que nos permite tener más de una tarea al mismo tiempo sin tener que estar iterativamente verificando su funcionamiento. Este **job** se presenta en seguida:

```
BEGIN JOB;
JOBSUMMARY=UNCONDITIONAL;
JOBSUMMARYTITLE=PRUEBA;
CLASS=50;
RUN (DASU)OBJECT/EL_DEPURADOR ON DCAA333;
END JOB.
```

1.1.2.1 Portabilidad del Programa y Ambiente de Programación.

Por las características que posee el lenguaje ALGOL dentro de los equipos de la Serie A de UNISYS es sumamente portable entre equipos de la misma familia no importando el tamaño, la capacidad o la antigüedad del equipo, así por ejemplo en nuestro caso el programa lo mismo puede correr en el equipo A12 que en el equipo A12-B, es decir, son totalmente portables y compatibles entre equipos. Su única limitante sería que se respetaran las versiones del sistema operativo bajo el cual estuviera corriendo y por tanto la versión del compilador que estuviera actuando, aunque este escollo se puede salvar recompilando el fuente del programa en la versión del compilador ALGOL que correspondiera; a futuro y con respecto a esto último podemos decir que la recompilación mínima tendría que suceder cada dos cambios de versión de MCP ya que por ejemplo la

versión actual del MCP es la 4.0.2 pero puede correr programas compilados desde la versión 3.8 del sistema operativo, así que el ejecutable de este programa será válido hasta la versión 4.2 del MCP teniendo que recompilar el fuente para crear otro ejecutable a partir de la versión 4.3 del MCP.

En cuanto al ambiente de programación se había ya mencionado que es bajo el editor de texto y comandos llamado CANDE el cual es estándar en la Serie A. Ahora bien el acceso al mainframe puede ser tanto por teleproceso, es decir vía terminales propias de la máquina, las cuales aprovechan al máximo los potenciales del sistema en cuanto a sus funciones de edición, como a través de telnet⁶ por cualquier PC o conexión de otros equipos, el mismo equipo SUN, aunque con esto no se aprovechan al máximo sus características y se vuelve más complicada la edición y el manejo en sí del sistema; también existe un paso intermedio el cual es la emulación de una terminal del equipo en una PC por medio de una utilidad llamada Infoview la cual nos permite tener las mismas funciones que una terminal de teleproceso en una PC con las ventajas de manejar gráficos, colores, y hasta el mouse para su manejo, pero con la desventaja de tener que estar trabajando a través de la red pública Internet la cual a veces tiene problemas en cuanto al tráfico y la rapidez de respuesta del equipo.

1.1.3 Transferencia de datos al ambiente UNIX.

Una vez que se tiene la información semi-depurada en el mainframe **A12**, el siguiente paso es la transportación de ésta al ambiente UNIX, propiamente a la estación de trabajo.

⁶ Programa estándar para la emulación de terminales virtuales remotas de un sistema de computo.

Para lograr esta tarea se puede pensar en varias alternativas: obtener respaldos en cinta de la información depurada, etc. Sin embargo debemos recordar que los sistemas operativos de las máquinas son totalmente incompatibles, sus dispositivos de almacenamiento secundario muy diferentes, etc. A pesar de todas estas desventajas tenemos un factor en común: ***ambas máquinas están conectadas en RedUNAM***. Por lo que debe haber una forma en que todas las máquinas que se encuentran conectadas a RedUNAM, y en general a redes mucho más grandes como puede ser la ***Internet***, puedan compartir la información que se almacena en ellas.

Este es un problema típico de las redes de computadoras, ya que se debe de recordar que el fin último de toda red de computadoras es la de ***compartir la información***, por lo que se establecieron una serie de estándares en lo que respecta a las comunicaciones entre máquinas conectadas a una red.

Distintos sistemas operativos tienen diferentes convenciones para denominar a un archivo, así como diferentes formas de presentación, etc. La transferencias de archivos entre dos sistemas diferentes requieren la solución de estas incompatibilidades.

La base de este estándar data de los finales de los 60's y principios de los 70's en los Estados Unidos, y que surge como un proyecto de la Agencia de Defensa llamada ***ARPANET (Red de la Agencia de Proyectos de Investigación Avanzada)*** y su objetivo era el siguiente: mantener la comunicación entre dos puntos o ***hosts*** aún y cuando se tuviera un ataque enemigo (actualmente las fallas más comunes son por el clima, manchas solares, malas conexiones; que por bombas atómicas). El trabajo de este proyecto de investigación fue el inicio de

las redes de computadoras a nivel mundial que actualmente conocemos como **Internet**, la cual no es más que **una red de redes**.

En el modelo de la ARPAnet, las comunicaciones siempre ocurren entre una fuente y un destino el cual puede desaparecer en cualquier momento (cualquier falla común) y fue diseñada para requerir un mínimo de información de las computadoras que son clientes.

Para enviar un mensaje a través de la red, lo único que tiene que hacer la computadora es poner la información a transmitir en un **sobre**, llamado paquete **IP (Internet Protocol)** y "direccionar" correctamente el sobre con la información. La comunicación entre computadoras, y no la red de hecho, son responsables de que la información llegue correctamente. La filosofía de esta forma de trabajo es que cada computadora pueda "hablar" con cualquier otra como un "par".

Debido al gran éxito que se tuvo con esta filosofía de trabajo la **ISO (Organización Internacional de Normalización)** trató de normalizarla creando el esquema o modelo **OSI (Interconexión de Sistemas Abiertos)**. Este modelo se basa en el uso de "capas" para definir una arquitectura de red, las cuales realizan una función muy específica y en las que se definen protocolos normalizados internacionalmente.

Sin embargo la gente involucrada con esto no podía esperar años para la normalización de esto sin poder aplicarlo. Así se dieron los primeros **softwares** que cumplían con este protocolo, fue puesto en marcha en una gran diversidad de máquinas y fue la única manera de comunicación entre ellas. Esto fue muy atractivo para las universidades e incluso para el gobierno.

Por aquella época se estaba creando el concepto de **LAN's (redes de área local)**, esta tecnología fue madurando hasta que en 1983 fue común para computadoras de escritorio. Antes de esto se fue desarrollando el sistema operativo **UNIX** en la Universidad de Berkeley, el cual contemplaba la comunicación entre computadoras incluyendo una serie de programas para ello. Esto creó una nueva necesidad: en vez de conectarse a un servidor local por lugar de trabajo, se pensaba en la interconexión a través de la red ARPANet que más tarde se convertiría en la Internet.

Así, podemos decir que ARPANet tiene protocolos que, a grandes rasgos, cubre el mismo campo que el cubierto por los protocolos definidos por ISO-OSI. El protocolo denominado **IP** es un protocolo sin conexión, es decir, se basa en el sistema postal: cada mensaje (carta) lleva consigo la dirección del destinatario y cada una de ellas se encamina independientemente por el sistema; y se diseñó para manejar la interconexión de un número grande de redes **LAN y WAN**. El modelo OSI no trató el aspecto de interconexión de redes en forma natural, sino como una idea posterior.

El protocolo de transporte de datos de la ARPA, es un protocolo orientado a la conexión denominado **TCP (Protocolo de Control de la Transmisión)**, es decir, que el usuario del servicio establece primero la conexión, la utiliza y después la termina; que en general se parece al protocolo de transporte del modelo OSI, pero difiere en cuanto a formato y otros detalles. **Este protocolo es el que usa el sistema operativo UNIX de Berkeley**, lo que hace que se haya difundido ampliamente, aún y cuando no sea parte del modelo OSI.

La transferencia de archivos es una situación típica en la que es deseable y apropiado tener un servicio orientado a conexión fiable. El propietario de un archivo quiere tener toda la seguridad de que todos los bits lleguen correctamente y en el mismo orden en que se enviaron.

Teniendo como base estos dos protocolos, las redes ARPA ofrecen un servicio: el de transferencia de archivos vía red y que están soportados por diversos protocolos, en este caso el de transferencia de archivos se denomina **FTP (File Transfer Protocol - Protocolo de Transferencia de Archivos)**.

Los objetivos del **FTP** son:

- Promover el intercambio de archivos (programas y/o datos).
- Reforzar de forma indirecta o implícita (vía programas) el uso de computadoras remotas conectadas a la red.
- Para proteger al usuario de las posibles variaciones de los sistemas de archivos debido a los diferentes sistemas operativos.
- Transferencia de la información de una manera eficiente y con mínimo de errores.

Entonces utilizando el programa **FTP** que el sistema operativo UNIX tiene, podemos hacer la transferencia de archivos entre los dos equipos aún y cuando estos sean muy diferentes.

Desde el ambiente UNIX, una transferencia de archivos se ve de la siguiente forma:

```
tzetzal 24:> ftp unamiii
```

En esta parte establecemos comunicación con el equipo A-12 UNISYS, en este caso el nombre del equipo es UnamIII y tzetzal es el nombre del equipo SUN SparcStation 10:

```
Connected to unamiii.  
220-Unisys A-Series FTP Server at UNAMIII; Version 40.118.101  
220 Session 22 on Thursday June 9,1994 16:56:44
```

Hasta ahora solo hemos establecido la conexión, pero falta tener acceso al sistema A-12, para ello necesitamos una clave (Name) que en este caso es dasu, y un **password** o contraseña para poder accederlo (algunos sistemas permiten el acceso de usuarios **anónimos** es decir usuarios que no tienen una clave en el sistema, pero tienen acceso restringido a la información):

```
Name (unamiii:root): dasu  
331 Password required for login  
Password:  
230-FTP user 'DASU' logged in at UNAMIII  
230 on Thursday June 9,1994 16:56:51  
ftp>
```

Pedimos entonces al programa que nos muestre los archivos que se encuentran almacenados en la A-12:

```
ftp> dir  
200 Port request accepted  
150 Initiating data transfer  
CANDE/STARTUP  
OBJECT/PBUENO  
OBJECT/EL_DEPURADOR  
OBJECT/EL_DEPURADOR/ANTIGUO  
ARCHIVO  
SALIDA449  
PLANTEL208
```

Indicamos al programa que las transferencias van a hacerse en forma de caracteres ascii y con el comando `get ARCHIVO archivo.a12` le indicamos que nos interesa traer el archivo denominado ARCHIVO y que lo deje en el sistema SUN con el nombre de archivo.a12, después de eso cerramos la conexión con el comando `quit` y estamos en el **prompt** del sistema SUN.

```
ftp>
ftp> ascii
200 Representation type is ASCII
ftp> get ARCHIVO archivo.a12

ftp> quit
221-FTP user 'DASU' logged out at UNAMIII
221 Session 22 ended on Thursday June 9,1994 16:59:3 -Goodbye
tzetzal 25:>
```

Como se puede observar la transferencia de archivos vía el programa **ftp** de UNIX, que es una implementación del protocolo FTP del modelo de red ARPA, es muy simple, tan sólo con algunos comandos tenemos la posibilidad de tener acceso a un sinnúmero de documentos e información que se encuentran almacenados en diversas partes en computadoras conectadas a red.

La descripción del protocolo FTP se puede obtener vía red como un **RFC (Request for Comments)** es decir un documento que guarda información relativa a algún tema en específico, por ejemplo la información aquí descrita fue tomada del RCF959 el cual es una descripción del FTP.

1.1.4 Especificación del formato de la información para su consulta

La información con la que se va a trabajar es la de historias académicas. Este es un documento en el que se registra el avance académico del alumno en todos los planteles de la Universidad. Cada uno de los planteles entrega a DGAE la relación de todos sus alumnos y es esta dirección la encargada de realizar la verificación de la información y de darle formato adecuado para la impresión y almacenamiento en microfichas.

Capítulo III Diseño de la propuesta.

Una vez depurado de los caracteres de impresión y de control a través de un programa en ALGOL, estos archivos se transfieren al equipo SUN para su formateo final. Estos archivos contienen las historias académicas de todos los alumnos del plantel y el formato del archivo es como el que se muestra a continuación.

				08/ABR/94			
				ABUNDIZ VELAZQUEZ ISRAEL		9362639-4	
				E.N.E.P. ACATLAN ECONOMIA		208 93	
				LIC EN ECONOMIA		21 94-1	
72	300	24.00	13				
			3				
0	0	0.00	16				
72	300	24.00	7.53				
PRIMER SEMESTRE							
208 0241 06				GEOGRAFIA ECONOMICA	B OR 93-1	1650718	1151 1
208 0291 06				HISTORIA ECONOMICA GENERAL I	S OR 93-1	1650726	1151 1
208 0322 06				INTRODUCCION A LA ECONOMIA	B OR 93-1	1650745	1151 1
208 0361 06				ECONOMIA POLITICA I	S OR 93-1	1650753	1151 1
208 0368 04				CIES I	B OR 93-1	1650775	1151 1
208 0497 08				MATEMATICAS I	NA EX 94-1	1775914	EA51 2 2
SEGUNDO SEMESTRE							
208 0292 06				HIST ECONOMICA GENERAL II	NA EX 93-2	1718183	EA51 1 1
208 0362 06				ECONOMIA POLITICA II	S OR 93-2	1710573	2251 1
208 0369 04				CIES II	B OR 93-2	1710591	2251 1
208 0500 06				METODOLOGIA DE LAS C SOCIAL	B OR 93-2	1710615	2251 1
208 0755 06				SOCIOLOGIA Y POLITICA I	B OR 93-2	1710621	2251 1
TERCER SEMESTRE							
208 0293 06				HIST ECON DE AMERICA LATINA	B OR 94-1	1765636	1352 1
208 0363 06				ECONOMIA POLITICA III	S OR 94-1	1765664	1351 1
208 0370 04				CIES III	B OR 94-1	1765691	1352 1
208 0756 06				SOCIOLOGIA Y POLITICA II	MB OR 94-1	1765717	1351 1
208 0814 06				TEORIA ECONOMICA I	NP EX 94-1	1775929	EA51 1 1
08/ABR/94							
				ACEVEDO PEA OFELIA		9058174-8	
				E.N.E.P. ACATLAN ECONOMIA		208 90	
				LIC EN ECONOMIA		21 94-1	
262	300	87.33	54				
			0				
14	0	0.00	54				
276	300	92.00	9.18				
PRIMER SEMESTRE							
208 0241 06				GEOGRAFIA ECONOMICA	MB OR 90-1	1353260	1102 1
208 0291 06				HISTORIA ECONOMICA GENERAL I	MB OR 90-1	1353271	1102 1
208 0322 06				INTRODUCCION A LA ECONOMIA	MB OR 90-1	1353299	1102 1
208 0361 06				ECONOMIA POLITICA I	MB OR 90-1	1353310	1102 1
208 0368 04				CIES I	MB OR 90-1	1353342	1102 1
208 0497 08				MATEMATICAS I	B OR 90-1	1353369	1102 1
SEGUNDO SEMESTRE							
208 0292 06				HIST ECONOMICA GENERAL II	MB OR 90-2	1401357	2202 1
208 0362 06				ECONOMIA POLITICA II	MB OR 90-2	1401372	2201 1
208 0369 04				CIES II	B OR 90-2	1401398	2202 1
208 0498 08				MATEMATICAS II	B EX 90-2	1414821	EA01 1 1
208 0500 06				METODOLOGIA DE LAS C SOCIAL	MB OR 90-2	1401428	2201 1
208 0755 06				SOCIOLOGIA Y POLITICA I	MB OR 90-2	1401437	2202 1

Como se observa en el listado el tipo y orden de la información es el que aparece en las historias impresas. Los datos que nos proporciona el archivo son los siguientes:

PRIMERA LÍNEA:

- Fecha de impresión.

SEGUNDA LÍNEA:

- Nombre del alumno.
- Número de cuenta.
- Número de páginas de las que consta la historia.

TERCERA LÍNEA:

- Nombre del plantel.
- Clave del plantel.
- Año de ingreso.

CUARTA LINEA:

- Carrera.
- Clave carrera.
- Semestre actual.

QUINTA LINEA:

- Avance de créditos obligatorios.
- Total créditos obligatorios.
- Porcentaje de avance.
- Número de materias aprobadas.

SEXTA LINEA:

- Número de asignaturas no aprobadas.

SEPTIMA LINEA:

- Avance de créditos optativos.
- Total créditos optativos.

- Porcentaje de avance de optativos.
- Número de asignaturas cursadas en total.

OCTAVA LINEA:

- Avance de créditos totales.
- Total créditos.
- Porcentaje de avance total.
- Promedio.

A continuación aparece la información sobre las materias cursadas

- Clave planteel donde se cursó la materia.
- Clave asignatura.
- Numero de créditos.
- Nombre de la asignatura.
- Calificación.
- Tipo de Examen.
- Periodo.
- Número de folio del acta.
- Grupo.
- Número de exámenes con los que se aprobó la materia.

Como se puede observar en algunos casos, como en el avance de crédito, la información no resulta clara para el usuario. Es por esto que se le debe dar un formato adecuado de salida, así como hacer una serie de modificaciones para facilitar el indexamiento e identificación de las diferentes historias académicas.

Para realizar todas estas acciones se eligió una de las herramientas más versátiles del sistema operativo UNIX: el ***lenguaje de programación AWK***. A continuación se enumeran los diferentes pasos que se siguieron para la depuración del programa.

1. Separación de historias académicas.
2. Unión de las historias académicas que corresponden a un mismo alumno.
3. Depuración de la información para eliminar líneas en blanco dentro de la historia académica.
4. Formato del encabezado y corrimiento de la información.

1. Separación de historias académicas.

El archivo proveniente de la A12 contiene en forma continua las historias de todos los alumnos del plantel correspondiente. Como se explicará más adelante la herramienta de indexamiento utilizada requiere identificar cada una de las historias de los alumnos como si fuera un documento aparte. Si se le da como archivo de entrada este archivo se necesita separar cada historia por un retorno de carro, para que de manera lógica identifique a cada historia como un documento. Así que el primer paso fue determinar la manera de separar cada historia de alumno.

La información se presenta con la misma estructura con la que aparece en la historia académica a excepción de que no aparece la forma que indica de que datos se trata, como avance de créditos, promedio, etc. Cuando nos es entregada la historia académica aparece en la parte superior la fecha en la cual fue impresa y esta siempre aparece en todas las hojas correspondientes a una sola historia. Por lo tanto, la fecha fue el patrón para identificar el principio de cada una de las historias.

2. Unión de las historias académicas que corresponden a un mismo alumno.

Si la historia académica de un alumno abarca más de una hoja se genera nuevamente el encabezado con toda la información del alumno y se imprime una segunda hoja de la historia con el resto de la información. En el archivo esto aparece como otra historia, lo cual no deseamos, pues queremos que al realizar búsquedas por el nombre de un alumno aparezca toda la información de una sola vez. Es por esto que el siguiente paso fue identificar cuando la historia de un alumno continúa en otra página, eliminar el encabezado de esta segunda página y pegar la información para que quede como una sola historia.

3. Depuración de la información para eliminar líneas en blanco dentro de la historia académica.

Para realizar el indexamiento es necesario separar cada historia por un retorno de carro. El funcionamiento del indexador utilizado se explicará con mayor detalle en otro capítulo, aquí sólo se explica la razón por la que se debió cuidar el detalle de líneas en blanco entre historias. El indexador realiza las búsquedas sobre un sólo archivo, el archivo de historias de todo el plantel, sin embargo al realizar una búsqueda de un alumno en particular sólo debe regresar la historia correspondiente a dicho alumno, es decir un sólo documento. Para poder dividir el archivo en documentos es necesario separar cada historia por retornos de carro, convirtiéndose así en el documento de un alumno cualquiera. Al realizar la búsqueda por el nombre solo regresa el documento o párrafo donde aparece el nombre de dicho alumno. De aquí la importancia que esta separación no exista en otro lado que no sea entre historias.

4. Formato del encabezado y corrimiento de la información.

Como se observa es necesario dar un formato a la información para que el usuario pueda identificar de manera clara toda la información que se presenta. Este formato incluye el poner un encabezado o títulos para saber de que información se trata. En cuanto al corrimiento de la información, en el archivo original los datos aparecen al principio con 12 caracteres en blanco lo que no permite ver la información de manera adecuada en una terminal con 80 columnas por pantalla.

Todo esto quedo implementado en un programa en awk que conforme llegan los archivos de la A12 se les depura para dejarlos listos para su indexamiento.

1.1.5 La herramienta AWK en el ambiente Unix

AWK es el comando más complejo de UNIX ya que es un lenguaje de programación hecho y derecho con una estructura muy similar a C y que ofrece una característica muy peculiar: la descomposición del archivo de entrada en registros y campos y su selección a través del reconocimiento de patrones.

Esta herramienta fue diseñada e implementada originalmente por Alfred Aho, Peter Weinberger y Brian Kerhghan en 1977. Su nombre está formado precisamente por las iniciales de los tres apellidos de sus autores. A pesar de que fue ideado para escribir programas muy cortos, sus facilidades y características pronto atrajeron a usuarios que comenzaron a escribir programas considerablemente más largos. Esto ocasionó la necesidad de nuevas cosas que

no se encontraban en el lenguaje original, de manera que se publicó una nueva versión mejorada en 1985.

Una de las principales mejoras en esta nueva versión era la posibilidad para los usuarios de definir sus propias funciones, además de incluir muchas funciones intrínsecas y otras características adicionales para escribir programas complejos.

ESTRUCTURA DEL LENGUAJE

Este tema no pretende ser una explicación completa de AWK, tan sólo se mencionarán algunas de las estructuras más importantes y de las que se emplearon en el programa para facilitar la comprensión del mismo.

Un programa en awk es una secuencia de uno o más enunciados patrón-acción de la siguiente manera:

```
patrón { acción }
```

```
patrón { acción }
```

```
•
```

```
•
```

```
•
```

La operación básica consiste en **leer una secuencia de líneas de entrada** una tras otra buscando líneas que coincidan con alguno de los patrones del programa. Cada línea de entrada es comparada contra el patrón en turno, si coincide la acción correspondiente, que puede incluir múltiples instrucciones, ésta es ejecutada. Entonces la siguiente línea es leída y el proceso vuelve a comenzar.

Ya sea el patrón o la acción pueden ser omitidos, si un patrón no tiene acción, la acción que realiza por default es la impresión de la línea completa que cumple con él.

AWK al leer su línea de entrada la divide en campos, el separador que toma por default es el espacio en blanco, sin embargo el usuario puede indicarle que tome otro cualquiera. Para hacer referencia a cada uno de los campos se utiliza \$1 para el primero, \$2 para el segundo y así sucesivamente. La línea entera o registro es llamada \$0.

A continuación se muestran una serie de tablas de referencia de patrones, acciones operadores más comunes que ayudaran a la comprensión del programa para realizar la conversión.

RESUMEN DE ACCIONES

Expresiones con constantes, asignaciones, llamadas a función

print lista de expresiones

printf (formato, lista de expresiones)

if (expresión) enunciado

if (expresión) enunciado **else** enunciado

for (expresión inicial; expresión de terminación; expresión de cambio)

enunciado

while (expresión) enunciado

{ enunciados }

ALGUNAS VARIABLES AUTOMÁTICAS

FS controla el separador de campos de entrada

NF número de campos en el registro actual

NR número de registros leídos en el momento

ALGUNAS FUNCIONES DE CADENA INCLUIDAS

match(s,r)

verifica si la cadena **s** contiene alguna subcadena que coincida con la expresión regular **r**, regresa la posición o 0 si no fue encontrada.

regresa una subcadena de **s** de **n** carácter comenzando en la posición **p**.

substr(s,p,n)

EJEMPLOS DE ESPECIFICACIONES DE printf

%s	Febrero	Febrero
%10s	Febrero	Febrero
%-10s	Febrero	Febrero
%.3s	Febrero	Feb
%10.3s	Febrero	Feb
%-10.3	Febrero	Feb

OPERADORES MAS UTILIZADOS	
asignación	x=2
condicional	x?y:z si x regresa y si no z
OR lógico	
AND lógico	&&
relacionales	== <= >= !=
No lógico	!
incremento	++
decremento	--
campo	\$

EJEMPLOS DE ESPECIFICACIONES DE printf		
%s	Febrero	Febrero
%10s	Febrero	Febrero
%-10s	Febrero	Febrero
%.3s	Febrero	Feb
%10.3s	Febrero	Feb
%-10.3	Febrero	Feb

1.1.6 El programa en AWK

```

#####
# REALIZADO POR:          Montes de Oca Mendoza Gabriel Ivan
#                       Padilla Chávez Ana Cecilia
#                       Valencia Figueroa Edgar
#
# FECHA DE ACTUALIZACION: 26 de Junio de 1994.
#
# LENGUAJE DE PROGRAMACION: NAWK
#
#####
# DESCRIPCION : Este programa presenta en un formato adecuado los datos
#               del encabezado de la historia academica. El formato que se
#               presenta al usuario es el siguiente:
#
#
# NOMBRE: ARCE LARA HUGO ALEJANDRO          No.CUENTA:9157912-2
# PLANTEL: E.N.E.P. ACATLAN ECONOMIA        CLAVE: 208
# CARRERA: LIC EN ECONOMIA                  CLAVE: 21
# GENERACION: 91
# SEMESTRE: 94-1
#
-----
#               REPORTE DE AVANCE EN CREDITOS      |      REPORTE DE ASIGNATURAS
#
# Obligatorios  Cubiertos  Totales  %Avance  |      APROBADAS : 28
#               162        300      54.00%  |      NO APROBADAS: 2
# Optativos     0          0        0.00%  |      TOTAL      : 30
# Totales       162        300      54.00%  |
#
#               |      PROMEDIO      : 8.71
#
-----
#
# El programa lee cada línea del programa fuente e identifica el tipo
# de
# informacion que presenta la extrae y la almacena en la variable
# adecuada.
# Una vez que se ha extraído toda la informacion del encabezado la imprime
# en el formato arriba ilustrado.
# El resto de la informacion que contiene materias y calificaciones, se
# imprime tal cual, solo se le quitan los espacios en blanco de principio
# de cada línea para reducir el tamaño de la línea a 78 caracteres.
#
#
BEGIN {
    linea_fecha = 0;          # Bandera que se enciende al encontrar la
                              # línea con la fecha.
    num_hoja      = 1;        # Numero de hojas de que consta la historia
    Leo_plantel=0;
    Seg_Historia=0;
}

#
# La funcion formato toma las 7 lineas de encabezado y extrae la
# informacion y la pone en un formato especificado.

function formato() {

#
# No se imprime la fecha y en su lugar se imprime un retorno de carro que
# va
# a ser nuestro separador de historias.
#
    printf("\n");
#
# En la siguiente línea se encuentran nombre del alumno, no_cuenta, no. de
# página de la historia y páginas de la que consta. Estos dos últimos

```

```

# datos
# se eliminan del reporte
#
    for ( i = 1; i <= (NF - 3); i++)
        nombre = nombre " " $i;
        no_cuenta=$(NF - 2);

# En esta línea se encuentran: nombre del plantel, clave del plantel y año
# de ingreso.

    getline;
    for ( i=1; i <= (NF - 2); i++)
        plantel=plantel " " $i;
        clave_p=$(NF - 1);
        ingreso=$(NF);

# En esta línea se encuentran: nombre carrera, clave y semestre. Si se
# trata de un alumno con cambio de carrera o plantel, la historia
# anterior no incluye nombre de carrera ni clave.

    getline;
    if (NF == 1)
    {
        carrera="";
        clave_c="";
    }
    else
    {
        for ( i=1; i <= (NF - 2); i++)
            carrera=carrera " " $i;
        clave_c=$(NF - 1);
    }
    semestre=$NF;

    # Verifica si el alumno llevo alguna otra carrera.

    getline;
    if ( match($0,/ALUMNO PRESENTA REGISTRO/) )
    {
        Lee_Plantel=1; # Enciende la bandera para leer el plantel
                      # donde presenta otro registro.
        Num_Historia=1;
    }
    else {
        Seg_Historia=0; # Apaga banderas
        Num_Historia=0;
    }
    Cred_Ob_c=$1;
    if ( Seg_Historia )
    {
        Mat_Aprob=$2;
        Cred_Ob="";
        Avance_Cred_Ob="";
    }
    else
    {
        Cred_Ob=$2;
        Avance_Cred_Ob=$3;
        Mat_Aprob=$4;
    }

    getline;
    Mat_Reprob = $1;

    getline;
    Cred_Op_c=$1;
    if ( Seg_Historia )
    {
        Cred_Op="";
        Avance_Cred_Op="";
        Total_Mat=$2;
    }
    else

```

```

{
Cred_Op=$2;
Avance_Cred_Op=$3;
Total_Mat=$4;
}
if (Lee_Plantel)
Reg_Plantel=$NF;

getline;
Cred_Tot_c=$1;
if (Seg_Historia)
{
Cred_Tot="";
Avance_Cred="";
Promedio=$2;
}
else
{
Cred_Tot=$2;
Avance_Cred=$3;
Promedio=$4;
}

printf(" NOMBRE: %-35sNo. CUENTA:%s\n",nombre,no_cuenta);
printf(" PLANTEL: %-35sCLAVE:%s\n",plantel,clave_p);
printf(" CARRERA: %-35sCLAVE:%s\n",carrera,clave_c);
printf(" GENERACION: %s\n",ingreso);
printf(" SEMESTRE: %s\n",semestre);
print " ";

if (Lee_Plantel)
{
print " ALUMNO PRESENTA REGISTRO ACADEMICO EN OTRO PLANTEL";
print " ALUMNO REGISTRADO EN PLANTEL " Reg_Plantel;
print " ";
Lee_Plantel=0;
}

}

print "-----";
ASIGNATURAS\n");
printf(" REPORTE DE AVANCE EN CREDITOS | REPORTE DE
);
printf("
Cubiertos Totales %%Avance | \n");
printf(" APROBADAS :
%2s\n",Mat_Aprob);
printf(" Obligatorios %3s %3s %6s%% | NO APROBADAS:
%2s\n",Cred_Ob_c, Cred_Ob, Avance_Cred_Ob, Mat_Reprob);
printf(" Optativos %3s %3s %6s%% | TOTAL :
%2s\n",Cred_Op_c, Cred_Op, Avance_Cred_Op, Total_Mat);
printf(" Totales %3s %3s %6s%%
|\n",Cred_Tot_c,Cred_Tot,Avance_Cred);
printf(" PROMEDIO :
%5s\n",Promedio);
printf("
\n");
print " ";
nombre="";
plantel="";
carrera="";

```


1.2 Pruebas finales.

La prueba de nuestro software es un elemento crítico para la garantía de calidad del mismo y representa un último repaso de las especificaciones, del diseño y de la codificación. Asimismo la **Ingeniería del software** nos dice que las pruebas que se hagan a un software (o a un sistema completo) deben de ser exhaustivas y nos da una serie de reglas que sirven como objetivos de la prueba:

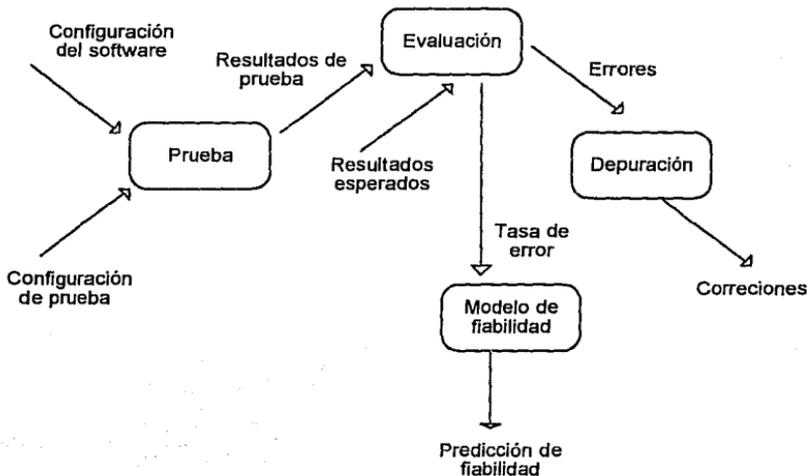
1. La prueba es un proceso de ejecución de un programa con la intención de descubrir algún error.
2. Un buen caso de prueba es aquel que tiene una alta probabilidad de mostrar un error no descubierto hasta entonces.
3. Una prueba tiene éxito si se descubre un error no detectado hasta entonces.

Nos podemos apoyar en estas tres reglas para hacer pruebas a nuestros programas sin embargo, cabe resaltar que los programas realizados no pretenden ser todo un caso de estudio de Ingeniería del software, sino que simplemente son una herramienta que nos permitió depurar la información de las historias académicas. Una frase bastante pesimista en lo que se refiere a las pruebas de software dice: ***"La prueba no puede asegurar la ausencia de defectos; sólo pueden demostrar que existen defectos en el software"***.

Las pruebas finales de los dos programas depuradores expuestos en capítulos anteriores se pueden dividir en dos: la prueba en el ambiente de programación del mainframe **A-12** a través de su lenguaje nativo, **ALGOL** y las pruebas en el

ambiente de la estación de trabajo **SparcStation 10** (corriendo la versión del **UNIX** de SUN la cual es **Solaris 2.2**) y más específicamente las pruebas de la herramienta **AWK** sobre los datos arrojados por la A-12.

A continuación podemos definir las pruebas a través del siguiente diagrama:



La configuración de nuestro software fue explicada en capítulos anteriores, en los cuales se definieron especificaciones y descripciones tanto del lenguaje de programación (ALGOL y la herramienta AWK) como del problema que deben solucionar: **depurar** lo más posible la información generada para la grabación de las historias en microfichas y **formatear** dicha información para su consulta una vez grabada en CD-ROM.

• Pruebas del programa en ALGOL.

La configuración de la prueba en este caso fue la información que se tiene en cinta magnética la cual es usada para el proceso de microfilmación. Esta información fue transferida de la cinta al mainframe **A-12**.

A continuación se presenta un fragmento de la información que se tiene en cintas:

```

.....
.....dr.....m.....XEROX HIST ACAD 1 TA...HAPROD.....
...U-H-A-M.....HA-941-208-AC-
941.....
.....
.....
.....
.....
XEROX HIST ACAD 1 TANTO.....STA17002 CANDE.1.....
08/ABR/94 ..... ABUNDIZ VELAZQUEZ ISRAEL
9362639-4 1 1 ..... E.N.E.P. ACATLAN (ECONOMIA)
208 93..... LIC EN ECONOMIA 21
94-1 ..... 72 300 24.00 13 .....
3 ..... 0 0 0.00 16 ..... 72 300
24.00 ..... 7.53 ..... PRIMER SEMESTRE .....
208 0241 06 GEOGRAFIA ECONOMICA B OR 93-1 1650718 1151 1 .....
208 0322 06 INTRODUCCION A LA ECONOMIA B OR 93-1 1650745 1151 1 .....
208 0361 06 ECONOMIA POLITICA I S OR 93-1 1650753 1151 1 .....
208 0368 04 CIES I B OR 93-1 1650775 1151 1 .....
208 0497 08 MATEMATICAS I NA EX 94-1 1775914 EA51 2 2
.....
..... 208 0291 06 HISTORIA ECONOMICA GENERAL I S OR 93-1 1650726 1151 1
..... SEGUNDO SEMESTRE ..... 208 0292 06 HIST ECONOMICA GENERAL
II NA EX 93-2 1718183 EA51 1 1 ..... 208 0362 06 ECONOMIA POLITICA II
S OR 93-2 1710573 2251 1 ..... 208 0369 04 CIES II B OR
OR 93-2 1710591 2251 1 ..... 208 0500 06 METODOLOGIA DE LAS C SOCIAL B OR
93-2 1710615 2251 1 ..... 208 0755 06 SOCIOLOGIA Y POLITICA I B OR
93-2 1710621 2251 1 ..... TERCER SEMESTRE ..... 208 0293
06 HIST ECON DE AMERICA LATINA B OR 94-1 1765636 1352 1 ..... 208 0363 06
ECONOMIA POLITICA III S OR 94-1 1765664 1351 1 ..... 208 0370 04
CIES III B OR 94-1 1765691 1352 1 ..... 208 0756 06
SOCIOLOGIA Y POLITICA II MB OR 94-1 1765717 1351 1 ..... 208 0814 06
TEORIA ECONOMICA I NP EX 94-1 1775929 EA51 1 1 .....

```

Como se puede observar la información de las historias académicas está totalmente mezclada con información y caracteres de control de impresión, los cuales son interpretados por la unidad de microfilmación para grabarlos en la microficha.

Una vez que pasan por el programa depurador se tiene la siguiente salida:

				08/ABR/94		
ABUNDIZ VELAZQUEZ ISRAEL				9362639-4	1	1
E.N.E.P. ACATLAN ECONOMIA				208	93	
LIC EN ECONOMIA				21	94-1	
72	300	24.00	13			
			3			
0	0	0.00	16			
72	300	24.00	7.53			
PRIMER SEMESTRE						
208 0241	06	GEOGRAFIA ECONOMICA		B OR 93-1	1650718	1151 1
208 0291	06	HISTORIA ECONOMICA GENERAL I		S OR 93-1	1650726	1151 1
208 0322	06	INTRODUCCION A LA ECONOMIA		B OR 93-1	1650745	1151 1
208 0361	06	ECONOMIA POLITICA I		S OR 93-1	1650753	1151 1
208 0368	04	CIES I		B OR 93-1	1650775	1151 1
208 0497	08	MATEMATICAS I		NA EX 94-1	1775914	EA51 2 2
SEGUNDO SEMESTRE						
208 0292	06	HIST ECONOMICA GENERAL II		NA EX 93-2	1718183	EA51 1 1
208 0362	06	ECONOMIA POLITICA II		S OR 93-2	1710573	2251 1
208 0369	04	CIES II		B OR 93-2	1710591	2251 1
208 0500	06	METODOLOGIA DE LAS C SOCIAL		B OR 93-2	1710615	2251 1
208 0755	06	SOCIOLOGIA Y POLITICA I		B OR 93-2	1710621	2251 1
TERCER SEMESTRE						
208 0293	06	HIST ECON DE AMERICA LATINA		B OR 94-1	1765636	1352 1
208 0363	06	ECONOMIA POLITICA III		S OR 94-1	1765664	1351 1
208 0370	04	CIES III		B OR 94-1	1765691	1352 1
208 0756	06	SOCIOLOGIA Y POLITICA II		MB OR 94-1	1765717	1351 1
208 0814	06	TEORIA ECONOMICA I		NP EX 94-1	1775929	EA51 1 1

				08/ABR/94		
ACEVEDO FERRER INGRID				8906263-4	1	1
E.N.E.P. ACATLAN ECONOMIA				208	92	
LIC EN ECONOMIA				21	94-1	
174	300	58.00	30			
			0			
0	0	0.00	30			
174	300	58.00	8.80			

PRIMER SEMESTRE						
208 0241	06	GEOGRAFIA ECONOMICA		MB OR 92-1	1552721	1102 1
208 0291	06	HISTORIA ECONOMICA GENERAL I		MB OR 92-1	1552729	1102 1
208 0322	06	INTRODUCCION A LA ECONOMIA		MB OR 92-1	1552749	1102 1
208 0361	06	ECONOMIA POLITICA I		B OR 92-1	1552757	1102 1
208 0368	04	CIES I		MB OR 92-1	1552782	1102 1
208 0497	08	MATEMATICAS I		S OR 92-1	1552803	1102 1
SEGUNDO SEMESTRE						
208 0292	06	HIST ECONOMICA GENERAL II		B OR 92-2	1606372	2202 1
208 0362	06	ECONOMIA POLITICA II		MB OR 92-2	1606384	2202 1
208 0369	04	CIES II		B OR 92-2	1606398	2201 1
208 0498	08	MATEMATICAS II		B OR 92-2	1606420	2202 1
208 0500	06	METODOLOGIA DE LAS C SOCIAL		MB OR 92-2	1606426	2202 1
208 0755	06	SOCIOLOGIA Y POLITICA I		MB OR 92-2	1606431	2201 1

Etcétera.

Para poder llegar a esta salida el programa tuvo que ser reestructurado varias veces, debido a que los caracteres de control indeseables del archivo en algunas ocasiones no fueron eliminados y debido a la naturaleza de la información estos

estaban en posiciones totalmente aleatorias, por lo que no se pudo establecer un patrón para su eliminación. Sin embargo se puede afirmar que el 95% de estos caracteres fueron eliminados.

Otro problema que se detectó en el programa fue la incapacidad de detectar cierta "basura" del archivo, es decir, dentro de éste se tenían ciertos campos los cuales eran información no útil para la historia o simplemente caracteres fuera de orden o de toda secuencia; por desgracia debido a que estos son totalmente aleatorios el programa depurador no puede saber si estos son información útil o si simplemente es basura. Tal es el caso de la siguiente historia:

		SEGUNDO SEMESTRE			
208	0292	06	HIST ECONOMICA GENERAL II		B OR 92-2 1606372 2202 1
208	0362	06	ECONOMIA POLITICA II	1-1-1	MB OR 92-2 1606384 2202 1
208	0369	04	CIES II		B OR 92-2 1606398 2201 1
208	0498	08	MATEMATICAS II	HISTORIAS	B OR 92-2 1606420 2202 1
208	0500	06	METODOLOGIA DE LAS C SOCIAL		MB OR 92-2 1606426 2202 1
208	0755	06	SOCIOLOGIA Y POLITICA I		MB OR 92-2 1606431 2201 1
			TERCER SEMESTRE	CUARTO SEMESTRE	
208	0293	06	HIST ECON DE AMERICA LATINA		B OR 93-1 1650731 1302 1
208	0363	06	ECONOMIA POLITICA III		B OR 93-1 1650758 1302 1
208	0370	04	CIES III		MB OR 93-1 1650783 1352 1
208	0499	08	MATEMATICAS III		MB OR 93-1 1650800 1301 1
208	0756	06	SOCIOLOGIA Y POLITICA II		MB OR 93-1
			DW MATEMATICAS 94-1		NP 3221
208	0814	06	TEORIA ECONOMICA I		MB OR 93-1 1650816 1352 1

Estos caracteres están en la historia por diversos motivos:

- Errores en la captura de datos.
- Modificaciones hechas al momento de procesar.
- Mal manejo de la información en cinta, etc.

Sin embargo para efectos prácticos del proyecto y debido a que estos errores son mínimos se permitió que existieran.

La Ingeniería del software nos dice además que se deben de hacer pruebas de **caja blanca y de caja negra**, dependiendo de la complejidad del mismo. Una

prueba de caja blanca se centra en la estructura de control que tiene el programa, además se debe verificar que todas las sentencias del programa al menos se ejecuten una vez y que se ejerciten todas las condiciones lógicas. Las pruebas de caja negra son diseñadas para validar los requerimientos funcionales sin fijarse en el funcionamiento interno del programa, para este caso se diseñan pruebas que cubran cierta área del programa y sea probado. El análisis de los datos de salida puede ser un parámetro a evaluar para ver si se encuentra en los límites aceptables.

Debido a la simplicidad del software desarrollado no fueron necesarias las pruebas de *caja blanca*, la única función del programa es la de leer caracter por caracter el archivo de datos; si este caracter se encuentra entre los de control o dentro de los no deseados, se desecha. El resultado final es almacenado en un archivo de salida.

Sin embargo las pruebas de *caja negra* fueron ampliamente usadas, debido a las características de la información se tuvieron que hacer modificaciones hasta que la información resultante estuviera en un límite aceptable, esto es: menos de 1% - 2% de información no útil o "*basura*".

Otro de los detalles que se pueden observar de las historias es que debido a que la información que presenta solo cabe en una hoja tamaño carta, existen varias historias que se repiten, es decir, se tiene más de una hoja por historias académicas totales (tal es el caso de los alumnos de semestres avanzados) tal es el caso de ACEVEDO PEA OFELIA que tiene dos hojas de historia académica. Se puede observar además que existe un número consecutivo al final de cada una de ellas, este caracter no pudo ser eliminado por el programa en ALGOL.

• Pruebas del programa en AWK.

Una vez que la información se tiene en un formato legible, el siguiente paso fue exportar a la estación de trabajo SparcStation 10 y realizar otro proceso de depuración. Este paso fue mucho más sencillo que en el ambiente A-12, debido a que UNIX nos provee de muchas herramientas para el manejo de textos y caracteres, tal es el caso de la herramienta **AWK**.

El programa se mostró en secciones anteriores y el objetivo de este en primer paso fue la eliminación de las historias repetidas para evitar redundancia en la información y además de cierta forma toda esta información conforma una sola historia. Como segundo paso fue necesario quitar la numeración de cada historia, es decir, el **número de hoja** de cada una de ellas.

El resultado del programa fue el siguiente:

NOMBRE: ABUNDIZ VELAZQUEZ ISRAEL	No. CUENTA: 9362639-4
PLANTEL: E.N.E.F. ACATLAN ECONOMIA	CLAVE: 208
CARRERA: LIC EN ECONOMIA	CLAVE: 21
GENERACION: 93	
SEMESTRE: 94-1	

REPORTE DE AVANCE EN CREDITOS				REPORTE DE ASIGNATURAS	
	Cubiertos	Totales	%Avance	APROBADAS	: 13
Obligatorios	72	300	24.00%	NO_APROBADAS:	3
Optativos	0	0	0.00%	TOTAL	: 16
Totales	72	300	24.00%	PROMEDIO	: 7.53

PRIMER SEMESTRE					
208 0241 06	GEOGRAFIA ECONOMICA	B OR 93-1	1650718	1151	1
208 0291 06	HISTORIA ECONOMICA GENERAL I	S OR 93-1	1650726	1151	1
208 0322 06	INTRODUCCION A LA ECONOMIA	B OR 93-1	1650745	1151	1
208 0361 06	ECONOMIA POLITICA I	S OR 93-1	1650753	1151	1
208 0368 04	CIES I	B OR 93-1	1650775	1151	1
208 0497 08	MATEMATICAS I	NA EX 94-1	1775914	EA51	2 2
SEGUNDO SEMESTRE					
208 0292 06	HIST ECONOMICA GENERAL II	NA EX 93-2	1718183	EA51	1 1
208 0362 06	ECONOMIA POLITICA II	S OR 93-2	1710573	2251	1
208 0369 04	CIES II	B OR 93-2	1710591	2251	1
208 0500 06	METODOLOGIA DE LAS C SOCIAL	B OR 93-2	1710615	2251	1
208 0755 06	SOCIOLOGIA Y POLITICA I	B OR 93-2	1710621	2251	1
TERCER SEMESTRE					
208 0293 06	HIST ECON DE AMERICA LATINA	B OR 94-1	1765636	1352	1
208 0363 06	ECONOMIA POLITICA III	S OR 94-1	1765664	1351	1
208 0370 04	CIES III	B OR 94-1	1765691	1352	1

Migración de la información al ambiente UNIX.

208 0756 06 SOCIOLOGIA Y POLITICA II MB OR 94-1 1765717 1351 1
 208 0814 06 TEORIA ECONOMICA I NP EX 94-1 1775929 EA51 1 1

NOMBRE: ACEVEDO FERRER INGRID
 PLANTEL: E.N.E.P. ACATLAN ECONOMIA
 CARRERA: LIC EN ECONOMIA
 GENERACION: 92
 SEMESTRE: 94-1

No. CUENTA: 8906263-4
 CLAVE: 208
 CLAVE: 21

REPORTE DE AVANCE EN CREDITOS				REPORTE DE ASIGNATURAS	
	Cubiertos	Totales	%Avance	APROBADAS	: 30
Obligatorios	174	300	58.00%	NO APROBADAS:	0
Optativos	0	0	0.00%	TOTAL	: 30
Totales	174	300	58.00%	PROMEDIO	: 8.80

PRIMER SEMESTRE

208 0241 06 GEOGRAFIA ECONOMICA MB OR 92-1 1552721 1102 1
 208 0291 06 HISTORIA ECONOMICA GENERAL I MB OR 92-1 1552729 1102 1
 208 0322 06 INTRODUCCION A LA ECONOMIA MB OR 92-1 1552749 1102 1
 208 0361 06 ECONOMIA POLITICA I B OR 92-1 1552757 1102 1
 208 0368 04 CIES I MB OR 92-1 1552782 1102 1
 208 0497 08 MATEMATICAS I S OR 92-1 1552803 1102 1

SEGUNDO SEMESTRE

208 0292 06 HIST ECONOMICA GENERAL II B OR 92-2 1606372 2202 1
 208 0362 06 ECONOMIA POLITICA II MB OR 92-2 1606384 2202 1
 208 0369 04 CIES II B OR 92-2 1606398 2201 1
 208 0498 08 MATEMATICAS II B OR 92-2 1606420 2202 1
 208 0500 06 METODOLOGIA DE LAS C SOCIAL MB OR 92-2 1606426 2202 1
 208 0755 06 SOCIOLOGIA Y POLITICA I MB OR 92-2 1606431 2201 1

TERCER SEMESTRE

208 0293 06 HIST ECON DE AMERICA LATINA B OR 93-1 1650731 1302 1
 208 0363 06 ECONOMIA POLITICA III B OR 93-1 1650758 1302 1
 208 0370 04 CIES III MB OR 93-1 1650783 1352 1
 208 0499 08 MATEMATICAS III MB OR 93-1 1650800 1301 1
 208 0756 06 SOCIOLOGIA Y POLITICA II MB OR 93-1 1650806 1301 1
 208 0814 06 TEORIA ECONOMICA I MB OR 93-1 1650816 1352 1

CUARTO SEMESTRE

208 0004 04 ANALISIS D EDOS FINANCIEROS B OR 93-2 1710531 2452 1
 208 0164 08 ESTADISTICA I B OR 93-2 1710548 2451 1
 208 0294 06 HIST ECONOMICA D MEXICO I S OR 93-2 1710566 2451 1
 208 0364 06 ECONOMIA POLITICA IV B OR 93-2 1710580 2452 1
 208 0371.04 CIES IV B OR 93-2 1710599 2453 1
 208 0815 06 TEORIA ECONOMICA II B OR 93-2 1710627 2451 1

QUINTO SEMESTRE

208 0099 04 CONTABILIDAD SOCIAL B OR 94-1 1765590 1551 1
 208 0165 08 ESTADISTICA II B OR 94-1 1765602 1552 1
 208 0295 04 HIST ECONOMICA D MEXICO II MB OR 94-1 1765642 1552 1
 208 0365 06 ECONOMIA POLITICA V MB OR 94-1 1765669 1551 1
 208 0372 04 CIES V MB OR 94-1 1765698 1553 1
 208 0816 06 TEORIA ECONOMICA III B OR 94-1 1765729 1551 1

NOMBRE: ACEVEDO PEA OFELIA
 PLANTEL: E.N.E.P. ACATLAN ECONOMIA
 CARRERA: LIC EN ECONOMIA
 GENERACION: 90
 SEMESTRE: 94-1

No. CUENTA: 9058174-8
 CLAVE: 208
 CLAVE: 21

REPORTE DE AVANCE EN CREDITOS				REPORTE DE ASIGNATURAS	
	Cubiertos	Totales	%Avance	APROBADAS	: 54
Obligatorios	262	300	87.33%	NO APROBADAS:	0
Optativos	14	0	0.00%	TOTAL	: 54
Totales	276	300	92.00%	PROMEDIO	: 9.18

Capitulo III Diseño de la propuesta.

PRIMER SEMESTRE						
208	0241	06	GEOGRAFIA ECONOMICA	MB OR 90-1	1353260	1102 1
208	0291	06	HISTORIA ECONOMICA GENERAL I	MB OR 90-1	1353271	1102 1
208	0322	06	INTRODUCCION A LA ECONOMIA	MB OR 90-1	1353299	1102 1
208	0361	06	ECONOMIA POLITICA I	MB OR 90-1	1353310	1102 1
208	0368	04	CIES I	MB OR 90-1	1353342	1102 1
208	0497	08	MATEMATICAS I	B OR 90-1	1353369	1102 1
SEGUNDO SEMESTRE						
208	0292	06	HIST ECONOMICA GENERAL II	MB OR 90-2	1401357	2202 1
208	0362	06	ECONOMIA POLITICA II	MB OR 90-2	1401372	2201 1
208	0369	04	CIES II	B OR 90-2	1401398	2202 1
208	0498	08	MATEMATICAS II	B EX 90-2	1414821	EA01 1
208	0500	06	METODOLOGIA DE LAS C SOCIAL	MB OR 90-2	1401428	2201 1
208	0755	06	SOCIOLOGIA Y POLITICA I	MB OR 90-2	1401437	2202 1
TERCER SEMESTRE						
208	0293	06	HIST ECON DE AMERICA LATINA	MB OR 91-1	1457315	1301 1
208	0363	06	ECONOMIA POLITICA III	MB OR 91-1	1457352	1301 1
208	0370	04	CIES III	MB OR 91-1	1457385	1301 1
208	0499	08	MATEMATICAS III	S OR 91-1	1457414	1302 1
208	0756	06	SOCIOLOGIA Y POLITICA II	B OR 91-1	1457421	1302 1
208	0814	06	TEORIA ECONOMICA I	MB OR 91-1	1457427	1301 1
CUARTO SEMESTRE						
208	0004	04	ANALISIS D EDOS FINANCIEROS	S OR 91-2	1500787	2401 1
208	0164	08	ESTADISTICA I	MB OR 91-2	1500813	2401 1
208	0294	06	HIST ECONOMICA D MEXICO I	MB OR 91-2	1500829	2401 1
208	0364	06	ECONOMIA POLITICA IV	MB OR 91-2	1500840	2401 1
208	0371	04	CIES IV	MB OR 91-2	1500863	2404 1
208	0815	06	TEORIA ECONOMICA II	S OR 91-2	1500897	2401 1
QUINTO SEMESTRE						
208	0099	04	CONTABILIDAD SOCIAL	B OR 92-1	1552689	1501 1
208	0165	08	ESTADISTICA II	MB OR 92-1	1552702	1501 1
208	0295	04	HIST ECONOMICA D MEXICO II	MB OR 92-1	1552742	1502 1
208	0365	06	ECONOMIA POLITICA V	MB OR 92-1	1552768	1501 1
208	0372	04	CIES V	MB OR 92-1	1552796	1503 1
208	0816	06	TEORIA ECONOMICA III	MB OR 92-1	1552828	1502 1
SEXTO SEMESTRE						
208	0005	04	ANAL DEL CAMBIO SOC EN MEXCO	B OR 92-2	1606348	2601 1
208	0131	04	DESARROLLO Y SUBDESLELO I	MB OR 92-2	1606354	2601 1

Como se puede observar también se eliminó la fecha de la elaboración de la historia ya que para efectos prácticos es un dato poco usado dentro de la misma.

Sin embargo dentro de las pruebas no fue posible percatarse de un problema que no se podía ver en la estación de trabajo: la consulta de la información, en la mayoría de los casos, será hecha vía PC en red; algunas veces será consultada por otras estaciones de trabajo. Como se sabe la capacidad de desplegar texto en un monitor de PC es de 80 columnas⁷ (los renglones no importan ya que la aplicación permite el uso de las flechas de navegación del cursor). Si se observa con atención las historias estas rebasan dicho límite por lo que las historias se verán así:

⁷ Cabe señalar que los minitores VGA pueden desplegar más de 80 columnas en modo texto, sin embargo no todas las PC's conectadas a red tienen este tipo de monitor, por lo que es bueno estandarizarlo a 80 columnas.

Migración de la información al ambiente UNIX.

CUARTO SEMESTRE

2452	208 0004 04	ANALISIS D EDOS FINANCIEROS	B OR 93-2	1710531
	1			
2451	208 0164 08	ESTADISTICA I	B OR 93-2	1710548
	1			
2451	208 0294 06	HIST ECONOMICA D MEXICO I	S OR 93-2	1710566
	1			
2452	208 0364 06	ECONOMIA POLITICA IV	B OR 93-2	1710580
	1			
2453	208 0371 04	CIES IV	B OR 93-2	1710599
	1			
2451	208 0815 06	TEORIA ECONOMICA II	B OR 93-2	1710627
	1			

Por lo que el programa fue depurado para eliminar caracteres en blanco al principio y al fin de cada renglón de la historia, a continuación se muestra la salida de éste:

NOMBRE: ABUNDIZ VELAZQUEZ ISRAEL
 PLANTEL: E.N.E.P. ACATLAN ECONOMIA
 CARRERA: LIC EN ECONOMIA
 GENERACION: 93
 SEMESTRE: 94-1

No. CUENTA:9362639-4
 CLAVE:208
 CLAVE:21

REPORTE DE AVANCE EN CREDITOS				REPORTE DE ASIGNATURAS	
Obligatorios	Cubiertos	Totales	%Avance	APROBADAS	: 13
Optativos	72	300	24.00%	NO APROBADAS:	3
Totales	0	0	0.00%	TOTAL	: 16
	72	300	24.00%	PROMEDIO	: 7.53

PRIMER SEMESTRE

208 0241 06	GEOGRAFIA ECONOMICA	B OR 93-1	1650718	1151	1
208 0291 06	HISTORIA ECONOMICA GENERAL I	S OR 93-1	1650726	1151	1
208 0322 06	INTRODUCCION A LA ECONOMIA	B OR 93-1	1650745	1151	1
208 0361 06	ECONOMIA POLITICA I	S OR 93-1	1650753	1151	1
208 0368 04	CIES I	B OR 93-1	1650775	1151	1
208 0497 08	MATEMATICAS I	NA EX 94-1	1775914	EA51	2 2

SEGUNDO SEMESTRE

208 0292 06	HIST ECONOMICA GENERAL II	NA EX 93-2	1718183	EA51	1 1
208 0362 06	ECONOMIA POLITICA II	S OR 93-2	1710573	2251	1
208 0369 04	CIES II	B OR 93-2	1710591	2251	1
208 0500 06	METODOLOGIA DE LAS C SOCIAL	B OR 93-2	1710615	2251	1
208 0755 06	SOCIOLOGIA Y POLITICA I	B OR 93-2	1710621	2251	1

TERCER SEMESTRE

208 0293 06	HIST ECON DE AMERICA LATINA	B OR 94-1	1765636	1352	1
208 0363 06	ECONOMIA POLITICA III	S OR 94-1	1765664	1351	1
208 0370 04	CIES III	B OR 94-1	1765691	1352	1
208 0756 06	SOCIOLOGIA Y POLITICA II	MB OR 94-1	1765717	1351	1
208 0814 06	TEORIA ECONOMICA I	NP EX 94-1	1775929	EA51	1 1

NOMBRE: ACEVEDO FERRER INGRID
 PLANTEL: E.N.E.P. ACATLAN ECONOMIA
 CARRERA: LIC EN ECONOMIA
 GENERACION: 92
 SEMESTRE: 94-1

No. CUENTA:8906263-4
 CLAVE:208
 CLAVE:21

Capítulo III Diseño de la propuesta.

REPORTE DE AVANCE EN CREDITOS				REPORTE DE ASIGNATURAS	
	Cubiertos	Totales	%Avance	APROBADAS	: 30
Obligatorios	174	300	58.00%	NO APROBADAS:	0
Optativos	0	0	0.00%	TOTAL	: 30
Totales	174	300	58.00%	PROMEDIO	: 8.80

PRIMER SEMESTRE					
208 0241 06	GEOGRAFIA ECONOMICA	MB OR 92-1	1552721	1102	1
208 0291 06	HISTORIA ECONOMICA GENERAL I	MB OR 92-1	1552729	1102	1
208 0322 06	INTRODUCCION A LA ECONOMIA	MB OR 92-1	1552749	1102	1
208 0361 06	ECONOMIA POLITICA I	B OR 92-1	1552757	1102	1
208 0368 04	CIES I	MB OR 92-1	1552782	1102	1
208 0497 08	MATEMATICAS I	S OR 92-1	1552803	1102	1
SEGUNDO SEMESTRE					
208 0292 06	HIST ECONOMICA GENERAL II	B OR 92-2	1606372	2202	1
208 0362 06	ECONOMIA POLITICA II	MB OR 92-2	1606384	2202	1
208 0369 04	CIES II	B OR 92-2	1606398	2201	1
208 0498 08	MATEMATICAS II	B OR 92-2	1606420	2202	1
208 0500 06	METODOLOGIA DE LAS C SOCIAL	MB OR 92-2	1606426	2202	1
208 0755 06	SOCIOLOGIA Y POLITICA I	MB OR 92-2	1606431	2201	1
TERCER SEMESTRE					
208 0293 06	HIST ECON DE AMERICA LATINA	B OR 93-1	1650731	1302	1
208 0363 06	ECONOMIA POLITICA III	B OR 93-1	1650758	1302	1
208 0370 04	CIES III	MB OR 93-1	1650783	1352	1
208 0499 08	MATEMATICAS III	MB OR 93-1	1650800	1301	1
208 0756 06	SOCIOLOGIA Y POLITICA II	MB OR 93-1	1650806	1301	1
208 0814 06	TEORIA ECONOMICA I	MB OR 93-1	1650816	1352	1
CUARTO SEMESTRE					
208 0004 04	ANALISIS D EDOS FINANCIEROS	B OR 93-2	1710531	2452	1
208 0164 08	ESTADISTICA I	B OR 93-2	1710548	2451	1
208 0294 06	HIST ECONOMICA D MEXICO I	S OR 93-2	1710566	2451	1
208 0364 06	ECONOMIA POLITICA IV	B OR 93-2	1710580	2452	1
208 0371 04	CIES IV	B OR 93-2	1710599	2453	1
208 0815 06	TEORIA ECONOMICA II	B OR 93-2	1710627	2451	1
QUINTO SEMESTRE					
208 0099 04	CONTABILIDAD SOCIAL	B OR 94-1	1765590	1551	1
208 0165 08	ESTADISTICA II	B OR 94-1	1765602	1552	1
208 0295 04	HIST ECONOMICA D MEXICO II	MB OR 94-1	1765642	1552	1
208 0365 06	ECONOMIA POLITICA V	MB OR 94-1	1765669	1551	1
208 0372 04	CIES V	MB OR 94-1	1765698	1553	1
208 0816 06	TEORIA ECONOMICA III	B OR 94-1	1765729	1551	1

NOMBRE: ACEVEDO PEA OFELIA
 PLANTEL: E.N.E.P. ACATLAN ECONOMIA
 CARRERA: LIC EN ECONOMIA
 GENERACION: 90
 SEMESTRE: 94-1

No. CUENTA: 9058174-8
 CLAVE: 208
 CLAVE: 21

REPORTE DE AVANCE EN CREDITOS				REPORTE DE ASIGNATURAS	
	Cubiertos	Totales	%Avance	APROBADAS	: 54
Obligatorios	262	300	87.33%	NO APROBADAS:	0
Optativos	14	0	0.00%	TOTAL	: 54
Totales	276	300	92.00%	PROMEDIO	: 9.18

PRIMER SEMESTRE					
208 0241 06	GEOGRAFIA ECONOMICA	MB OR 90-1	1353260	1102	1
208 0291 06	HISTORIA ECONOMICA GENERAL I	MB OR 90-1	1353271	1102	1
208 0322 06	INTRODUCCION A LA ECONOMIA	MB OR 90-1	1353299	1102	1
208 0361 06	ECONOMIA POLITICA I	MB OR 90-1	1353310	1102	1
208 0368 04	CIES I	MB OR 90-1	1353342	1102	1
208 0497 08	MATEMATICAS I	B OR 90-1	1353369	1102	1
SEGUNDO SEMESTRE					
208 0292 06	HIST ECONOMICA GENERAL II	MB OR 90-2	1401357	2202	1
208 0362 06	ECONOMIA POLITICA II	MB OR 90-2	1401372	2201	1
208 0369 04	CIES II	B OR 90-2	1401398	2202	1

Migración de la información al ambiente UNIX.

208 0498 08	MATEMATICAS II	B EX 90-2 1414821	EA01 1 1
208 0500 06	METODOLOGIA DE LAS C SOCIAL	MB OR 90-2 1401428	2201 1
208 0755 06	SOCIOLOGIA Y POLITICA I	MB OR 90-2 1401437	2202 1
	TERCER SEMESTRE		
208 0293 06	HIST ECON DE AMERICA LATINA	MB OR 91-1 1457315	1301 1
208 0363 06	ECONOMIA POLITICA III	MB OR 91-1 1457352	1301 1
208 0370 04	CIES III	MB OR 91-1 1457385	1301 1
208 0499 08	MATEMATICAS III	S OR 91-1 1457414	1302 1
208 0756 06	SOCIOLOGIA Y POLITICA II	B OR 91-1 1457421	1302 1
208 0814 06	TEORIA ECONOMICA I	MB OR 91-1 1457427	1301 1
	CUARTO SEMESTRE		
208 0004 04	ANALISIS D EDOS FINANCIEROS	S OR 91-2 1500787	2401 1
208 0164 08	ESTADISTICA I	MB OR 91-2 1500813	2401 1
208 0294 06	HIST ECONOMICA D MEXICO I	MB OR 91-2 1500829	2401 1
208 0364 06	ECONOMIA POLITICA IV	MB OR 91-2 1500840	2401 1
208 0371 04	CIES IV	MB OR 91-2 1500863	2404 1
208 0815 06	TEORIA ECONOMICA II	S OR 91-2 1500897	2401 1
	QUINTO SEMESTRE		
208 0099 04	CONTABILIDAD SOCIAL	B OR 92-1 1552689	1501 1
208 0165 08	ESTADISTICA II	MB OR 92-1 1552702	1501 1
208 0295 04	HIST ECONOMICA D MEXICO II	MB OR 92-1 1552742	1502 1
208 0365 06	ECONOMIA POLITICA V	MB OR 92-1 1552768	1501 1
208 0372 04	CIES V	MB OR 92-1 1552796	1503 1
208 0816 06	TEORIA ECONOMICA III	MB OR 92-1 1552828	1502 1
	SEXTO SEMESTRE		
208 0005 04	ANAL DEL CAMBIO SOC EN MEXCO	B OR 92-2 1606348	2601 1
208 0131 04	DESARROLLO Y SUBDESLELO I	MB OR 92-2 1606354	2601 1

El resultado: historias académicas que son vistas en cualquier monitor de PC conectado a red.

Con este último programa se cierran las pruebas de depuración de la información y solo resta la implementación para ser visualizadas y ser consultadas vía red.

2.

Implementación.

En los capítulos anteriores se habló sobre el método que se sigue actualmente en el

almacenamiento de historias, se realizó una descripción del proceso seguido, así como de los equipos involucrados en él.

Basándonos en las necesidades de los usuarios y utilizando los recursos con los que en la actualidad la Universidad, como son la existencia de una red interna conectada a Internet y la adquisición de estaciones de trabajo más poderosas, se realizó la implementación de una aplicación que permitiera mejorar el proceso que actualmente se tiene.

Como se planteará a lo largo de este punto, emplearemos el modelo cliente/servidor para el diseño de la aplicación, utilizando como es obvio la infraestructura de red con la que cuenta la UNAM.

Este sistema constituye uno de los pasos importantes hacia la modernización de los sistemas de administración escolar y se ha querido realizarlo con base a los recursos disponibles dentro de la Universidad y los que se ofrecen como servicio público en la red internacional.

2.1 Descripción de los sistemas de recuperación y búsqueda.

Una de las partes medulares de la aplicación es el sistema empleado para realizar de manera eficiente y rápida la búsqueda y recuperación de la información. Como ya se explicó anteriormente se estudiaron varias opciones, como el manejar un DBMS, programas en el mercado realizados específicamente para realizar búsquedas en CD - ROMs, etc.

Después del análisis realizado se llegó a la conclusión que utilizando un software de dominio público se llegaba a los mismos resultados y con una eficiencia bastante aceptable. Es por esto que se trabajó en la adaptación de los programas fuentes de dos de las aplicaciones más importantes dentro de la Internet: **WAIS y GOPHER**.

El sistema **WAIS** se encarga de realizar indexamientos a la información y a través de lo que se denomina **search engine** que permite realizar las búsquedas sobre la información indexada. Este **search engine** no sólo trabaja sobre WAIS sino que puede enlazarse con otras herramientas para que éstas puedan realizar búsquedas sobre la información.

Precisamente la herramienta GOPHER es quien provee la interfaz adecuada al usuario para poder acceder a la información. A través de pantallas con menús de navegación permite acceder a una gran variedad de servicios, entre ellos WAIS, de una manera más amigable para el usuario.

De la utilización y adaptación de estas dos herramientas nace parte de nuestro sistema para la realización de búsquedas y recuperación de historias académicas. Sin embargo concientes de que en la Universidad cada día aumenta el número de equipo de cómputo con mayores capacidades gráficas y de que el usuario está cada día más acostumbrado a utilizar interfaces GUI, buscamos una herramienta más poderosa: **MOSAIC**.

Como se describirá más adelante MOSAIC cada día va cobrando mayor fuerza como una de las herramientas que ha venido a facilitar el uso de todos los servicios de la Internet, ofreciendo una interfaz gráfica completa para los usuarios.

2.1.1 Aplicación de WAIS al indexamiento y búsqueda de historias.

INTRODUCCIÓN

El éxito y gran desarrollo de la red Internet se debió principalmente a que permitió el acceso a información de una manera eficiente y con un gran alcance. Sin embargo este desarrollo implicó no sólo un incremento en el número de máquinas conectadas sino en la cantidad y variedad de la información disponible. Debido a ello surgió la necesidad de crear una herramienta que permitiera realizar búsquedas y recuperación de información en esta red. Con la nueva implementación se deseaba cubrir las siguientes necesidades:

- Interfaz amigable.
- Manejo de información de fuentes múltiples como imágenes, sonidos, etc.
- Localización de fuentes de información

- Implementación de un modelo de acceso múltiple sin un lenguaje especializado.
- Protocolo abierto de conexión de la interfaz de usuario al servidor.

Una propuesta de solución a este problema es *Wide Area Information Server* o mejor conocido como **WAIS**. WAIS es un aplicación de red que permite realizar peticiones a bases de datos de multimedia (generalmente son de texto) utilizando un estándar de consultas y un protocolo de recuperación denominado Z39.50. Uno de sus grandes beneficios es que proporciona una interfaz de usuario común a una gran variedad de fuentes de información residentes en cualquier parte de Internet.

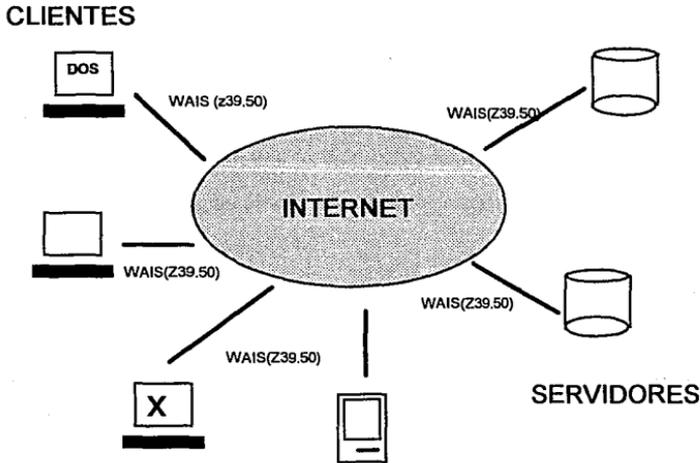
WAIS constituye uno de los ejemplos más recientes de aplicaciones de red que ayudan a que Internet sea un recurso más útil para los usuarios no relacionados con el campo de la computación, debido a la facilidad de su interacción con el lenguaje natural. Además viene a representar una herramienta poderosa que permite a los proveedores de información hacer accesibles sus datos.

PERSPECTIVA DE WAIS

El proyecto de WAIS fue desarrollado por un equipo encabezado por Brewster Kahle en Thinking Machines Corporation. El sistema está compuesto de tres partes:

- **CLIENTES.**
- **SERVIDORES.**
- **PROTOCOLO QUE LOS CONECTA.**

El **CLIENTE** es la interfaz de usuario se encarga de realizar las peticiones y mostrar los resultados, el **SERVIDOR** realiza el indexamiento y la recuperación de documentos y el **PROTOCOLO** es utilizado para transmitir las consultas y respuestas. El cliente y el servidor están separados a través del protocolo quien es el que se encarga de la comunicación entre ellos. Esto se puede ver en la siguiente figura:



Del lado del cliente las consultas son formuladas en inglés. La aplicación de cliente traduce la petición al protocolo de WAIS y lo transmite. El servidor recibe la transmisión, traduce el paquete recibido y busca documentos que cumplan con las condiciones de la consulta. El algoritmo de ocurrencias en WAIS se maneja con base en el número de ocurrencias de las palabras que forman parte de la petición en el documento, la localización de las palabras en el documento, la frecuencia de ellas dentro del conjunto y el tamaño del documento.

ARQUITECTURA DEL PROTOCOLO

Z39.50 es un Estándar Nacional Americano que fue aprobado en 1988 por NISO (National Information Standards Organization) un comité acreditado por ANSI. Este organismo se encarga de desarrollar y promover los estándares utilizados en los servicios bibliotecarios, edición y otras industrias relacionadas con la información.

Direcciona las tendencias en áreas como la recuperación de información, preservación de materiales, transferencia de información, formas y registros, identificación de sistemas, formatos de publicación y equipo.

Es un protocolo de la capa de aplicación en el modelo OSI, cuyo propósito es permitir que una computadora operando como un cliente pueda realizar recuperación de peticiones específicas de información. El propósito de esta interfaz es establecer un protocolo en la capa 2 del modelo OSI para las aplicaciones de búsqueda/recuperación.

Este protocolo está basado en el estándar Z39.50-1988 el cual es una definición en el servicio de recuperación de información y una especificación para las aplicaciones bibliotecarias. Este estándar especifica un servicio en la capa de aplicación de OSI y un protocolo para la recuperación de información. El protocolo IR permite que una aplicación en una computadora haga peticiones a la base de datos de otra. El protocolo especifica los procedimientos y estructuras para la conformidad de requerimientos de búsqueda (incluyendo la sintaxis de una petición), requerimientos para la transmisión de registros de bases de datos

base de datos de otra. El protocolo especifica los procedimientos y estructuras para la conformidad de requerimientos de búsqueda (incluyendo la sintaxis de una petición), requerimientos para la transmisión de registros de bases de datos localizados por una búsqueda, la respuesta del requerimiento, control de acceso y control de recursos.

Existen dos categorías de cambios de la versión de 1988 y la de 1992 (versión 2). Estos cambios fueron necesarios para seguir la misma línea que **SR (Search and Retrieval Service Definition and Protocol Definition)**, que es un estándar de ISO, implementar formas necesarias para los desarrolladores y para proveer la suficiente funcionalidad para que su implementación fuera económicamente justificable.

Los cambios principales fueron:

- Utilización de ASN.1 (este es un estándar de OSI para describir estructuras de datos).
- Identificador de objetos.
- Seguridad
- Control de acceso

Base de datos

WAIS utiliza para realizar las búsquedas un indexador diseñado por Brewster Kahle denominado Indexador Serial, el cual es un simple archivo invertido no muy diferente de los sistemas IR existentes.

El indexador serial analiza los archivos o el archivo de entrada y crea una base de datos compuesta de 7 archivos la cual, es el índice utilizado para realizar las búsquedas. Si suponemos que el nombre base de nuestro índice es "**hist**" los archivos tendrán los siguientes nombres:

hist.inv	También conocido como "el archivo de registros", el cual consta de un término seguido por un conjunto de registros donde cada uno describe donde se encuentra la ocurrencia de ese término en los archivos originales. Un registro está formado por un peso, un número de documento (doc) y un indicador de posición. El archivo es indexado con el archivo de diccionario (dct) y los términos se ordenan alfabéticamente.
-----------------	---

hist.doc	<p>Esta es una lista lineal con una entrada por cada documento. Un documento puede ser un archivo completo o una pieza de un archivo (como los archivos de correo ya que es una concatenación de muchos mensajes donde cada mensaje debe ser manejado como un documento independiente). La información que guarda el archivo es la siguiente:</p> <p><i>id_nomarch</i>: posición dentro del archivo del documento.</p> <p><i>filename_id</i>: posición dentro del archivo de nombres (fn).</p> <p><i>headline_id</i>: posición en el archivo de encabezados.</p> <p><i>start_character</i>: posición en el archivo donde este documento comienza.</p> <p><i>end_character</i>: posición donde termina el documento. El valor es 0 si se trata de un archivo completo.</p> <p><i>document_length</i>: longitud del documento en caracteres.</p> <p><i>number_of_lines</i>: número de líneas de las que consta el documento.</p> <p><i>date</i>: fecha de indexación.</p>
-----------------	---

hist.fnm	registro de los nombres de archivos en la base con la fecha de archivo y tipo de archivo. El tipo es una cadena. Indexado por la posición en el archivo, para que este archivo no pueda ser editado después de que el índice sea construido.
hist.hlm	lista de los encabezados, Indexado por posición.
hist.dct	archivo que contiene el diccionario el cual es un árbol binario de dos niveles. El primer bloque son apuntadores que corresponden uno por cada 1000 entradas del resto del diccionario. Cada entrada es un registro de longitud fija de la palabra con la posición dentro del resto del archivo. El resto del archivo son bloques como los del primer bloque, pero cada entrada es una palabra más la posición dentro del archivo invertido (inv). Todo el diccionario está en orden alfabético.
hist.src	descripción utilizada para acceder la base de datos. También es regresada como respuesta al query "help". Este archivo no es sobrescrito una vez que está creado. Sin embargo los encargados de la base debe editarlo para agregar una buena descripción de lo que esta base contiene.
hist. status	este archivo lo utiliza la herramienta de búsqueda en RAM para obtener parámetros del usuario.

Indexamiento

Un nuevo índice es construido al realizar el análisis de los archivos de entrada, encontrando las palabras en ellos y creando las tablas de nombres de archivos, encabezados y documentos. Las palabras son pasadas entonces a rutinas definidas en *irext.h* las cuales definen los límites del **frontend** (interfaz al usuario) y del **backend** (programa de indexamiento).

El indexador serial crea archivos invertidos intermedios llamados *inv0*, *inv1*, etc. Estos son creados por la acumulación de palabras en una tabla hash en memoria. Cada archivo *invN* se encuentra en orden alfabético, de tal manera que puedan ser mezclados fácilmente para formar el archivo *inv*. Esta mezcla se realiza logarítmicamente, para que sea más rápida. Desafortunadamente, esto significa que antes que la mezcla se realice, el índice ocupa el doble de espacio del que tendrá al final.

En la mezcla final, el archivo de diccionario es generado al realizar el vaciado de las posiciones de comienzo de los términos en el archivo inverso final.

El agregar información a una base de datos existente se realiza al agregar información en las tablas de nombres de archivos, encabezados y documentos al momento del análisis, pero esperando para mezclar los nuevos archivos *invN* al viejo archivo *inv* hasta que toda la nueva mezcla esta hecha. Teóricamente es posible hacerlo mientras la base esta en uso.

Búsqueda

Una consulta (query) es analizada para extraer las palabras buscadas y los documentos relevantes y pasa a través de la función `search_word` (definida en `irext.h`). El backend hace todo lo que sea necesario para realizar la búsqueda y regresar algún estado. Entonces la función `best_hit` es llamada iterativamente para encontrar el documento que mejor se acerca a la consulta.

Estos resultados se utilizan para buscar y encontrar los encabezados y nombres de archivos que va a regresar al usuario.

El indexador serial realiza esto al cargar la puntuación para un término en particular y sumándolos dentro de un arreglo de puntos totales. Este arreglo tiene un registro para cada documento. Un documento aumenta su puntuación si contiene términos de la consulta.

Los puntos importantes de este diseño son:

- Realizar una plataforma flexible para la experimentación.
- Portable
- Bajo overhead de búsqueda y acceso
- Búsqueda rápida
- Fácil de implementar.
- El tamaño de la base resultante no es una prioridad.

CONFIGURACION DE UN SERVIDOR

1. Manipular la información de tal manera que se encuentre en un formato entendible para el indexador de WAIS llamado *waisindex* o realizar las modificaciones necesarias al programa fuente para que reconozca el formato de la información. Cabe mencionar que *waisindex* actualmente maneja cerca de 25 formatos de documentos.

2. Buscar un directorio donde la capacidad sea la suficiente para almacenar los índices y la información. No se puede mover o cambiar la información fuente una vez que los índices son construidos.

3. Ejecutar el indexador **waisindex** en el directorio donde se encuentra la información. Se debe tomar en cuenta que genera un índice de todo el texto completo por lo que se requiere el doble de espacio.

Implementación en el indexamiento de historias académicas

El primer paso para realizar el indexamiento de las historias académicas fue determinar el formato que se iba a utilizar para la información. Las historias académicas se encuentran en un sólo archivo, por lo que se eligió uno de los formatos de waisindex que permite tratar como documentos independientes los párrafos separados por un retorno de carro dentro de un sólo archivo. Para el caso de las historias académicas, se implementó un programa que separa las historias de cada alumno por un retorno de carro, de esta manera al realizar el indexamiento y la posterior recuperación sólo se recupera la historia que cumple con la petición, no todo el documento.

A continuación presentamos la descripción de la herramienta de indexación:

NOMBRE

waisindex Indexa archivos.

SINTAXIS

```
waisindex [ -d nombre_arch_indices ] [ - a ] [ - r ] [ - mem ]
           [ -t tipo ] archivo1 [archivo2 ... ]
```

DESCRIPCION

Waisindex crea un índice de las palabras en los archivos de tal forma que puedan ser buscados de una manera eficiente por la herramienta **waissearch**. El índice requiere tanto espacio en disco como lo requiere el archivo original.

OPCIONES

-d nombre_arch_indices

Capítulo III Diseño de la propuesta.

Este nombre es el nombre de la base que van a tomar todos los archivos generados por `waisindex` y es el nombre a través del cual se va a hacer referencia para realizar las búsquedas.

- a Agrega información a un índice ya existente.
- r Realiza indexamiento recursivo de directorios.
- t tipo

Aquí se especifica el tipo de documento que se va a indexar. En las nuevas implementaciones se pueden manejar hasta 25 formatos de documentos. A continuación se muestran los más comunes:

- text** Información tipo texto
- ftp** documentos que se encuentran en servidores de ftp.
- gif** imágenes en formato gif.
- html** formato utilizado por Mosaic.
- para** párrafos separados por un retorno de carro y que para el indexador son tratados como documentos individuales.
- ps** formato Postscript.
- tiff** imágenes con formato tiff.

2.1.2 Aplicación de GOPHER a la visualización de historias.

Así como la Internet creció en cantidad de información disponible, también se fue expandiendo cada vez más, contando con un número creciente de hosts. Por este motivo no sólo fue importante el poder buscar una información específica, sino saber, dentro de este gran número de máquinas, en donde se encuentra.

Debido a ello fue necesario desarrollar una nueva herramienta que de manera transparente para el usuario, nos permitiera el acceso de todo tipo de información que se encuentra distribuida a lo largo de diferentes hosts de la red.

GOPHER combina las características de los servicios de **BBS** y la de las bases de datos para realizar desplazamientos a lo largo de varios documentos, así como búsquedas de documentos que contengan ciertas palabras o frases.

Este servicio fue desarrollado en el departamento de Computer and Information Services de la Universidad de Minnesota.

TIPOS DE INFORMACIÓN

GOPHER maneja una gran variedad de tipos de datos, donde cualquiera de ellos puede ser accesado con una tecla o un simple click del mouse. A continuación se presentan los tipos más comunes:

- DIRECTORIOS
- ARCHIVOS DE TEXTO
- TEMAS DE BUSQUEDA
- SESIONES DE TELNET
- DIRECTORIOS TELEFONICOS
- MULTIMEDIA
- TEXTO FORMATEADO

DIRECTORIOS	Esta es la unidad básica de información. Un directorio es una lista de documentos. Los directorios ofrecen una forma fácil de realizar desplazamientos a lo largo de temas de información. Los temas son organizados en áreas específicas, lo que hace más fácil encontrar la información.
ARCHIVOS DE TEXTO	Contienen en formato ASCII la información.
TEMAS DE BUSQUEDA	Este es un tipo especial de directorio. Se puede especificar palabras llaves para buscar un tema. Sólo los documentos que concuerdan con el criterio establecido son presentados.
SESIONES DE TELNET	Este punto es una referencia a una sesión remota. Seleccionarlos nos permite una conexión a un sistema utilizando una emulación de terminal.
DIRECTORIOS TELEFONICOS	Este documento es un tipo especial de búsqueda basado en el servidor Ph/Qi desarrollado en la Universidad de Illinois. Este proporciona un formato especial para el manejo de información de este tipo.

MULTIMEDIA	Soporta un gran número de archivos con formato de multimedia incluyendo imágenes, audio y video. En los E.U. se encuentran disponibles archivos con las imágenes del tiempo, archivos con los debates presidenciales en audio, video en tiempo real utilizando el formato MPEG.
TEXTO FORMATEADO	Varios servidores de de GOPHER soportan desplegar documentos que no se encuentran en formato texto como son documentos en postscript.

¿Cómo encontrar información?

Existen dos maneras: una se realiza a través de la búsqueda y la otra a través de desplazamientos.

Desplazamientos

Partimos de la raíz de Gopher y a partir de este nivel vamos seleccionando temas que nos va llevando a niveles más internos hasta que encontramos la información que necesitamos.

Búsqueda

De manera alterna, podemos indicarle a GOPHER que haga este trabajo por nosotros al especificarle un tema de búsqueda. Podemos seleccionar un tipo de dato de búsqueda de algún tema y nos preguntara por palabras llaves del tema

que nos interese y al final nos presenta una serie de documentos relacionados con la búsqueda.

CONFIGURACION DE UN SERVIDOR DE GOPHER

El primer paso es lograr la compilación de los programas fuentes de GOPHER, una vez logrado esto se debe dejar activo en memoria un proceso que atienda todos los requerimientos hechos por los clientes para acceder el servidor de GOPHER. A este proceso se le debe indicar el directorio de datos a partir del cual va a acceder la información.

Este proceso se llama *gopherd* y se levanta con el siguiente comando:

```
gopherd /datos_gopher 70
```

bajo el directorio */datos_gopher* se va a encontrar toda la información que el servidor va a poder mostrar a sus clientes, el número 70 indica el número de puerto a través del cual se va a acceder el servidor.

¿Cómo agregar información al servidor de gopher?

Una vez que se encuentra listo el proceso que atenderá las peticiones el siguiente paso es colocar la información. Como se ha mencionado antes esta información debe de ir en el directorio que para el caso se le ha especificado al proceso, para nuestro ejemplo el directorio */datos_gopher*.

Si deseáramos tener información sobre letras de canciones seleccionadas por géneros y años tendríamos que realizar lo siguiente:

- Cambiarnos al directorio /datos_gopher.
- Crear un directorio denominado canciones.
- Cambiarnos al directorio canciones.
- Crear directorios para los diversos géneros: jazz, rock, heavy metal, etc.
- Cambiarnos al directorio deseado en este caso heavy metal.
- Crear los directorios correspondientes a los años que deseamos tener.
- Cambiarnos al directorio del año deseado.
- Copiar en ese directorio los archivos que contienen las letras de las canciones, Dichos archivos se llamaran como el titulo de la canción.

Cuando se haga el acceso al servidor aparecerá un tema denominado canciones, al seleccionarlo aparecerá una pantalla que contendrá todos los temas de los cuales podremos seleccionar alguno de ellos. Al seleccionarlo aparecen los años y al seleccionarlo apareceran los titulos de las canciones de ese año. Al escoger una regresará el documento que contiene la letra de dicha canción.

Como se podrá observar los nombres de los directorios y archivos deberán ser alusivos al tema para que sea fácil la identificación por parte del usuario.

El servidor de GOPHER tiene la capacidad de realizar ligas a otras máquinas, y es precisamente esta característica la que le permite un acceso distribuido de información a lo largo de la red. La manera de definir estas ligas es a través de un archivo de liga cuyo nombre debe comenzar forzosamente con un punto para que sea reconocido como tal. Un archivo liga puede contener a su vez varias ligas. Este archivo consta de 5 líneas con el siguiente formato:

Name=Letras de canciones.

Type=1

Port=150

Path=1/canciones

Host=tzetzal.dcaa.unam.mx

La línea Name indica la sentencia que el usuario observará en el menú de temas.

La línea Type indica el tipo de documento que se trata. La siguiente lista nos muestra los tipos que se manejan:

0	Archivo de texto
1	Directorio
2	CSO servidor de nombres
7	Archivo indexado
8	Sesión TELNET
s	Sonido

La línea de Path contiene la ruta a través de la cual el cliente podrá acceder la información.

Las líneas de Port y Host indican el nombre del host que contiene la información así como el puerto a través del cual se podrá acceder la información.

Acceso a archivos indexados

El acceso a este tipo de información se maneja de una forma especial, se debe de crear un directorio cuyo nombre comience con un punto. Bajo ese directorio se deben crear los índices del archivo o archivos especificados a través de un programa de indexamiento como sería el waisindex de WAIS. La liga a ese directorio de índices se crea con el formato que se especificó en el párrafo anterior:

Name=Historias.

Type=7

Port=150

Path=7/.waisindex/historias

Host=tzetzal.dcaa.unam.mx

El contenido del directorio .waisindex contiene los archivos generados por el indexamiento. Como se explicó en el capítulo referente a WAIS el indexador denominado waisindex genera una serie de archivos con el nombre de la base especificado, para nuestro ejemplo, todos los archivos comenzarían con la palabra historias. Es a través de estos archivos y de las ligas que guarda GOPHER con WAIS que se logra realizar la recuperación de documentos.

BÚSQUEDA DE LOS PROGRAMAS FUENTES DE GOPHER.

Al investigar más sobre esta herramienta pudimos encontrar el *host* de la Universidad en donde se desarrolló, la Universidad de Minessota, este lugar

también es conocido como **Internet Gopher Software**, y mucho de este software está en la máquina *boombox.micro.unm.edu*. Esta máquina se encuentra en la red Internet, por lo que solo se necesita una máquina con **FTP** y que estuviera conectada a su vez a la Internet.

Al lograr la conexión con el servidor boombox nos cambiamos al directorio */pub/gopher*. Aquí se encuentran versiones para diferentes tipos de máquinas por ejemplo hay versiones para:

- Macintosh
- NeXTstep
- PC's
- Unix
- VMS
- MVS
- OS/2

para casi todas ellas hay tanto la versión de servidor como para el cliente y además se cubre la mayor parte de los tipos de computadoras más comunes en el mundo.

Una vez que se ha encontrado la versión que se ajusta a nuestro hardware y sistema operativo se pasa entonces a transferirlo a la máquina local, en nuestro caso se utilizó la versión **UNIX**.

DESCRIPCIÓN GENERAL DEL PROGRAMA FUENTE DE GOPHER.

A continuación pasaremos a dar una descripción muy general del funcionamiento de Gopher la cual se realizó con base en los programas fuentes. Pensamos que con esta explicación basta, ya que no es el caso de este trabajo ver que hace el programa paso a paso sino solamente como lo hace en general para poder implementarlo y utilizarlo.

En la primera parte del diagrama de flujo general encontramos que el primer paso es establecer la conexión para esto es necesario que por vía de *telnet*¹ una máquina se conecta a otra que sea servidor de Gopher. Una vez que se logra establecer la sesión el servidor pedirá un *login name* (nombre de alguna cuenta en el servidor) y un *password* (clave secreta) para poder entrar a él. Una vez hecho esto se corre un *login script* automáticamente que manda correr a su vez al cliente de Gopher. Pasamos al siguiente punto del diagrama que es cuando se llama al cliente de Gopher; primero se cerciora que el *daemon* este corriendo, este *daemon* es el programa residente en memoria que constituye en sí al servidor de Gopher. Si no lo encuentra manda un mensaje de error al usuario y lo saca de sesión; si lo encuentra entonces establece una sesión entre cliente servidor de Gopher e inmediatamente muestra el menú principal de éste.

Dentro de las opciones que podrá contener este menú están los tipos de objetos que puede manejar Gopher como lo son :

- Archivos de Texto
- Sonidos
- Directorios

¹ TELNET es un programa que permite establecer sesiones con máquinas remotas conectadas a red.

- Textos Indexados
- Sesiones Telnet

Pasemos ahora a explicar que sucede al escoger alguno de ellos: si se escoge un archivo de texto entonces presenta al archivo completo y al terminar se tiene la opción de salir en cuyo caso se sale de Gopher o de regresar al menú anterior, el cual no necesariamente es el menú principal.

Si es un sonido entonces buscará una utilería que pueda soportarlo además del periférico respectivo (por ejemplo una bocina); si no lo encuentra manda un mensaje de error pero si lo encuentra entonces se escuchará el sonido; en cualquiera de los casos se dará la opción de salir o de regresar al menú anterior.

Si es una sesión telnet entonces intentara la conexión; si esta resulta se conecta a otro host, al salir de ésta regresará a Gopher para tener la opción de salir o de continuar al menú anterior; si no se logra la conexión desplegara un mensaje de error.

Si es un directorio entonces desplegara un menú de archivos que se encuentran dentro de él, es por eso que no siempre es necesario regresar al menú principal si es que existe alguno intermedio.

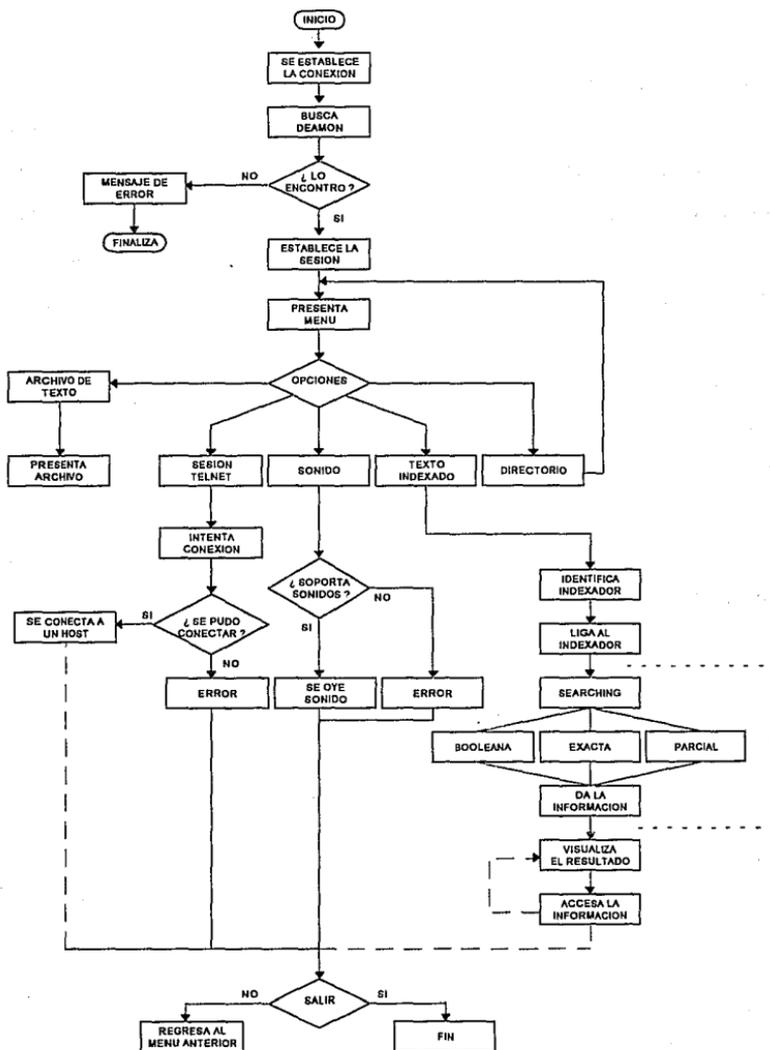
Pasando al tema que más nos interesa que es el de los archivos indexados, debemos saber que al solicitar este servicio Gopher identifica el indexador utilizado en el archivo (en nuestro caso particular es WAIS), para poder hacer una liga a él. A partir de aquí² pasa el control al indexador que permite hacer

² En el diagrama esta parte se encuentra marcada con líneas punteadas.

búsquedas (searching) ya sean booleanas exactas o parciales. Una vez que la encuentra entonces da la información junto con el control a Gopher para que este pueda visualizar el resultado en forma de menú de todas las ocurrencias sobre el tópico solicitado. Ahora bien, si se completo correctamente la búsqueda de la información solicitada será posible su despliegue, regresar al menú de resultados y acceder la información de otra, y así sucesivamente hasta que se decida regresar al menú anterior o salir.

Cabe señalar que al terminar la sesión de Gopher que el *login script* inició, se encarga también de terminar el proceso cliente/servidor del Gopher y de la conexión con el equipo.

A continuación se presenta el diagrama de flujo para mayor claridad de la explicación anterior:



2.1.3 Aplicación MOSAIC como interfaz gráfica para el usuario.

Como vimos anteriormente, las aplicaciones de **WAIS** y **GOPHER** nos permiten presentar al usuario, por un lado, un método muy rápido de indexamiento total de documentos y búsqueda de información (que es el caso de **WAIS**) y por otro lado presentar una interfaz muy sencilla y poderosa para la visualización de la información (que es el caso de **GOPHER**), podemos decir entonces que **GOPHER** tiene como *search engine* a **WAIS**.

Se puede agregar que estos dos programas constituyen de hecho, **estándares** para la consulta masiva de información a nivel red, un caso más concreto a nivel **Internet**. Muchos de los servicios que provee esta **red de redes** están basados en estas implementaciones. Desde consultas bibliográficas, servicios meteorológicos hasta consulta de la programación de conciertos en las diferentes salas del Centro Cultural Universitario (estos servicios pueden ser accedidos consultando en **GOPHER** que se tiene implementado en la máquina **condor.dgsca.unam.mx** que se encuentra en la Dirección General de Servicios de Cómputo Académico). Debido a que estos programas fueron diseñados para la consulta de información en red, donde cualquiera se puede conectar, se puede decir que estos son **servicios públicos**. Sin embargo se debe de considerar que la implementación que estamos llevando a cabo no debe de ser pública, es decir, no cualquiera puede tener acceso al servicio de consulta de historias académicas, es por ello que se implementó un nivel de seguridad, sin embargo dentro de este **GOPHER** tenemos la posibilidad de conectarnos a cualquier otro dentro de RedUNAM o de la Internet.

Aunque estos dos programas parecen haber resuelto todo el problema siguen siendo interfaces muy simples para el usuario, es decir, no presentan una interfaz **gráfica** como lo provee **MS Windows** u **OpenWindows** en el caso de Solaris.

Dentro de la investigación hecha para la realización de este trabajo se buscó una interfaz que fuese gráfica y de fácil uso. Debemos recordar que la tecnología siempre va adelante y que en un futuro, la mayoría de las aplicaciones presentarán esta interfaz.

Si lo vemos desde una perspectiva histórica podemos tomar el ejemplo del sistema operativo **DOS** y las interfaces **GUI**. Todo empezó en 1977 cuando ya se tenía en mente presentar al usuario una nueva interfaz más amigable al usuario que el simple **C:** que presentaba DOS. En 1982 en Las Vegas se hizo una demostración de lo que podía significar una interfaz gráfica al usuario con la característica **WYSIWYG**, es decir *what-you-see-is-what-you-get* ("lo que ves es lo que obtienes"). Después de esto, todos los fabricantes de software querían incluir esta característica en sus equipos. Así en enero de 1983 Apple anuncia su Apple Lisa. Para noviembre de ese mismo año Microsoft anuncia **Windows** y que próximamente iba a ser liberado. De esta forma comienza una especie de "guerra" por la mejor interfaz gráfica. En enero de 1984 Apple anuncia su **Apple Macintosh** y DEC anuncia su **Graphics Environment Manager**. En noviembre de 1985 Microsoft lanza **Windows 1.0**. En 1988 fue un año de controversias, ya que Apple demanda a Microsoft y HP por una violación de derechos de autor sobre su interfaz gráfica diseñada para su **Macintosh**. En ese mismo año **IBM** lanza **OS/2**.

Como podemos ver, las interfaces gráficas ganan cada vez, más terreno; con esto no queremos decir que una es mejor que otra, simplemente es un ejemplo de

lo que significa tener una interfaz de este estilo para el usuario, ya que para él es más fácil manipular un **ícono** en la pantalla que recordar un comando. El ser humano es un ser visual por excelencia.

Tomado en cuenta esto, nuestra investigación nos llevó hasta **MOSAIC**. Dentro de la comunidad de la Internet se ha empezado a sentir una inquietud con respecto a interfases más amigables de las herramientas de la Internet. En medio de todo esto se encuentra **MOSAIC**, una aplicación **de dominio público, sin costo** que tiende a convertirse en un estándar y definirá cómo el usuario podrá obtener información.

Producido por el **National Center for Supercomputing Applications (NCSA)**, **MOSAIC** provee de un "buscador" dentro de **WWW** (explicado en el capítulo de *Teoría Básica*) que puede traer documentos de **hipertexto** así como otras formas de texto, gráficas, imágenes, imágenes en movimiento (películas) y audio de **WWW, Gopher** y servidores de **FTP**. Además este puede acceder estos formatos en una red local o sistemas locales.

La ventaja que presenta esto es que puede funcionar en varias plataformas, la más importante para nuestro trabajo es que puede correr en la mayoría de los sistemas **UNIX**, además de correr en **Apple Macintosh** y PC's corriendo **MS Windows**.

Para nosotros, administradores de sistemas y usuarios no necesitamos saber mucho de él, ya que éste provee de un método uniforme para acceder diferentes tipos de información. Protege al usuario de hacer trabajo innecesario para traer información y para su consulta (como por ejemplo en el FTP anónimo, ya que en

este se debe de tener conocimiento de la dirección IP de la máquina o su nombre, además de los comandos básicos para su manejo). Para los administradores significa menos "*lee el manual*" para responder respuestas del usuario, y para el usuario menos "*¿dónde está el administrador del sistema?*"

Mosaic se encuentra disponible, sin cargos, via FTP anónimo en la siguiente dirección: `ftp.ncsa.uiuc.edu`.

El software esta ubicado bajo el subdirectorio `/Mosaic`. Es bueno primero leer el documento `README` el cual contiene una explicación del contenido de los diferentes subdirectorios ahí ubicados. Así dentro de `/Mosaic/Mosaic-binaries` encontramos el programa para nuestro sistema operativo, **Solaris** y también el programa para **PC's** corriendo **DOS**. El programa asume que en la máquina donde se va a correr tiene acceso a la red Internet, lo cual es obvio ya que el potencial del programa reside en tener acceso a esta red, la **UNAM** tiene acceso a esta red.

DESCRIPCIÓN DE MOSAIC

A continuación se dará una descripción técnica de las posibilidades de **Mosaic**.

Como hemos dicho anteriormente **Mosaic** fue desarrollado en el NCSA de la Universidad de Illinois, este programa provee el acceso a un universo de información ya sea en sistemas conectados a la red Internet o simplemente sistemas *stand alone* como en una red local, puede acceder información como parte de un **ambiente de colaboración asíncrono** entre usuarios.

Mosaic utiliza una mezcla de **hipermedios distribuidos**, organización jerárquica y búsquedas funcionales de información que se basan en el "descúbralo y accéselo". Esta diseñado también para funcionar en redes tipo **WAN** y manejar diferentes tipos de formatos y protocolos para que el usuario pueda acceder a todo tipo de información.

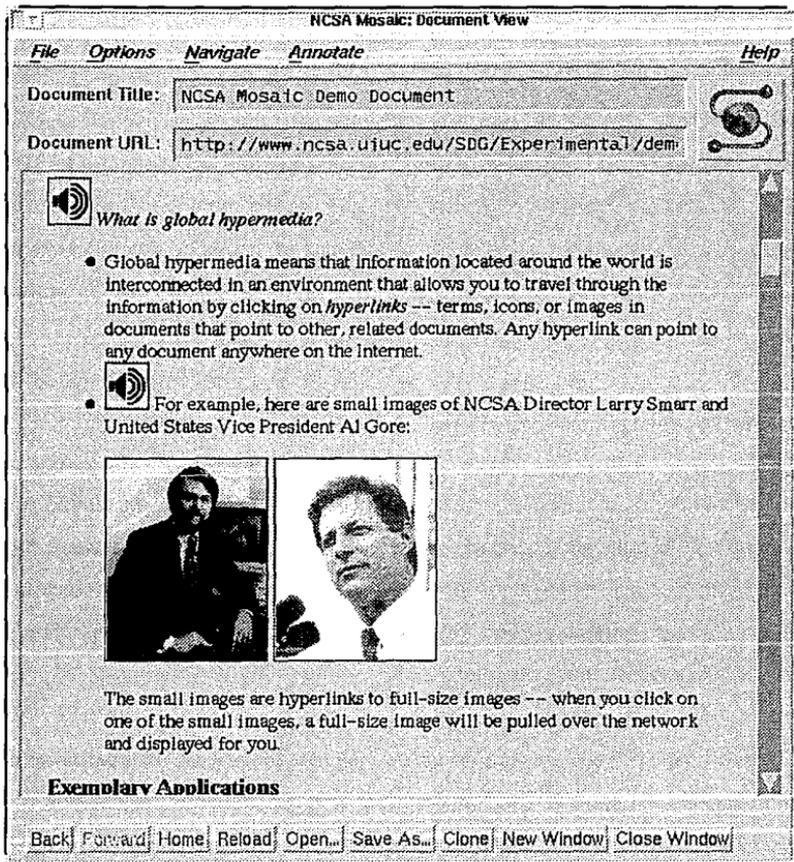
El programa se basa en una arquitectura de **cliente/servidor** para la distribución de la información: un **servidor** que se encuentra en una máquina en la Internet puede manejar un **query** (consulta) de un **cliente** ubicado en cualquier punto de la misma red. Las unidades de infomación que los servidores mandan a los clientes se les denomina **documentos** . Estos documentos pueden contener simple texto, texto preformateado, gráficas dentro del documento, sonidos e información tipo **multimedia** como puede ser imágenes en movimiento, fotos fijas, etc.; además pueden contener **hiperligas** a otros documentos que pueden llevar al usuario a otros sitios dentro de la Internet.

Las versiones iniciales de los clientes de **Mosaic** tienen las siguientes capacidades:

- Despliegue gráfico de texto (simple o formateado), hipertexto así como imágenes o gráficas dentro del documento, audio y secuencias de video así como la posibilidad de manejar información del tipo científica en forma de multimedia o hipermedia.
- Una interfaz gráfica inteligente que permite al usuario las funciones básicas de un ambiente de este estilo, selección de tipos de letra, **cut and paste** de texto formateado y demás funciones inherentes al usuario.

- Información de la navegación entre documentos, historia de los mismos y mecanismos de historia general.
- Soporta una lista de "hosts" (máquinas en la red) de documentos interesantes para el usuario que pueden ser monitoreados por posibles cambios o actualizaciones.
- Búsqueda de información incluyendo accesos a **WAIS y GOPHER**.
- Capacidades de colaboración asíncrona, incluyendo texto y anotaciones sonoras.
- Soporte de comunicaciones basado totalmente en **TCP/IP** incluyendo como protocolo nativo **HTTP , Gopher, FTP, WAIS, Archie, etc.**

Cada vez que el programa es invocado, este presenta un **home page** es decir, una hoja de presentación del primer punto donde arranca, normalmente el **home page** de la Universidad de Illinois. Cada palabra subrayada significa una **hiperliga** a otro documento. A continuación se presenta el documento de demostración de NCSA:



Mosaic provee de capacidades de hipermedia que toman ventaja de toda la información almacenada en la Internet. Cada pieza de información, cualquier documento, puede tener un nombre **único** que lo identifica en la red, por lo que puede ser ubicado, examinado, accedido, hacer anotación sobre él, recordado, etc. por **Mosaic**.

El esquema que utiliza para la indentificación de toda esta gama de documentos se llama **Uniform Resource Locator (URL)** el cual fue diseñado para identificar de manera única y con una nomenclatura específica cada documento en la red. **URL** puede identificar documentos accedados vía FTP o **HTTP (HyperText Transfer Protocol, explicado posteriormente)**, servidores de noticias, mandar **queries** a WAIS para búsquedas en bases de datos del mismo, documentos en GOPHER utilizando **search engine** de WAIS (que es nuestro caso), etc.

¿ Qué tenemos como resultado de todo esto ?

El resultado es una completa transparencia para el usuario en cuanto a la búsqueda y consulta de cualquier documento: **Mosaic** puede acceder a toda esta información sin meter al usuario en complicaciones técnicas inherentes al manejo de documentos y conexión a diferentes máquinas. Simplemente se interactua con la información y ¡ listo !

Otra de las ventajas que anteriormente se describian era la **colaboración asíncrona** entre usuarios. Esto quiere decir que los usuarios distribuidos a lo largo de la red puedan trabajar colectivamente en un proyecto o en la construcción de artículo en el formato HTML.

Una vez conocidas las ventajas y bondades de este programa solo resta explicar dos aspectos descritos anteriormente: **URL y HTML**.

¿ Qué es un **URL** ? URL es un **Uniform Resource Locator**, es decir, una forma de indentificación de archivos a lo largo de la red. Pensemos un momento

que es la *extensión en red* del concepto de **nombre de archivo**: uno no solamente puede identificar a un archivo en un subdirectorio cualquiera, sino que se debe especificar en que máquina se encuentra y máxime si se encuentra dentro de la Internet. Este archivo o **documento** puede ser accedido por diferentes medios y en algunos casos no simplemente se trata de una archivo plano: los **URL's** pueden apuntar a **queries**, documentos almacenados en bases de datos, resultados de un **archie**, etc.

El concepto de esto es realmente simple: "si existe en la red, uno puede apuntar a él". A continuación se presentan algunos ejemplos de ello.

- Archivos URL: supongamos que el archivo `leeme.txt` se encuentra en un servidor llamado `ftp.servidor.unam.mx` bajo el directorio `/pub/archivos`. El URL respectivo sería el siguiente:

```
file://ftp.servidor.unam.mx/pub/archivos/leeme.txt
```

Si queremos acceder a todos los archivos del directorio `/pub` sería:

```
file://ftp.servidor.unam.mx/pub
```

- Gopher URL: para el caso de un gopher se tiene una diferencia, ya que los servicios de Gopher se basan en puertos, por ejemplo para el servicio que se está implementando sería:

```
gopher://tzetzal.dcaa.unam.mx/
```

En el caso de que se tuviera corriendo el servicio en el puerto por default que es el 70. Si se tiene en algún otro puerto se tiene lo siguiente:

`gopher://tetzal.dcaa.unam.mx:1300/`

En este caso se esta especificando que se realice la conexión por el puerto 1300.

- HTTP URL's: asimismo para la transferencia de documentos mediante el protocolo de transporte de hipertexto se tiene lo siguiente:

`http://tetzal.dcaa.unam.mx/~edgar/edpres.html`

Esto significa que se accederá el documento que se encuentra en el **home directory** de edgar llamado `edpres.html`. De hecho esta es la página de presentación del usuario Edgar Valencia Figueroa. En el caso de que el servicio este corriendo en algún otro puerto se especifica:

`http://tetzal.dcaa.unam.mx:1300/~edgar/edpres.html`

Como se puede ver, este método de ubicación de archivos y servicios es muy simple de manejar y se pueden tener tantas variantes como se quiera.

Básicamente el protocolo *http* funciona como servidor de documentos, de hecho una implementación de este protocolo es el programa *httpd* el cual se puede decir que es el **servidor** de documentos de hipertexto para Mosaic. Este

programa al ejecutarlo es residente en memoria y se basa en un archivo de configuración llamado *httpd.conf* , en el cual definimos las características de nuestro servidor, como pueden ser tipo de servidor, puerto de acceso, nombre del administrador, dirección de *email* del mismo para cualquier aclaración, definición del *log* de errores y nombre de nuestra máquina.

Además se basa en otro archivo de configuración llamado *access.conf* el cual controla el acceso de usuarios a documentos, con lo que da la pauta para la implementación de seguridad y restricción en el acceso de documentos, parte importante dentro del sistema.

Por último solo resta especificar el lenguaje que nos permite crear los documentos de hipertexto, esto se logra mediante **HTML (HyperText Markup Language)** . Este lenguaje es muy simple y se basa en el uso de *marcas* que ejecutan alguna acción en específico. De hecho si uno ha trabajado con los formateadores de texto de UNIX *troff*, *nroff* y con *TEX* y *L^ATEX* el manejo de **HTML** es aún más simple.

Un documento de **HTML** se basa en *marcas* que le indican un formato de información acerca de una estructura en particular. Para poder poner estas *marcas* nos basamos en los caracteres siguientes:

<	Inicio de la marca
directiva	acción a efectuar
>	Fin de la marca

Que dan como resultado la siguiente sintaxis:

<directiva>

Así, el siguiente ejemplo produciría una salida que se muestra a continuación:

```
<title>Direccion de Computo para la Administracion Academica.</title>

<p>
<h1>UNIVERSIDAD NACIONAL AUTONOMA DE MEXICO</h1>
<h1>Direccion de Computo para la Administracion Academica</h1>
```

El Departamento de Apoyo a Sistemas Unisys te da la bienvenida ! <p>

Mi nombre es Edgar Valencia Figueroa, tesista de la carrera de Ingenieria en Computacion. Las actividades que realizo dentro del Departamento son las siguientes:<p>

```
<dl>
<ul><ul>
  <li> Administracion del equipo Unix de la Dependencia.
  <li> Antencion a nuestros usuarios.
  <li> Desarrollo del proyecto de migracion de la informacion de
        historias academicas a CDROM y su consulta via RedUnam.
  <li> Desarrollo del proyecto de Kioscos de Informacion de la UNAM.
  <li> Pruebas y puesta en funcionamiento de nuevo hardware y software.
  <li> Docencia.
</ul></ul>
</dl>
```

Los topicos de mi interes son los siguientes:

```
<ol><ol>
  <li> Administracion de equipos SUN.
  <li> Nuevas tecnologias.
  <li> CDROM.
  <li> Redes.
  <li> Computadoras Personales.
</ol></ol>
```

Me gustaria discutir alguno de estos temas:<p>

```
<i><b>edgar@tztetal.dcaa.unam.mx</i></b><p>
<Address>Edgar Valencia Figueroa</address><p>

<h1> Quieres ver algo <a href="psycho.mpg"> gracioso </a>? </h1>
```

NCSA Mosaic: Document View

File Options Navigate Annotate Help

Document Title: **Dirección de Computo para la Administración Académica.**

Document URL: **http://fzetal.dcaa.unam.mx/home/edgar/fzetal_dcaa/cdprec.html**




UNIVERSIDAD NACIONAL AUTONOMA DE MEXICO

El Departamento de Apoyo a Sistemas Unisys te da la bienvenida !

Mi nombre es **Edgar Valencia Figueroa**, tesista de la carrera de Ingeniería en Computación. Las actividades que realizo dentro del Departamento son las siguientes:

- Administración del equipo Unix de la Dependencia.
- Atención a nuestros usuarios.
- Desarrollo del proyecto de migración de la información de historias académicas a CDROM y su consulta vía RedUnam.
- Desarrollo del proyecto de Kioscos de Información de la UNAM.
- Pruebas y puesta en funcionamiento de nuevo hardware y software.
- Docencia.

Los tópicos de su interés son los siguientes:

Data transfer complete.

Back Forward Home Reload Open... Save As... Clone New Window Close Window

Como se puede observar, cada una de las directivas finaliza con un / (**slash**), además de que existe un sinnúmero de directivas para especificar texto, imágenes, sonidos, animaciones, **ligas a otros documentos**, texto preformateado, efectos especiales y un sinnúmero de características que se puede agregar al texto.

Así entonces, se abre toda una gama de posibilidades para la consulta masiva de documentos, de hecho se pueden crear sistemas bastante robustos y completos si antes se hace un análisis de necesidades y se contempla este programa. Como se verá en el capítulo de perspectivas de desarrollo, se pueden implementar sistemas de información de mucha ayuda tanto para los académicos como para los alumnos de la UNAM.

2.2 Proceso de compilación y adecuación de los programas.

Una vez descrito los programas de indexamiento, recuperación de historias académicas y de presentación de las mismas así como de su consulta vía red (Wais y Gopher respectivamente), solo resta obtener los fuentes y compilarlos en nuestro servidor UNIX.

La obtención de los fuentes es muy simple: solo basta hacer un **FTP anónimo** hacia esta dirección: *boombox.micro.umn.edu*, los fuentes del programa se encuentran en el subdirectorio */pub/unix* respectivamente para WAIS y para Gopher.

Ya que se tienen estos se deben de adecuar a la configuración que tiene el servidor. Esta adecuación es extremadamente necesaria, ya que cada sistema tiene diferentes sistemas operativos; sin embargo tienen una cosa en común: todos los sistemas UNIX tienen un compilador de lenguaje "C"³ por lo que si se obtienen los fuentes, se modifican según necesidades y se compilan, se tiene un ejecutable para una plataforma específica, en nuestro caso para Solaris 2.2.

Aquí evidentemente salta la pregunta: ¿cómo es posible obtener los fuentes de un programa que es tan completo? ¿No se tiene un uso restringido del mismo? Para poder contestar a estas preguntas es necesario comprender que todo lo que se puede acceder en la red Internet es de **dominio público** es decir, cualquiera puede tener acceso a esos programas y de modificarlo según su conveniencia, sin embargo estos tienen derechos de autor que deben de ser respetados; no se tienen ninguna restricción en cuanto a su uso, ya que el objetivo de la red es la de compartir información.

Otro de los motivos por los que se ofrecen los programas fuentes es que cada sistema es totalmente diferente, es decir, la máquina original en la que se diseñó y escribió el programa tiene una configuración totalmente diferente a las máquinas en la Internet, por ejemplo si el programa fue compilado en la DEC Station que utiliza Ultrix (sistema operativo) es obvio que el ejecutable obtenido no pueda ser corrido en una SparcStation 10 corriendo Solaris 2.2. Por lo que los fuentes juegan un papel importante.

³ Al menos todos los sistemas UNIX deberían tener este compilador y es responsabilidad del administrador del equipo proveerlo a sus usuarios.

Para poder compilar los fuentes en nuestro sistema primero se debe de configurar el servidor para que pueda acceder a todas las librerías que los programas requieren. Esto se logra mediante el uso de **variables de ambiente** que el sistema operativo identifica y las aplica según sea el caso.

Una **variable de ambiente** es un espacio de la memoria de la máquina en donde se almacenan ciertas opciones que el sistema operativo busca para tener cierto comportamiento para el usuario.

Las variables de ambiente utilizadas para propósito de la compilación de los programas son las siguientes:

```
CC=gcc -traditional
```

Esta variable indica que el compilador de "C" que se quiere utilizar se llama gcc y que una de sus opciones sea `-traditional` es decir, el compilador intentará soportar algunos aspectos de los compiladores tradicionales de "C".

```
LD_LIBRARY_PATH=/usr/openwin/lib:/usr/openwin/lib/server
```

Esta variable nos da la *ruta (path)* en donde se buscarán las librerías que intervienen en el ambiente gráfico del sistema; si es que estas se necesitan.

Una vez que se tienen todas estas variables de ambiente definidas podemos proceder con la compilación. Básicamente la mayoría de los programas que se obtienen de la red (programas de dominio público), basan su compilación en un archivo llamado **Makefile**. Este archivo contiene una serie de comandos que

permiten obtener un programa ejecutable el cual se basa en el compilador del sistema. La ejecución de estos archivos se lleva a cabo a través de la utilería del sistema operativo (UNIX) denominada **make**. Para mayor referencia del uso de **make**, véase el **Apéndice II**.

El compilador de "C" que el sistema utiliza es el denominado GNU C. Este compilador no es el compilador estándar que todos los sistemas UNIX traen, sino que es parte de uno de los proyectos de **Free Software Foundation**. Esta fundación está dedicada a eliminar las restricciones de copia, distribución y modificación que se puede hacer a un software.

La palabra "free" no debe de tomarse en un sentido literal, más bien significa la "libertad" que se tiene para copiarlo y distribuirlo entre colegas y vecinos de la comunidad de cómputo, la libertad de modificarlo con lo que se tiene el control total del programa y no que el programa lo controle a uno como ocurre con el software comercial; por ello los fuentes se tienen disponibles. El proyecto GNU (Gnu's Not Unix) tiene como propósito integrar todo un software completo compatible con UNIX.

A continuación se darán algunas de las características de este compilador así como la descripción de algunas de sus opciones que sirvieron para la compilación y obtención de los programas usados.

Para comenzar diremos que este es un compilador integral. Este compilador procesa archivos de entrada de cuatro formas: los preprocesa, compila, ensambla y liga. Los nombres y sufijos que se puedan dar a los archivos fuentes determinan el tipo de lenguaje que se trata.

El ligado es la última etapa a menos que se especifiquen las opciones `-c`, `-S` o `-E` para evitarla. Para la etapa de ligado todos los archivos objeto (generalmente con extensión `.o`), librerías son pasados al programa desde la línea de comandos para su ligado.

Las opciones más comunes del compilador son las siguientes:

`-c`

Compilación de archivos fuentes, pero no el ligado. El compilador generará un archivo objeto como resultado de la salida.

`-o <archivo>`

Coloca el resultado de la compilación (archivo objeto) en el archivo especificado. Si no se especifica la salida por default será `a.out`.

`-ansi`

Con esta opción, la compilación del programa soportará todos los estándares ANSI relacionados con programas en "C".

`-traditional`

El compilador gcc tratará de soportar algunos aspectos de algunos compiladores de "C" tradicionales.

`-D[macro]`

Define una macro con la cadena especificada.

`-l [librería]`

Usa el nombre de [librería] en el proceso de ligado. El ligador busca en

lugares específicos por el nombre de la librería especificada, generalmente como `lib[libreria].a`. Si se desea especificar algún otro subdirectorio en donde buscar dichas librerías se puede especificar con `-L` y la ruta.

`-I <directorio>`

Con esta opción le indicamos en donde buscar los archivos "include" que los programas requieren.

Con estas opciones, en general, los programas fueron compilados, una vez que se tuvo toda la información recabada y la configuración necesaria, se procedió a correr la utilería `make` tanto para WAIS como para Gopher. Para mayor referencia véase los *Makefiles* de estos programas en el **Apéndice II**.

Obviamente el proceso no resultó sencillo, los primeros intentos por obtener el ejecutable fueron infructuosos, debido a errores de compilación, por lo que en algunas ocasiones se tuvo que modificar el programa fuente para que la compilación fuera exitosa. Básicamente las modificaciones que se hicieron al programa fue en el ámbito de la sintaxis, definición de variables y redefinición de las mismas, así como de una traducción de los mensajes arrojados a la pantalla, eliminación de algunas rutinas con el propósito de adecuar el programa a nuestras necesidades, inclusión de otros archivos "include", por mencionar los más importantes.

Pese a esto el proceso de compilación se llevó a cabo en una semana para ambos programas, obteniendo en ambos casos ejecutables sólidos y no corruptos. El siguiente paso era entonces las pruebas finales.

2.3 Pruebas finales.

Una vez que los programas obtenidos han sido depurados y compilados para obtener los ejecutables correspondientes (para WAIS y Gopher) se hicieron pruebas para cada uno.

Las pruebas que se establecieron para el caso de WAIS fueron muy simples y giran alrededor de la facilidad que presenta al realizar el indexamiento de los archivos y consulta de información. Debido a que en el *Makefile* correspondiente a WAIS indicamos el uso de expresiones booleanas para la búsqueda de historias también se hicieron pruebas para comprobar su funcionamiento, sin ningún problema encontrado.

En general se puede decir que para el caso del programa WAIS no hubo ninguna complicación y respondió a las expectativas que se tenían en un inicio del mismo.

Para el caso de Gopher, debido a que es la presentación que se da al usuario del sistema y con el que verdaderamente se interactúa se decidió tener una reunión con los usuarios del mismo, que en este caso es la **DGAE (Dirección General de Administración Escolar)**, quienes nos hicieron las siguientes observaciones:

1. Debido a que todos los mensajes están en inglés originalmente, se recomendó traducir todos estos, ya que sería más fácil para el usuario saber que está ocurriendo con el programa y sería mucho más fácil el reporte de fallas.

2. Impresión local y no remota de la información presentada en pantalla.
3. Niveles de seguridad para los usuarios del sistema y una clave privilegiada que pueda tener acceso a toda la información de historias académicas.
4. Ayuda en línea de los comandos más utilizados en el sistema.

En general el programa cumple, como se esperaba, con las necesidades primarias de un sistema que tuviera estas características. Sin embargo dadas las anteriores observaciones fueron necesarias varias modificaciones y reconfiguraciones al programa original para que cumpliera con estos requisitos. Cada uno de ellos presenta un caso diferente de investigación e implementación de nuevos programas, veamos cada caso por separado.

Caso 1. Traducción de mensajes.

Para este caso, la solución fue muy sencilla, simplemente se tuvo que traducir todos los mensajes de los archivos fuentes al español y recompilación del programa.

Caso 2. Impresión local.

Gracias a que Gopher tiene la posibilidad de imprimir el documento que se consulta, se decidió tener el servicio de impresión remota, es decir, las impresiones que se hicieran saldrían en las impresoras láser de alto volumen que se tienen en la dependencia, sin embargo para los usuarios resultaba un poco incomodo el tener que venir de sus lugares de trabajo hasta la DCAA para recoger las impresiones, por lo que se sugirió impresión local. Sin embargo se

presentaba un problema muy grande: como poder imprimir dentro de una sesión de UNIX un archivo a una impresora local. Generalmente estas impresoras estaban conectadas a PC's que tienen acceso a RedUNAM. Como primer paso se tuvo que investigar si esto era posible, en segundo lugar implementarlo y como último el probarlo.

Dentro del proceso de investigación nos basamos en las listas de correo que se tienen en la Internet. Como administradores del sistema estamos inscritos a varias listas de esta naturaleza en donde se discuten los siguientes tópicos:

- Administración de equipos SUN.
- LAN CDROM.
- NOVELL.
- Grupo de Administración y Seguridad de equipos UNIX.

La respuesta vino del grupo de administradores de redes NOVELL. A continuación se presenta la respuesta dada:

El siguiente *script* se debe de correr dentro de UNIX cada vez que el usuario desee imprimir el documento.

```
echo "Printing $1..."
echo "^[5i"
cat $1
echo "^[4i"
```

Lo que hace básicamente es recibir un parámetro desde la línea de comandos, que en este caso se denota como \$1 el cual es el nombre del archivo que se

desea imprimir localmente. La línea echo "Printing \$1..." tuvo que ser eliminada para evitar que salieran mensajes en la pantalla. Después lo que hace es establecer un código de control que le indica a la computadora local (una PC) que lo que recibe a continuación no será desplegado a la pantalla sino que será direccionado a la impresora, después con la instrucción cat \$1 se puede acceder el contenido del archivo, el cual sale impreso, al final establecemos el desplegado a pantalla.

Este *script* entonces es incluido como una herramienta para la impresión de historias académicas dentro de Gopher, esto se debe de configurar en el archivo `.gopherrc` que cada usuario tiene en un *home directory* y es un archivo de configuración propio de Gopher. Para la impresión de las historias Gopher genera un archivo temporal que generalmente lo ubica bajo el directorio `/tmp`, después este es pasado al programa seleccionado para la impresión como un parámetro, en este caso al *script*.

3. Niveles de Seguridad.

La seguridad que el usuario desea es aquella en donde se restringe la consulta de las historias académicas a cualquier dependencia usuaria del sistema, es decir, si por ejemplo administración escolar de la Facultad de Ingeniería desea consultar información de las historias académicas no debe de consultar la información de las historias académicas de otra facultad, sin embargo la DGAE si puede acceder a toda la información de las historias académicas puesto que ellos son la dirección encargada de la administración escolar.

Para implementar dicha seguridad nos valimos de la facilidad que ofrece el Gopher de crear nuevas sesiones con diferentes máquinas, mediante la configuración previa de las direcciones y puertos a donde debe conectarse.

La solución es muy simple, la interconexión a si misma, es decir, establecer una nueva sesión con la misma máquina, en este caso con tzetzal. Para poder lograr esto, se debe de tener la configuración de las sesiones para cada usuario del servicio (las diferentes escuelas y facultades) y una general para poder acceder a toda la información. Para tal efecto tenemos un archivo denominado .Link en donde Gopher da la oportunidad al usuario de definir el tipo de información que se visualiza en la pantalla. Esta información puede ser búsquedas en indexamientos de WAIS, sesiones FTP, etc. Dentro de estas está la opción de establecer sesiones remotas, por lo que la configuración del archivo sería la siguiente:

```
# Configuración para indexamiento Wais (tipo 7)
```

```
Type=7
```

```
Host=tzetzal.dcaa.unam.mx
```

```
Port=1300
```

```
Path=7/.waisindex/datos
```

```
# Configuración para una sesión TELNET (tipo 8)
```

```
Type=8
```

```
Host=tzetzal.dcaa.unam.mx
```

```
Port=1400
```

```
Path=Clave de acceso: facing
```

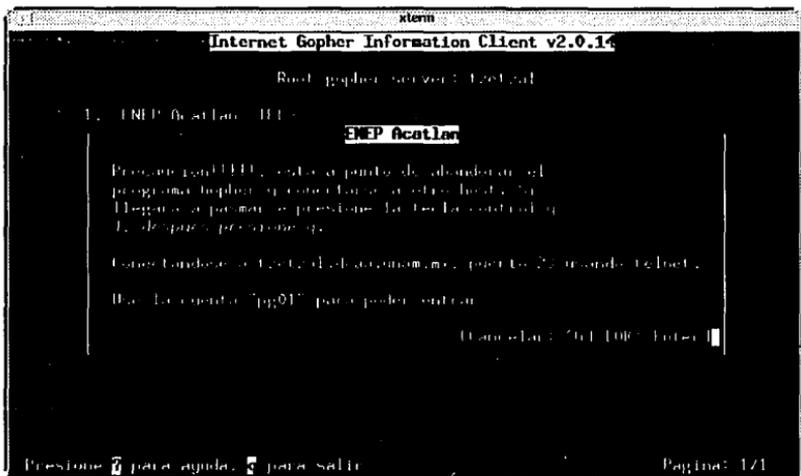
Una vez que se tiene implementada esta opción, en la pantalla en donde Gopher visualiza los datos aparece al final de la descripción del ítem <TEL> indicando que al acceder al ítem se hará una conexión a un nuevo host vía TELNET. El programa visualiza el nombre de la clave a la cual vamos a tener acceso al sistema nuevamente, he aquí donde tiene la seguridad, esta la proporciona el sistema operativo. A continuación se presenta como se visualiza dicha información en la pantalla:

```
xterm
Internet Gopher Information Client v2.0.14
Root gopher server: tzetzal

1. UNIP Oaxtlan <III>
2. Facultad de Ciencias <TEL>
3. Facultad de Ingeniería <III>

Presione ? para ayuda, q para salir          Pagina: 1/1
```

Al seleccionar esta opción el programa responde con lo siguiente:



Para la seguridad definimos claves normales de usuario, sin ningún privilegio, es decir un *shell* restringido, en donde se tiene la información específica de una escuela o facultad que quiere ser accesada. Dentro de la clave hacemos modificaciones al archivo `.profile` es decir, el archivo que el sistema operativo siempre ejecuta cuando se entra a una sesión en un *Bourne Shell*, obviamente las claves de los usuarios entran directamente a este *shell* por definición de la cuenta.

El archivo `.profile` definido es el siguiente:

Indicamos que este programa va a ser ejecutado por *Bourne Shell*:

```
#!bin/sh
```

Hacemos un "trap" de las señales de interrupción que el usuario pueda dar al momento de entrar a sesión, es decir, un "trap" lo que hace es interceptar estas señales y ejecutar un comando cuando estas se presenten, las señales van

numeradas del 0 al 15. La más común de las señales de interrupción es ^C.

```
trap "" 0 1 2 3 15
```

Guardamos en una variable "procesosh" el PID del shell de la sesión, se hace con el efecto de que si se presiona cualquier señal de interrupción el programa elimina la sesión y cierra la conexión establecida.

```
procesosh=`ps | grep " sh" | cut -c1,2,3,4,5,6`
```

También en otra variable guardamos el nombre del usuario.

```
usuario=`who am i|cut -d" " -f1`
```

Corremos el *daemon* de gopher, el cual es el encargado de atender cualquier petición de un cliente para acceder la información, en este caso se debe de acceder por el puerto 1400.

```
/home/gopher/gopher2.014/gopherd/gopherd -cu pg01 -o \  
/home/gopher/gopher2.014/gopherd/gopherd.conf \  
/bases/pg01/datos 1400
```

Guardamos el PID del *daemon* para posteriormente cuando acabe la sesión o se aborte este sea eliminado de la memoria.

```
proceso=`ps -u $usuario|grep gopherd|cut -c1,2,3,4,5,6`
```

Si por algún motivo el programa termina mal, es decir, es abortado, interrumpido o no termina normalmente, terminamos la sesión. Esto se logra preguntando cual fue el estado de terminación del programa mediante \$? , se hicieron pruebas y cuando el programa terminaba mal arrojaba un código "1" al sistema operativo, se compara y si es igual, termina el *daemon* de la memoria y se termina la sesión.

```
if [ $? -eq "1" ]; then
  echo $procesosh
  kill -9 $proceso
  kill -9 $procesosh
fi
```

Si todo va bien, hasta este punto se manda correr el cliente de Gopher por el puerto 1400 en este caso, cada usuario tendrá un *daemon* que atiende a los clientes por diferentes puertos.

```
/home/gopher/gopher2.014/gopher/gopher tzetzal 1400
```

Nuevamente si el cliente termina mal o es abortado, se elimina el *daemon* y se termina la sesión.

```
if [ $? -eq "1" ]; then
  echo $procesosh
  kill -9 $proceso
  kill -9 $procesosh
fi
```

Si todo termina normalmente, solo resta quitar el *daemon* de la memoria y salir de la sesión normalmente.

```
kill -9 $proceso
exit
```

Dicho proceso se muestra a continuación, primero el usuario se identifica:

```
xterm
Trying 132.248.27.10 ...
Connected to tzetzal.dcea.unam.mx.
Escape character is '^]'.

UNIX(r) System V Release 4.0 (Tzetzal)

login: pg01
Password: █
```

una vez identificado por el sistema como un usuario del mismo, entra a un nuevo Gopher para la consulta de las historias correspondientes a su nivel de seguridad, en este caso solo de consulta de una escuela:

```
xterm
Internet Gopher Information Client v2.0.14
Root gopher server: tzetzal
1. Historias Academicas ENEP Acatlan 94-I (GDROM) <9>
2. Historias Academicas ENEP Acatlan 94-II <?>

Presione █ para ayuda. █ para salir
Pagina: 1/1
```


xterm

Internet Gopher Information Client v2.0.14

Host: 100. Académicas, E.N.E.P. Acatlan, 94-11-378

1.	NOMBRE:	GONZALEZ ROYAS MARTO	No. CUENTA:	8701143-4
2.	NOMBRE:	AVILA SOLIS CARLOS	No. CUENTA:	8704679-5
3.	NOMBRE:	GONZALEZ RODRIGUEZ LILIAN	No. CUENTA:	8701340-5
4.	NOMBRE:	TRUJANO GARCIA OSCAR	No. CUENTA:	8701435-2
5.	NOMBRE:	GONZALEZ ARRIOLA DEGA	No. CUENTA:	8702467-3
6.	NOMBRE:	SIQUEIRA ROMERO GABRIEL GONZ	No. CUENTA:	8702943-3
7.	NOMBRE:	SALAZAR AVILA FELIX CONRUPCION	No. CUENTA:	8703034-3
8.	NOMBRE:	SALAZAR ROMERO ABRAHAM	No. CUENTA:	8712099-3
9.	NOMBRE:	ROMERO DURAN OSCAR	No. CUENTA:	8702961-1
10.	NOMBRE:	MENDOZA LOPEZ GABRIEL	No. CUENTA:	8709011-0
11.	NOMBRE:	RODRIGUEZ MUNOZ MYRTAM	No. CUENTA:	8709340-7
12.	NOMBRE:	OSIELANO BENTIN MARTA ANTONIA	No. CUENTA:	8709742-9
13.	NOMBRE:	ANGELIANO NIEVE II PATRICIA	No. CUENTA:	8703260-3
14.	NOMBRE:	CASIMIRO GARCIA RAUL	No. CUENTA:	8708413-3
15.	NOMBRE:	BALAZAR BERNAL MIGUEL ANGEL	No. CUENTA:	8706105-7
16.	NOMBRE:	GARCIA IBAÑERRA JUAN GUADALUPE	No. CUENTA:	8728276-2
17.	NOMBRE:	CRUZ LOPEZ MATEOS	No. CUENTA:	8753335-9
18.	NOMBRE:	VALDEZ MEDINA SERGIO	No. CUENTA:	8703087-3

Presione **?** para ayuda, **S** para salir, **M** para ir al menú superior. Página: 1/6

La información consultada es la siguiente:

NOBRE: ROMERO PEREZ MAURICIO
(1k) 74%

No. CUENTA:9452672-3

NOBRE: ROMERO PEREZ MAURICIO
PLANTEL: E.N.E.P. ACATLAN ECONOMIA
CARRERA: LIC EN ECONOMIA
GENERACION: 94
SEMESTRE: 94-1

No. CUENTA:9452672-3
CLAVE:208
CLAVE:21

REPORTE DE AVANCE EN CREDITOS				REPORTE DE ASIGNATURAS	
	Cubiertos	Totales	%Avance	APROBADAS	: 6
Obligatorios	36	300	12.00%	NO APROBADAS:	0
Optativos	0	0	0.00%	TOTAL	: 6
Totales	36	300	12.00%	PROMEDIO	: 9.33

PRIMER SEMESTRE
208 0241 06 GEOGRAFIA ECONOMICA B OR 94-1 1765618 1102 1
208 0291 06 HISTORIA ECONOMICA GENERAL I MB OR 94-1 1765626 1102 1

[Ayuda: ?] [Salir: u] [Pagina Abajo: Espacio]

Al final de la sesión, Gopher regresa al menú original y se puede hacer alguna otra consulta.

4. Ayuda en línea.

Para obtener ayuda en Gopher, basta presionar la tecla "?" en ese momento se desplegara un archivo con los comandos más usados dentro de Gopher. Este archivo de ayuda se puede configurar dentro de conf.h o simplemente añadiendo la línea al programa gopher.c. A continuación se presenta el archivo de ayuda:

Gopher Help File
(2k)

308

Ayuda rapida de Gopher

Utilizacion de Gopher

Use las flechas o sus equivalentes en vi/emacs para moverse.

Derecha, Return: "Entra"/Despliega elarticulo actual.
Izquierda, u: "Salir" del tema articulo/Va a un nivel superior.
Abajo: Mueve
Arriba: Mueve a la línea anterior.
>, +, Pgdwn, Espacio : Visualiza la siguiente pagina.
<, -, Pgup, b: Visualiza la pagina anterior.
0-9: Va a una línea especifica.
m: Regresa al menú principal.

Bookmarks

a : Agrega el artículo actual a la lista de bookmark.
A : Agrega el directorio actual/busca en la lista de bookmark.
v : Visualiza la lista de bookmark.
d : Borra un bookmark/entra a un directorio.

Otros comandos

q : Sale al prompt.
Q : Salida incondicional.
s : Salva el artículo actual en un archivo.
S : Salva la lista actual del menú en un archivo.
D : Transfiere un archivo.
r : va al menú raíz del artículo actual.
R : va al menú raíz del menú actual.
= : despliega información técnica acerca del artículo actual.
^ : despliega información técnica acerca del directorio actual.
o : Conecta hacia un nuevo servidor de gopher.
O : Cambia opciones.
/ : Busca por un tema en el menú.
n : Halla la siguiente búsqueda del tema.
g : "Sugerencias" via email al administrador del tema actual.
!, \$: Shell Escape (Unix) or Spawn subprocess (VMS).

Objetos Gopher:

Item tag	Tipo	Descripcion
ninguno)	0	archivo / 1 directorio

<)	s	archivo de sonido
<Picture>	I,g	archivo de imagen
<Movie>	;	archivo de película
<HQX>	4	archivo BinHexed Macintosh
<Bin>	9	archivo binario
<PC Bin>	5	archivo binario DOS
<CSO>	2	CSO (ph/qi) servidor phone-book
<TEL>	8	conexion telnet
<3270>	T	conexion telnet (emulacion IBM 3270)
<MIME>	M	Multi-purpose Internet Mail Extensions file
<HTML>	h	archivo HyperText Markup Language
<?>	7	tema indice-busqueda
<??>	(none)	forma ASK

[Ayuda: ?] [Salir: u] [Pagina Abajo: Espacio]

Al probar la totalidad del sistema, se vió que era consistente y seguro y que los usuarios quedaron satisfechos con este, sin embargo se pueden hacer muchas otras modificaciones según necesidades y puede tener muchas perspectivas de desarrollo como se verá en el capítulo de **Perspectivas de desarrollo**.

3.

Grabación del CD - ROM.

Como ya se había comentado en el capítulo introductorio, en el tema de la teoría básica del CD-ROM, uno de los más novedosos avances en el campo del almacenamiento en discos compactos, es la posibilidad de que el usuario realice la grabación de su información sin necesidad del laborioso proceso que para grabar un CD-ROM convencional se requiere. Esta nueva tecnología es llamada **Compact Disc Recordable** o **CD-R** y los discos que son grabados a través de este procedimiento son exactamente iguales a un CD-ROM convencional.

Una de sus características es que está diseñado para la producción a bajo costo de pequeñas cantidades de discos, lo cual, se ajusta perfectamente al tipo de aplicación que estamos desarrollando ya que la generación de historias se realiza cada semestre y sólo se requiere una cantidad muy pequeña de copias. Además de ello se obtiene la ventajas que ofrece el almacenamiento digital de información, que en este caso es el poder acceder la información a través de la red.

Debido a lo novedoso de esta tecnología en el país consideramos importante describir la forma en que se realiza el proceso de grabación de un CD grabable, tanto en el aspecto físico como en el lógico. Estos temas se explicarán de manera más explícita en los siguientes temas.

3.1 Características del hardware.

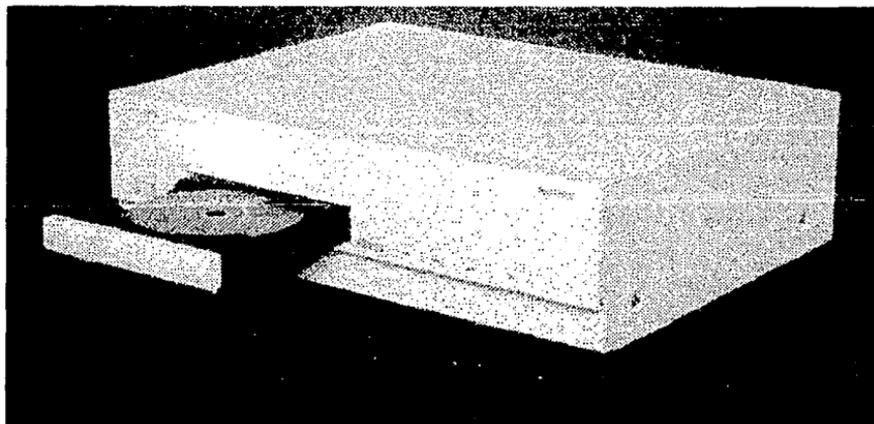
Los componentes de una cabeza de lectura/escritura de una unidad de CD-R son en esencia los mismos que una cabeza de lectura de una unidad de CD-ROM convencional. La diferencia radica en que la cabeza del CD-R tiene un rayo láser de mayor intensidad para quemar el disco y obtener las depresiones o **pits** que constituyen los datos.

La intensidad del rayo debe variar de acuerdo a la velocidad de rotación del disco. Para realizar una escritura más rápida es necesario que el disco gire más rápido, lo que implica que el rayo láser tenga pulsos más cortos. Además de la variación debido a la velocidad de rotación, también debe variar cuando la unidad se utiliza para lectura de discos, ya que su intensidad debe ser la normalmente utilizada en las unidades de CD-ROM.

Las ranuras formadas previamente en un CD grabable, las cuales tienen una profundidad de 100 nm. y 600 nm. de ancho, son usadas como guía de alineación. En vista de que en su estado virgen la capa teñida en oro es reflejante (como los **Lands** del CD-ROM), el rayo láser es utilizado para alterar la tintura de tal manera que disipa la luz en lugar de reflejarla. Esto crea áreas muy semejantes a los pits que se queman en el CD-ROM. Esta manera de realizar la grabación crea un disco que, ópticamente hablando, tiene los mismos **lands** y **pits** que tiene un CD-ROM estándar.

En las revistas especializadas de CD-ROM se manejan cuatro grabadores de CD: JVC Personal RomMaker de JVC Information Products, CDD 521 de Phillips,

CDW-900E de Sony y el grabador RCD-202 de Pinnacle. Son varios los artículos que se han publicado sobre análisis comparativos entre los cuatro equipos, principalmente de software. El equipo de grabación adquirido es el modelo de Phillips que se muestra a continuación.



En el siguiente cuadro comparativo se muestran algunas características importantes de los equipos como velocidad de grabación, formatos que puede manejar, interfaces a otros equipos, etc.

EQUIPO \ CARACTERÍSTICA	JVC	PINNACLE MICRO RCD-202	PHILIPS CDD 521	SONY CDW-900E
Plataformas	DOS, UNIX, MACINTOSH	MACINTOSH, WINDOWS	DOS, UNIX, MACINTOSH, WINDOWS	DOS, MACINTOSH
Software	Personal Archiver 1.05	Recordable CD 1.2	CD-Gen 2.12	PC ISO Formatter 1.10

Capítulo III Diseño de la propuesta.

Formatos de grabación	CD-DA CD-I CD-ROM XA ISO 9660 High Sierra HFS Mixed Mode UFS	CD-DA CD-I CD-ROM XA ISO 9660 HFS Photo-CD	CD-DA CD-I CD-ROM XA Multisession ISO 9660 Photo CD HFS Mixed Mode UFS	CD-DA CD-I CD-ROM XA HFS Mixed Mode
Tipo de medio	CDR-63, CDR-74 (580/680)	CDR-63, CDR-74 (580/680)	CDR-63, CDR-74 (580/680)	CDR-63, CDR-74 CDR-18
Carga del disco	Caddy	Caddy	Tray	Tray
Lectura de CD-R	SI	SI	SI	NO
Escritura de multisesión	SI	SI	SI	NO
Lectura de multisesión	SI	SI	SI	NO
Selección de sesión	SI	SI	SI	NO
Velocidad de escritura	Simple	Simple	Doble	Simple, Doble
Tiempo para escribir 600 MB en velocidad simple	60 min.	60 min.	Depende del software	74 min.
Tamaño del buffer	64 k	64k	256k escritura/64 k lectura	4 Mb
Tarjeta de interfase para PC	1510	Adaptec 1542	Adaptec 1542	Adaptec 1542
Conectores	SCSI-1 Centronics	SCSI-1 Centronics	50-pin SCSI-2	1) 36-pin half pitch ribbon-type. 2) 50-pin SCSI-2 3) optical fiber for audio
MTBF	20,000 POH	20,000 POH	25,000 POH	5,000 POH

La velocidad de escritura se refiere a la tasa de transferencia de datos entre el disco y el grabador. Estos equipos requieren que está tasa se mantenga relativamente constante si no se corre el riesgo que la grabación se realice en forma incorrecta. Estos equipos manejan dos velocidades: la simple es de 300 KB por segundo y de 600 KB para la velocidad doble. Para lograr un resultado adecuado, la máquina debe ser capaz de mantener esta tasa de transferencia. Es

posible que ya sea que el CPU sea demasiado lento, la velocidad del bus deficiente o en el caso de un servidor que la demanda de recursos por parte de otros procesos reste velocidad de transferencia a los controladores de disco.

Es por estos motivos que si se quiere lograr un resultado adecuado se debe realizar en una máquina de buen desempeño en el caso de PC y utilizar un disco con una interfaz tipo SCSI. En el caso del equipo que estamos empleado para realizar la grabación, Young Minds integra una caja negra que es una PC 386 con disco SCSI de 0.8 GB que libera al sistema operativo de mantener una tasa de transferencia uniforme, siendo el mismo software quien la controla a través de esta PC.

3.2 Características del software.



El software utilizado para la grabación del CD-ROM es **MakeDisc** de la compañía **Young Minds**. Este software realiza el proceso de **premaster** que formatea los archivos de datos para generar una imagen en disco de la información antes de ser grabada en CD-ROM.

Durante el proceso de **premaster** nuestros archivos pueden ser convertidos a una forma compatible con el estándar **ISO 9660**. Al realizar el **premaster** los datos con este estándar nos dará como resultado un CD-ROM compatible y que se puede usar con un gran número de sistemas de cómputo.

Los datos en un CD-ROM deben estar organizados en un formato que pueda ser leído por cualquier sistema de cómputo que cuente con una unidad lectora de

CD-ROM. MakeDisc no solo realiza el proceso de *premaster* con nuestros datos si no que al mismo tiempo hace posible lo anterior, es decir, puede hacer que los datos sean compatibles con el estándar ISO 9660.

El formato de datos mas comúnmente usado en los CD-ROM es el que se desprende del ISO 9660, este estándar es muy restrictivo y diseñado para el ambiente DOS. MakeDisc da a los usuarios de UNIX tres opciones que les permite tener todas las ventajas de las características del sistema de archivos UNIX y al mismo tiempo crear discos totalmente compatibles con el ISO 9660. Estas opciones incluyen *ISO 9660*, *ISO 9660/Rock Ridge*, o *ISO 9660 con tablas de traducción*.

Como ya mencionamos el formato estándar para los datos de un CD-ROM es el ISO 9660. Desde los sistemas DOS tenemos mas restricciones en cuanto a las convenciones de nombres uso de subdirectorios, por lo que hay tres niveles en el estándar ISO 9660:

- **ISO 9660, Nivel 1**

Este nivel es el mas ampliamente usado. Este es compatible con las convenciones de nombres y uso de subdirectorios DOS. Esto es, los nombres de archivos son de ocho caracteres en mayúsculas o minúsculas, con tres caracteres de extensión. En cuanto a los subdirectorios no sobrepasan los ocho niveles.

- **ISO 9660, Niveles 2/3**

Este es el nivel menos frecuentemente usado. Estos niveles restringen a 30 caracteres los nombres de archivos y directorios. Note que varias unidades de

CD - ROM soportan las convenciones de nombres de los niveles 2/3 y algunos sistemas operativos no pueden interpretar nombres largos de archivos y directorios. Es importante antes de escoger estos niveles estar seguros que el equipo en donde se leerán estos datos soporte estas convenciones.

Los usuarios de UNIX encontraron que sus conjunto de datos no podían ser bien traducidos al ISO 9660. Alrededor de 1991 la industria creó el **Rock Ridge Interchange** y el **System Use Sharing Protocols**.

Las extensiones Rock Ridge proveen lo mejor de ambos mundos: nombres de archivo y directorios estilo UNIX totalmente compatibles con el ISO 9660. El Protocolo Rock Ridge da a los usuarios UNIX la flexibilidad que necesitan, manteniendo la portabilidad de sus CD-ROM's.

Los componentes del proceso **premaster** los podemos catalogar en tres: los **insumos**, el **proceso** y los **resultados**. Cada uno de estos componentes estará a su vez compuesto de varios más.

Dentro de los insumos tenemos lo que son:

- **Características del volumen.-** Como lo son tipo de volumen a ser creado (ISO 9660, HFS, etc.), sus características de identificación (identificador, editor, etc.), y el tamaño del directorio raíz y la tabla de trayectorias, lo cual determina el número de directorios y archivos que contendrá nuestro disco. Muchas de estas opciones son estáticas y el programa las da por default.
- **Lista de archivos.-** Es una lista de los archivos que serán grabados en el CD-ROM.

- **Archivos de datos.**- Es decir, donde se encuentran físicamente estos archivos a ser incluidos en el disco. Estos pueden estar localizados en cualquier disco duro local, CD-ROM, o disco remoto (en red). En teoría cualquier dispositivo que el sistema de computo pueda ver como un dispositivo almacenador de información de acceso aleatorio susceptible de ser accedido.

Dentro del proceso tenemos los siguientes pasos:

- **Instrucciones especiales.**- Las cuales contienen comandos de procesamiento tales como conversiones de nombres no estándares al estándar ISO 9660, crear archivos o directorios ocultos, o eliminar números de versión de los nombres de los archivos.
- **Estado del programa y mensajes.**- Es posible capturar estos mensajes en la bitácora (log file) para su posterior análisis, las estadísticas indican el volumen del disco ocupado, el número de directorios y archivos en cada uno, así como de los que se encuentran todavía disponibles.
- **Premaster, simulación y validación.**- Estos pasos son en si lo fuerte del programa. Como ya se menciona el premaster es una imagen, una "copia", del CD-ROM real, la cual se comporta de manera idéntica al CD-ROM cuando sea ya grabado; esta imagen comúnmente es hecha en disco duro (a veces en cinta), para que una vez realizada solo se vaya copiando tal cual al momento de ir grabando el CD-ROM. La simulación permite analizar el comportamiento que tiene la imagen (el premaster) ante situaciones críticas y de carga de trabajo, y así predecir el comportamiento del CD-ROM una vez grabado, es decir, se hacen pruebas del resultado final sin tener que echar a perder un CD virgen, y poder corregirlas a tiempo; este es un paso sumamente importante para obtener un éxito rotundo en la producción del CD-ROM. Por último la

validación es que, una vez ya analizado y probado el comportamiento del premaster se de el visto bueno, es decir, se compruebe que no haya errores que afecten el comportamiento del disco, además se hace una comparación entre los archivos fuente y la imagen del CD - ROM byte a byte para detectar errores; una vez hecho esto se puede pasar a la grabación física del CD-ROM.

Entre los resultados que se pueden obtener tenemos a:

- **Imagen física.-** Esta es la salida estándar, la cual puede escribirse a disco, cinta, o al grabador de CD's. Este termino se refiere a un archivo muy grande que contiene (byte por byte) los datos que contendrá el CD final.
- **Imagen Virtual.-** Esta imagen, en cambio, contiene apuntadores a los archivos de datos y no crea una copia byte a byte de los datos. Esta imagen virtual puede ser usada para simular al CD, pero modificaciones a la organización del disco pueden ser hechas sin tener la actual re-copia de los archivos de datos.
- **Imagen del archivo.-** Esta imagen puede ser grabada solo en cinta o disco duro, y requiere de otro proceso para llevarla al CD.
- **CD-ROM.-** Todos los anteriores resultados se pueden considerar como temporales y no finales, en los que se pueden hacer modificaciones sin muchas complicaciones, y podemos estar "jugando" con ellas sin que ello afecte al producto final, claro si se corrigen antes de crearlo. Pero una vez creado el CD-ROM, no hay vuelta atrás; todos los errores serán ya incorregibles y en caso de que existan será necesario realizar otro premaster o corregir el existente y después grabar otro CD-ROM pero nunca el mismo. Entonces podemos afirmar que este si es el producto final de todo este largo proceso.

3.3 Criterios de grabación de la información.

Una vez que ya teníamos tanto el hardware como el software que íbamos a utilizar para poder grabar el CD-ROM teníamos entonces que decidir que información es la que íbamos a grabar en él.

La información académica de un semestre de todos los alumnos de la UNAM, tanto de licenciatura como de bachillerato, ocupa alrededor de 600 Mb, es decir, prácticamente todo un CD-ROM. Pero hay que recordar que al momento de indexar la información se crean archivos de índices necesarios para la búsqueda de la información que se vaya a solicitar; entonces si aumentamos los índices generados por el indexamiento de esa información ocuparíamos poco mas de lo que le cabe al disco, por lo que la información total tendría que ser grabada en dos CD-ROM's, con el notorio desperdicio de uno de ellos que no alcanzaría a ser totalmente grabado.

Con lo anterior decidimos que había que grabar una parte en el disco duro del SparcStation, pero el problema era qué información grabar en el disco. Analizando esta información, los índices y las historias, nos dimos cuenta que lo fundamental es la información de las historias, la cual en dado caso que se llegara a perder por cualquier cosa o borrar del disco sería mucho mas difícil recuperarla ya que se tendría que llevar a cabo nuevamente todo el proceso desde el bajar la información respaldada en cintas del equipo Unisys, ya que todos los resultados intermedios del proceso son borrados en los dos equipos por las políticas de espacio en disco que se tienen y que no tiene caso guardarlos y que ocupen espacio útil para otras cosas.

Mientras que los índices, en caso de pérdida, se recuperarían con solo volver a correr el proceso de indexamiento de la información, la cual se encontraría en su totalidad respaldada en CD-ROM (dos o mas copias de cada semestre). Además al tener los índices en disco duro reducimos un poco el tiempo de acceso a la información dado que el acceso al índice sería mucho mas rápido y solo se accedería al CD-ROM para leer la información académica; mientras que de otra manera primero se accedería para leer el índice y luego nuevamente para leer la información. Otra razón es que por las políticas de uso del espacio en disco llegaría un momento en que por el almacenamiento de información (que ocupa mas espacio que los índices) se rebasaría la cantidad de espacio disponible en disco duro, y con los índices se podría estar dando un mantenimiento para no tener todos guardados sino a lo mucho dos o tres, con la facilidad de recuperarlos en cualquier momento.

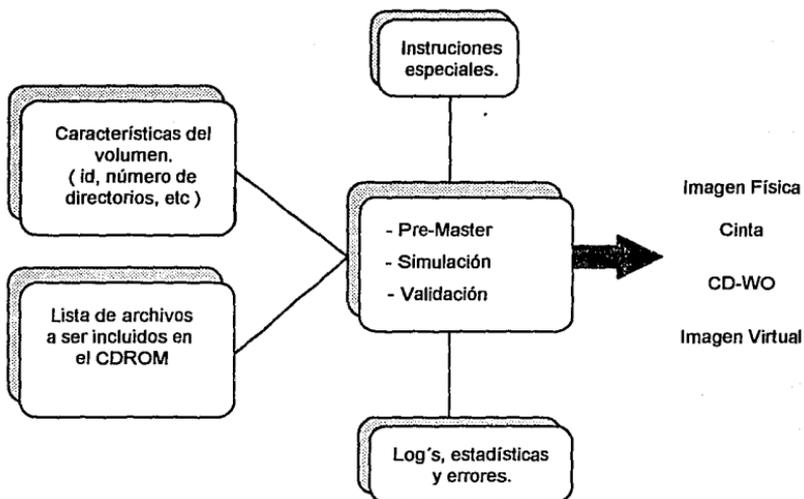
Es así que en conclusión se decidió grabar la información de las historias en el CD-ROM, mientras que los índices se almacenaran en disco duro.

3.4 Pre-master, pruebas y grabación.

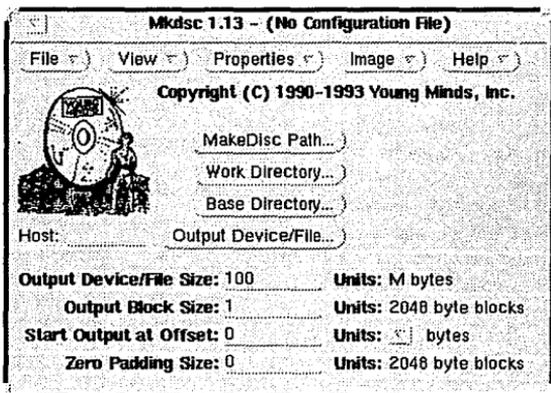
Una vez definidos los criterios de grabación de la información en el CD-ROM, solo resta realizar el proceso de premaster de la información y su posterior grabación al CD.

Debemos comprender que la producción de un CD-ROM no puede ser descrita como un solo paso que involucra un programa y ciertas acciones, sino que es

toda una serie de etapas a seguir, pruebas de la información, aceptación de estas y grabación de la misma. Dicho proceso se muestra a continuación:



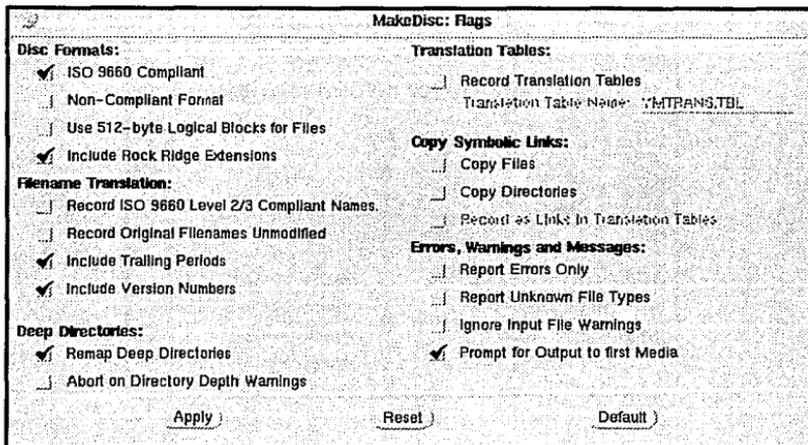
Debemos también de tener presente el software involucrado para dicho proceso, en este caso tenemos disponible el software de Young Minds descrito anteriormente. En cada etapa del proceso nos apoyaremos con las ventanas que ofrece el software al usuario y de las opciones que este tiene. La interfaz que da al usuario es la siguiente:



El proceso de premaster de la información consiste en hacer una *imagen* de la información que contendrá el CD-ROM en algún otro dispositivo previo, como puede ser un disco duro alterno, para su prueba; dicha imagen se comportará exactamente igual que si estuviera grabada en el CD-ROM. Una vez que las pruebas de la información han sido satisfechas, se procede a grabar el disco, dicho proceso se conoce como *quemado del CD-ROM o corte de la información*.

- **Etapas 1. Definición de las características del volumen a crear.**

En esta etapa se definen todas las características del disco que se va a crear, como pueden ser: ISO 9660, RRIP, etc., sus características que lo identifican (ID [identificador], nombre del productor, etc.) definición de la ruta de la información a ser grabada, entre otra información. El software de YMI nos permite definir estas características muy fácilmente, como se muestra a continuación:



Como se puede observar, nos permite definir el disco ISO 9660, incluir extensiones Rock Ridge, grabar tabla de traducción, etc. Para nuestro caso, las opciones que se deben de poner son las siguientes:

1. ISO 9660, es decir, que nuestros CD-ROM cumplan con este estándar, ya que en un futuro estos CD-ROM's van ha ser accedados localmente por un equipo PC, por lo que debe de leer la información sin ningún problema.
2. Grabación de la tabla de traducción, es decir, los nombres de los archivos que se tienen almacenados en el CD-ROM (o imagen previa) difieren mucho de los nombres de archivos que una PC puede manejar (sólo 8 caracteres y una extensión de 3 contra 14 caracteres en el sistema operativo UNIX), por lo que esta tabla ayuda a traducir los nombres de los archivos para guardar la compatibilidad de los sistemas operativos que lo accesan.
3. Extensiones Rock Ridge. Cuando se definieron los estándares 9660 para la grabación de información en un CD-ROM, los usuarios del sistema operativo

UNIX se percataron de que los nombres de sus archivos en sus sistemas no se acoplaban del todo a este estándar, por lo que en 1991 se implementó el **Rock Ridge Interchange and System Use Sharing Protocols (RRIP)**.

Esto da la facilidad de tener nombres al estilo UNIX con una compatibilidad total con ISO 9660. Así se mantiene la portabilidad de la información entre sistemas UNIX.

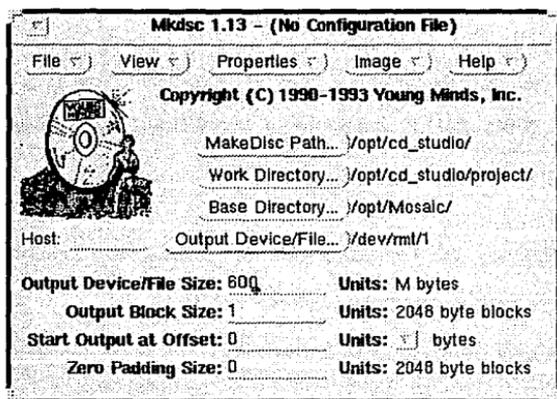
Además definimos las características del disco como nombre del mismo, identificador, etc. como se muestra a continuación:

MakeDisc: Volume Information	
Target System Type:	ISO 9660 RECEIVING SYSTEMS
Formal Disc Name:	VOLUME_NAME
Formal Disc Set Name:	VOLUME_SET_NAME
Publisher Name:	THE PUBLISHER NAME
Formatting by:	YOUR COMPANY
Applications:	APPLICATION_NAMES
Copyright Notice File:	COPYRGHT.FIL;1
Abstract File:	ABSTRACT.FIL;1
Bibliography File:	BIBLIOG.FIL;1
Creation Date/Time...)0000/00/00 00:00:00.00 +00
Modification Date/Time...)0000/00/00 00:00:00.00 +00
Expiration Date/Time...)0000/00/00 00:00:00.00 +00
Effective Date/Time...)0000/00/00 00:00:00.00 +00
<input type="button" value="Apply"/> <input type="button" value="Reset"/>	

- **Eta**pa 2. Definición de la información a grabar.

Una vez definidas las características físicas del disco, procedemos a escoger la información que se desea grabar, este proceso es muy sencillo, ya que el

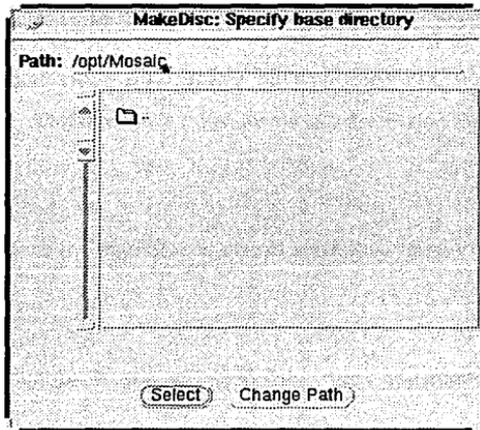
software de YMI permite la definición de una ruta en donde la información reside y de donde este la tomará para procesarla.



• **Etapa 3. Definición de área de trabajo y ruta del software de grabación.**

En esta etapa definimos una área en donde el software pueda dejar archivos temporales durante el premaster y grabación del CD-ROM. Además definimos la ruta en donde el software se encuentra.

El software de YMI nos permite establecer de forma clara y sencilla cada uno de los directorios que toman parte en dicha etapa, para ejemplificar esto, a continuación se muestra una de las ventanas en donde nos permite seleccionar el *base directory* de la información a grabar en CD-ROM.



- **Etapa 4. Definición de instrucciones especiales para el premaster y grabación.**

En este caso, el software hace el proceso automáticamente, por lo que el usuario no se debe preocupar por añadir instrucciones especiales al mismo.

- **Etapa 5. Status y mensajes de error.**

En este punto, el software es muy explícito en las acciones que va efectuando, ya que, paso a paso este nos indica que es lo que esta haciendo y nos reporta cualquier error que encuentre durante el proceso.

Solo resta definir el dispositivo que contendrá la imagen previa de la información (pre-master). En este caso se cuenta con un disco duro externo de 800 Mb, suficiente para guardar la información de un CD-ROM. En nuestro caso nos encontramos trabajando bajo UNIX, por lo que definimos a este dispositivo como el ***/dev/rmt/1***.

Definidos todos los componentes solo resta efectuar el *pre-master*, a continuación se da una secuencia de pasos para poder hacerlo, tomando en cuenta las ventanas anteriores y el software de **Young Minds**.

Paso 1.

Cargamos el programa **MakeDisc** desde el ambiente gráfico del Solaris.

Paso 2.

Definimos el **MakeDisc Path**, es decir, la ruta en donde se encuentra el software de YMI.

Paso 3.

Definimos el **Work Directory** es decir, la ruta que utilizará el programa como área de trabajo para la creación de archivos temporales. La información ahí almacenada no toma parte en el pre-master.

Paso 4.

Definimos el **Base Directory** es decir, el subdirectorio que contiene toda la información (archivos, programas) que deseamos grabar en el CD-ROM. Esta información toma parte del pre-master.

Paso 5.

Seleccionamos el **Output Device/File** es decir, el archivo/dispositivo que almacenará nuestro pre-master. Como mencionamos anteriormente este es **/dev/rmt/1**.

Paso 6.

Definimos el tamaño del dispositivo de salida en Mb. El tamaño de default es 100 Mb. Debemos recordar que si queremos grabar la totalidad de un CD-ROM, debemos especificar 540 Mb para discos de 63 minutos o 640 Mb para discos de 74 minutos.

Paso 7.

Definición de banderas (*flags*), es decir las propiedades del disco (ISO 9660, Rock Ridge extensions, etc.) dentro del menú de *Properties*. En este caso las opciones son:

- ISO 9660 compliant.
- Include Rock Ridge Extensions.
- Record Translation Tables.

Las demás opciones en default.

Paso 8.

Definición dentro del menú de *Options* la información del volumen, es decir, nombre del CD-ROM, la información que se graba en el, fecha de creación, hora de creación, etc.

Paso 9.

Definidas todas las características podemos proceder a realizar el premaster, dentro del menú de *Image* seleccionamos *Premaster*, entonces desplegará las siguientes pantallas:

```

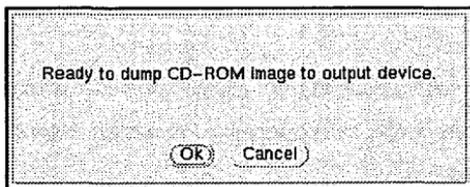
MakeDisc: Execution
    appearing in the specified file set will still be translated.
Now /opt/Mosaic/
Encountered 1 non-ISO 9660 compliant names.

      files      directories      totals
length
  identifier          1          0          1
  any len error       1          0          1
characters
  lower case          1          0          1
  special chars       1          0          1
  any char error      1          0          1
any error             1          0          1

cdlevel, ver. 1.11: Copyright 1991, 1992 By Young Minds, Incorporated
Tree not too deep. Tree unmodified.
cdback, ver. 1.11: Copyright 1990, 1991, 1992 By Young Minds, Incorporated
Total CD-ROM image size : 1697 blocks = 3475456 bytes
.
cdout, ver. 1.11: Copyright 1990, 1991, 1992 by Young Minds, Incorporated
    
```

En esta pantalla nos indica la ejecución paso por paso de los programas que componen al software. En primer lugar se ejecuta el **cdfront**, es decir un programa que efectúa un preprocesamiento para encontrar los archivos señalados para el premaster y verificar que efectivamente se pueden acceder. Después ejecuta el **cdlevel** que reconstruye la estructura de directorios si se encuentra con alguno que no cumpla con el ISO 9660. Al final ejecuta el **cdback** el cual genera la información necesaria para completar la imagen del CD-ROM.

Al final de la ejecución de estos programas visualiza la siguiente ventana:



Al dar OK se lleva a cabo el Premaster de la información. Se ejecuta el programa **cdout** el cual es el encargado de llevar los bloques de información al dispositivo de salida asegurando que cumplan con el estándar ISO 9660, además verifica el tamaño de la información por si esta fuese demasiada para el dispositivo que la almacena:

```

MakeDisc: Error Log
characters
  lower case      1          0          1
  special chars   1          0          1
  any char error  1          0          1
  any error       1          0          1

cdfront done at Mon Jul 11 09:14:10 1994
cdlevel start at Mon Jul 11 09:14:10 1994
cdlevel -t /opt/cd_studio/project/
cdlevel, ver. 1.11: Copyright 1991, 1992 By Young Minds, Incorporated
Tree not too deep. Tree unmodified.
cdback start at Mon Jul 11 09:14:10 1994
cdback /opt/cd_studio/project/
cdback, ver. 1.11: Copyright 1990, 1991, 1992 By Young Minds, Incorporated
Total CD-ROM image size : 1697 blocks = 3475456 bytes

cdback start at Mon Jul 11 09:14:10 1994
cdout start at Mon Jul 11 09:14:10 1994
cdout -w -b1 /opt/cd_studio/project/ /dev/rmt/1 600
cdout, ver. 1.11: Copyright 1990, 1991, 1992 by Young Minds, Incorporated
Press <return> when ready to begin CD-ROM image dump
on outdevice '/dev/rmt/1'

```

Presionamos <ENTER> y la imagen esta siendo grabada.

Acabado el pre-master toda la información de encuentra almacenada en `/dev/rmt/1`, por lo que el siguiente paso es la prueba de esta, para tal efecto debemos de copiar la información hacia una partición del disco duro de nuestra estación de trabajo, esto es con el propósito de no hacer la prueba directa en `/dev/rmt/1`.

Para llevar a cabo dicha acción copiamos con el comando de UNIX dd (device to device) la información, este comando lo que hace es copiar byte por byte la información:

```
tzetzal:1> dd if=/dev/rmt/1 of=/dev/rdisk/clt3d0s7
```

`if` especifica el dispositivo de entrada que en este caso es donde se encuentra el premaster.

`of` especifica el dispositivo de salida, en este caso una partición de disco.

Una vez hecha la copia procedemos a probarla, la imagen almacenada en disco duro se comportará exactamente como si estuviera en CD-ROM con la diferencia de que el tiempo de acceso es más rápido. Montando la partición:

```
tzetzal:2>mount -F hsfs -o ro /dev/dsk/clt3d0s7 /prueba
```

Con el comando mount montamos al sistema de archivos de UNIX la partición donde nuestra información esta contenida, previa definición del tipo de disco, -F hsfs en este caso del tipo High Sierra File System (tipo del CD-ROM), y especificar dentro de las opciones que sea de solo lectura (ro, read only) y que el punto donde se monta la información es el subdirectorio /prueba en el directorio raíz.

Ya montada la partición, procedemos a hacer pruebas de la información. Las pruebas que se hicieron fueron las siguientes:

1. Acceso de la información.
2. Lectura de la información.
3. Indexamiento y consulta.

La primera prueba consistió en acceder la información desde cualquier punto del sistema de archivos. Esto se hizo sin ningún problema. La segunda fue la lectura de la información, es decir, que el sistema operativo pudiera accederla, para tal efecto nos valimos del comando `more` y `cat` del UNIX que lo que hacen es desplegar el contenido de un archivo. No existió ningún problema.

La prueba más representativa fue el indexamiento de la información por `waisindex` y su consulta con `gopher`.

Al final de las pruebas, solo resta *quemar* el CD-ROM. Nuevamente con el software de YMI, dentro del menú de *Image* seleccionamos *CUT* con lo que la imagen almacenada en `/dev/rmt/1` es enviada al grabador. Después de aproximadamente 74 minutos, el CD-ROM está listo para ser leído en cualquier sistema UNIX y en cualquier PC que tenga una unidad de CD-ROM.


```

# datos
# se eliminan del reporte
#
    for ( i = 1; i <= (NF - 3); i++)
        nombre = nombre " " $i;
        no_cuenta=$(NF - 2);

# En esta línea se encuentran: nombre del plantel, clave del plantel y año
# de ingreso.
    getline;
    for ( i=1; i <= (NF - 2); i++)
        plantel=plantel " " $i;
        clave_p=$(NF - 1);
        ingreso=$(NF);

# En esta línea se encuentran: nombre carrera, clave y semestre. Si se
# trata de un alumno con cambio de carrera o plantel, la historia
# anterior no incluye nombre de carrera ni clave.
    getline;
    if (NF == 1)
    {
        carrera="";
        clave_c="";
    }
    else
    {
        for ( i=1; i <= (NF - 2); i++ )
            carrera=carrera " " $i;
        clave_c=$(NF - 1);
    }
    semestre=$NF;

    # Verifica si el alumno llevo alguna otra carrera.

    getline;
    if ( match($0,/ALUMNO PRESENTA REGISTRO/) )
    {
        Lee_Plantel=1; # Enciende la bandera para leer el plantel
                     # donde presenta otro registro.
        Num_Historia=1;
    }
    else {
        Seq_Historia=0; # Apaga banderas
        Num_Historia=0;
    }
    Cred_Ob_c=$1;
    if ( Seq_Historia )
    {
        Mat_Aprob=$2;
        Cred_Ob="";
        Avance_Cred_Ob="";
    }
    else
    {
        Cred_Ob=$2;
        Avance_Cred_Ob=$3;
        Mat_Aprob=$4;
    }

    getline;
    Mat_Reprob = $1;

    getline;
    Cred_Op_c=$1;
    if (Seq_Historia)
    {
        Cred_Op="";
        Avance_Cred_Op="";
        Total_Mat=$2;
    }
    else

```

```

{
Cred_Op=$2;
Avance_Cred_Op=$3;
Total_Mat=$4;
}
If (Lee_Plantel)
Reg_Plantel=$NF;

getline;
Cred_Tot_c=$1;
if (Seg_Historia)
{
Cred_Tot="";
Avance_Cred="";
Promedio=$2;
}
else
{
Cred_Tot=$2;
Avance_Cred=$3;
Promedio=$4;
}

printf("    NOMBRE: %-35sNo. CUENTA:%s\n",nombre,no_cuenta);
printf("    PLANTEL: %-35sCLAVE:%s\n",plantel,clave_p);
printf("    CARRERA: %-35sCLAVE:%s\n",carrera,clave_c);
printf("    GENERACION: %s\n",ingreso);
printf("    SEMESTRE: %s\n",semestre);
print "    ";

if (Lee_Plantel)
{
print " ALUMNO PRESENTA REGISTRO ACADEMICO EN OTRO PLANTEL";
print " ALUMNO REGISTRADO EN PLANTEL " Reg_Plantel;
print " ";
Lee_Plantel=0;
}

}

print "-----";
printf("    REPORTE DE AVANCE EN CREDITOS          |    REPORTE DE
ASIGNATURAS\n");
printf("-----|-----\n");
printf("    Cubiertos Totales %%Avance          |    APROBADAS :
%2s\n",Mat_Aprob);
printf("    Obligatorios %3s %3s %6s% |    NO_APROBADAS:
%2s\n",Cred_Ob_c, Cred_Ob, Avance_Cred_Ob, Mat_Reprob);
printf("    Optativos %3s %3s %6s% |    TOTAL :
%2s\n",Cred_Op_c, Cred_Op, Avance_Cred_Op, Total_Mat);
printf("    Totales %3s %3s %6s%
|\n",Cred_Tot_c,Cred_Tot,Avance_Cred);
printf("    |    PROMEDIO :
%6s\n",Promedio);
printf("-----\n");
print "    ";

nombre="";
plantel="";
carrera="";

```


1.2 Pruebas finales.

La prueba de nuestro software es un elemento crítico para la garantía de calidad del mismo y representa un último repaso de las especificaciones, del diseño y de la codificación. Asimismo la **Ingeniería del software** nos dice que las pruebas que se hagan a un software (o a un sistema completo) deben de ser exhaustivas y nos da una serie de reglas que sirven como objetivos de la prueba:

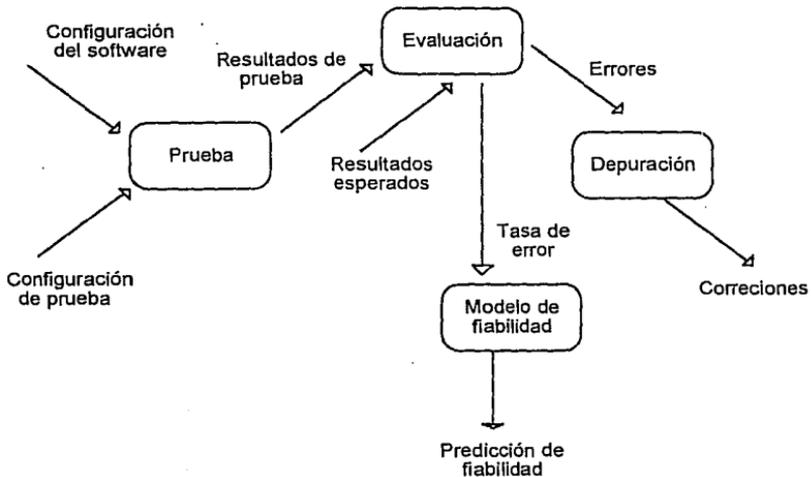
1. La prueba es un proceso de ejecución de un programa con la intención de descubrir algún error.
2. Un buen caso de prueba es aquel que tiene una alta probabilidad de mostrar un error no descubierto hasta entonces.
3. Una prueba tiene éxito si se descubre un error no detectado hasta entonces.

Nos podemos apoyar en estas tres reglas para hacer pruebas a nuestros programas sin embargo, cabe resaltar que los programas realizados no pretenden ser todo un caso de estudio de Ingeniería del software, sino que simplemente son una herramienta que nos permitió depurar la información de las historias académicas. Una frase bastante pesimista en lo que se refiere a las pruebas de software dice: ***"La prueba no puede asegurar la ausencia de defectos; sólo pueden demostrar que existen defectos en el software"***.

Las pruebas finales de los dos programas depuradores expuestos en capítulos anteriores se pueden dividir en dos: la prueba en el ambiente de programación del mainframe **A-12** a través de su lenguaje nativo, **ALGOL** y las pruebas en el

ambiente de la estación de trabajo *SparcStation 10* (corriendo la versión del *UNIX* de SUN la cual es *Solaris 2.2*) y más específicamente las pruebas de la herramienta *AWK* sobre los datos arrojados por la A-12.

A continuación podemos definir las pruebas a través del siguiente diagrama:



La configuración de nuestro software fue explicada en capítulos anteriores, en los cuales se definieron especificaciones y descripciones tanto del lenguaje de programación (ALGOL y la herramienta AWK) como del problema que deben solucionar: **depurar** lo más posible la información generada para la grabación de las historias en microfichas y **formatear** dicha información para su consulta una vez grabada en CD-ROM.

Una vez que pasan por el programa depurador se tiene la siguiente salida:

				08/ABR/94		
ABUNDIZ VELAZQUEZ ISRAEL				9362639-4	1	1
E.N.E.P. ACATLAN ECONOMIA				208	93	
LIC EN ECONOMIA				21	94-1	
72	300	24.00	13			
			3			
0	0	0.00	16			
72	300	24.00	7.53			
PRIMER SEMESTRE						
208	0241	06	GEOGRAFIA ECONOMICA	B OR 93-1	1650718	1151 1
208	0291	06	HISTORIA ECONOMICA GENERAL I	S OR 93-1	1650726	1151 1
208	0322	06	INTRODUCCION A LA ECONOMIA	B OR 93-1	1650745	1151 1
208	0361	06	ECONOMIA POLITICA I	S OR 93-1	1650753	1151 1
208	0368	04	CIES I	B OR 93-1	1650775	1151 1
208	0497	08	MATEMATICAS I	NA EX 94-1	1775914	EA51 2 2
SEGUNDO SEMESTRE						
208	0292	06	HIST ECONOMICA GENERAL II	NA EX 93-2	1718183	EA51 1 1
208	0362	06	ECONOMIA POLITICA II	S OR 93-2	1710573	2251 1
208	0369	04	CIES II	B OR 93-2	1710591	2251 1
208	0500	06	METODOLOGIA DE LAS C SOCIAL	B OR 93-2	1710615	2251 1
208	0755	06	SOCIOLOGIA Y POLITICA I	B OR 93-2	1710621	2251 1
TERCER SEMESTRE						
208	0293	06	HIST ECON DE AMERICA LATINA	B OR 94-1	1765636	1352 1
208	0363	06	ECONOMIA POLITICA III	S OR 94-1	1765664	1351 1
208	0370	04	CIES III	B OR 94-1	1765691	1352 1
208	0756	06	SOCIOLOGIA Y POLITICA II	MB OR 94-1	1765717	1351 1
208	0814	06	TEORIA ECONOMICA I	NP EX 94-1	1775929	EA51 1 1
						1
				08/ABR/94		
ACEVEDO FERRER INGRID				8906263-4	1	1
E.N.E.P. ACATLAN ECONOMIA				208	92	
LIC EN ECONOMIA				21	94-1	
174	300	58.00	30			
			0			
0	0	0.00	30			
174	300	58.00	8.80			
PRIMER SEMESTRE						
208	0241	06	GEOGRAFIA ECONOMICA	MB OR 92-1	1552721	1102 1
208	0291	06	HISTORIA ECONOMICA GENERAL I	MB OR 92-1	1552729	1102 1
208	0322	06	INTRODUCCION A LA ECONOMIA	MB OR 92-1	1552749	1102 1
208	0361	06	ECONOMIA POLITICA I	B OR 92-1	1552757	1102 1
208	0368	04	CIES I	MB OR 92-1	1552782	1102 1
208	0497	08	MATEMATICAS I	S OR 92-1	1552803	1102 1
SEGUNDO SEMESTRE						
208	0292	06	HIST ECONOMICA GENERAL II	B OR 92-2	1606372	2202 1
208	0362	06	ECONOMIA POLITICA II	MB OR 92-2	1606384	2202 1
208	0369	04	CIES II	B OR 92-2	1606398	2201 1
208	0498	08	MATEMATICAS II	B OR 92-2	1606420	2202 1
208	0500	06	METODOLOGIA DE LAS C SOCIAL	MB OR 92-2	1606426	2202 1
208	0755	06	SOCIOLOGIA Y POLITICA I	MB OR 92-2	1606431	2201 1

Etcétera.

Para poder llegar a esta salida el programa tuvo que ser reestructurado varias veces, debido a que los caracteres de control indeseables del archivo en algunas ocasiones no fueron eliminados y debido a la naturaleza de la información estos

estaban en posiciones totalmente aleatorias, por lo que no se pudo establecer un patrón para su eliminación. Sin embargo se puede afirmar que el 95% de estos caracteres fueron eliminados.

Otro problema que se detectó en el programa fue la incapacidad de detectar cierta "basura" del archivo, es decir, dentro de éste se tenían ciertos campos los cuales eran información no útil para la historia o simplemente caracteres fuera de orden o de toda secuencia; por desgracia debido a que estos son totalmente aleatorios el programa depurador no puede saber si estos son información útil o si simplemente es basura. Tal es el caso de la siguiente historia:

SEGUNDO SEMESTRE					
208	0292	06	HIST ECONOMICA GENERAL II		B OR 92-2 1606372 2202 1
208	0362	06	ECONOMIA POLITICA II	1-1-1	MB OR 92-2 1606384 2202 1
208	0369	04	CIES II		B OR 92-2 1606398 2201 1
208	0498	08	MATEMATICAS II	HISTORIAS	B OR 92-2 1606420 2202 1
208	0500	06	METODOLOGIA DE LAS C SOCIAL		MB OR 92-2 1606426 2202 1
208	0755	06	SOCIOLOGIA Y POLITICA I		MB OR 92-2 1606431 2201 1
			TERCER SEMESTRE	CUARTO SEMESTRE	
208	0293	06	HIST ECON DE AMERICA LATINA		B OR 93-1 1650731 1302 1
208	0363	06	ECONOMIA POLITICA III		B OR 93-1 1650758 1302 1
208	0370	04	CIES III		MB OR 93-1 1650783 1352 1
208	0499	08	MATEMATICAS III		MB OR 93-1 1650800 1301 1
208	0756	06	SOCIOLOGIA Y POLITICA II		MB OR 93-1
			DW MATEMATICAS 94-1	NP	3221
208	0814	06	TEORIA ECONOMICA I		MB OR 93-1 1650816 1352 1

Estos caracteres están en la historia por diversos motivos:

- Errores en la captura de datos.
- Modificaciones hechas al momento de procesar.
- Mal manejo de la información en cinta, etc.

Sin embargo para efectos prácticos del proyecto y debido a que estos errores son mínimos se permitió que existieran.

La Ingeniería del software nos dice además que se deben de hacer pruebas de *caja blanca* y *de caja negra*, dependiendo de la complejidad del mismo. Una

prueba de caja blanca se centra en la estructura de control que tiene el programa, además se debe verificar que todas las sentencias del programa al menos se ejecuten una vez y que se ejerciten todas las condiciones lógicas. Las pruebas de caja negra son diseñadas para validar los requerimientos funcionales sin fijarse en el funcionamiento interno del programa, para este caso se diseñan pruebas que cubran cierta área del programa y sea probado. El análisis de los datos de salida puede ser un parámetro a evaluar para ver si se encuentra en los límites aceptables.

Debido a la simplicidad del software desarrollado no fueron necesarias las pruebas de *caja blanca*, la única función del programa es la de leer caracter por caracter el archivo de datos; si este caracter se encuentra entre los de control o dentro de los no deseados, se desecha. El resultado final es almacenado en un archivo de salida.

Sin embargo las pruebas de *caja negra* fueron ampliamente usadas, debido a las características de la información se tuvieron que hacer modificaciones hasta que la información resultante estuviera en un límite aceptable, esto es: menos de 1% - 2% de información no útil o "*basura*".

Otro de los detalles que se pueden observar de las historias es que debido a que la información que presenta solo cabe en una hoja tamaño carta, existen varias historias que se repiten, es decir, se tiene más de una hoja por historias académicas totales (tal es el caso de los alumnos de semestres avanzados) tal es el caso de ACEVEDO PEA OFELIA que tiene dos hojas de historia académica. Se puede observar además que existe un número consecutivo al final de cada una de ellas, este caracter no pudo ser eliminado por el programa en ALGOL.

• Pruebas del programa en AWK.

Una vez que la información se tiene en un formato legible, el siguiente paso fue exportar a la estación de trabajo SparcStation 10 y realizar otro proceso de depuración. Este paso fue mucho más sencillo que en el ambiente A-12, debido a que UNIX nos provee de muchas herramientas para el manejo de textos y caracteres, tal es el caso de la herramienta **AWK**.

El programa se mostró en secciones anteriores y el objetivo de este en primer paso fue la eliminación de las historias repetidas para evitar redundancia en la información y además de cierta forma toda esta información conforma una sola historia. Como segundo paso fue necesario quitar la numeración de cada historia, es decir, el **número de hoja** de cada una de ellas.

El resultado del programa fue el siguiente:

NOMBRE:	ABUNDIZ VELAZQUEZ ISRAEL	No. CUENTA:	9362639-4
PLANTEL:	E.H.E.P. ACATLAN ECONOMIA	CLAVE:	208
CARRERA:	LIC EN ECONOMIA	CLAVE:	21
GENERACION:	93		
SEMESTRE:	94-1		

REPORTE DE AVANCE EN CREDITOS				REPORTE DE ASIGNATURAS	
	Cubiertos	Totales	%Avance	APROBADAS	: 13
obligatorios	72	300	24.00%	NO APROBADAS:	3
Optativos	0	0	0.00%	TOTAL	: 16
Totales	72	300	24.00%	PROMEDIO	: 7.53

PRIMER SEMESTRE					
208 0241 06	GEOGRAFIA ECONOMICA	B OR 93-1	1650718	1151	1
208 0291 06	HISTORIA ECONOMICA GENERAL I	S OR 93-1	1650726	1151	1
208 0322 06	INTRODUCCION A LA ECONOMIA	B OR 93-1	1650745	1151	1
208 0361 06	ECONOMIA POLITICA I	S OR 93-1	1650753	1151	1
208 0368 04	CIES I	B OR 93-1	1650775	1151	1
208 0497 08	MATEMATICAS I	NA EX 94-1	1775914	EA51	2 2
SEGUNDO SEMESTRE					
208 0292 06	HIST ECONOMICA GENERAL II	NA EX 93-2	1718183	EA51	1 1
208 0362 06	ECONOMIA POLITICA II	S OR 93-2	1710573	2251	1
208 0369 04	CIES II	B OR 93-2	1710591	2251	1
208 0500 06	METODOLOGIA DE LAS C SOCIAL	B OR 93-2	1710615	2251	1
208 0755 06	SOCIOLOGIA Y POLITICA I	B OR 93-2	1710621	2251	1
TERCER SEMESTRE					
208 0293 06	HIST ECON DE AMERICA LATINA	B OR 94-1	1765636	1352	1
208 0363 06	ECONOMIA POLITICA III	S OR 94-1	1765664	1351	1
208 0370 04	CIES III	B OR 94-1	1765691	1352	1

Migración de la información al ambiente UNIX.

208 0756 06 SOCIOLOGIA Y POLITICA II
208 0814 06 TEORIA ECONOMICA I

MB OR 94-1 1765717 1351 1
NF EX 94-1 1775929 EA51 1 1

NOMBRE: ACEVEDO FERRER INGRID
PLANTEL: E.N.E.P. ACATLAN ECONOMIA
CARRERA: LIC EN ECONOMIA
GENERACION: 92
SEMESTRE: 94-1

No. CUENTA:8906263-4
CLAVE:208
CLAVE:21

REPORTE DE AVANCE EN CREDITOS

	Cubiertos	Totales	%Avance
Obligatorios	174	300	58.00%
Optativos	0	0	0.00%
Totales	174	300	58.00%

REPORTE DE ASIGNATURAS

APROBADAS : 30
NO APROBADAS: 0
TOTAL : 30
PROMEDIO : 8.80

PRIMER SEMESTRE

208 0241 06 GEOGRAFIA ECONOMICA
208 0291 06 HISTORIA ECONOMICA GENERAL I
208 0322 06 INTRODUCCION A LA ECONOMIA
208 0361 06 ECONOMIA POLITICA I
208 0368 04 CIES I
208 0497 08 MATEMATICAS I

MB OR 92-1 1552721 1102 1
MB OR 92-1 1552729 1102 1
MB OR 92-1 1552749 1102 1
B OR 92-1 1552757 1102 1
MB OR 92-1 1552782 1102 1
S OR 92-1 1552803 1102 1

SEGUNDO SEMESTRE

208 0292 06 HIST ECONOMICA GENERAL II
208 0362 06 ECONOMIA POLITICA II
208 0369 04 CIES II
208 0498 08 MATEMATICAS II
208 0500 06 METODOLOGIA DE LAS C SOCIAL
208 0755 06 SOCIOLOGIA Y POLITICA I

B OR 92-2 1606372 2202 1
MB OR 92-2 1606384 2202 1
B OR 92-2 1606398 2201 1
B OR 92-2 1606420 2202 1
MB OR 92-2 1606426 2202 1
MB OR 92-2 1606431 2201 1

TERCER SEMESTRE

208 0293 06 HIST ECON DE AMERICA LATINA
208 0363 06 ECONOMIA POLITICA III
208 0370 04 CIES III
208 0499 08 MATEMATICAS III
208 0756 06 SOCIOLOGIA Y POLITICA II
208 0814 06 TEORIA ECONOMICA I

B OR 93-1 1650731 1302 1
B OR 93-1 1650758 1302 1
MB OR 93-1 1650783 1352 1
MB OR 93-1 1650800 1301 1
MB OR 93-1 1650806 1301 1
MB OR 93-1 1650816 1352 1

CUARTO SEMESTRE

208 0004 04 ANALISIS D EDOS FINANCIEROS
208 0164 08 ESTADISTICA I
208 0294 06 HIST ECONOMICA D MEXICO I
208 0364 06 ECONOMIA POLITICA IV
208 0371 04 CIES IV
208 0815 06 TEORIA ECONOMICA II

B OR 93-2 1710531 2452 1
B OR 93-2 1710548 2451 1
S OR 93-2 1710566 2451 1
B OR 93-2 1710580 2452 1
B OR 93-2 1710599 2453 1
B OR 93-2 1710627 2451 1

QUINTO SEMESTRE

208 0099 04 CONTABILIDAD SOCIAL
208 0165 08 ESTADISTICA II
208 0295 04 HIST ECONOMICA D MEXICO II
208 0365 06 ECONOMIA POLITICA V
208 0372 04 CIES V
208 0816 06 TEORIA ECONOMICA III

B OR 94-1 1765590 1551 1
B OR 94-1 1765602 1552 1
MB OR 94-1 1765642 1552 1
MB OR 94-1 1765669 1551 1
MB OR 94-1 1765698 1553 1
B OR 94-1 1765729 1551 1

NOMBRE: ACEVEDO PEA OFELIA
PLANTEL: E.N.E.P. ACATLAN ECONOMIA
CARRERA: LIC EN ECONOMIA
GENERACION: 90
SEMESTRE: 94-1

No. CUENTA:9058174-8
CLAVE:208
CLAVE:21

REPORTE DE AVANCE EN CREDITOS

	Cubiertos	Totales	%Avance
Obligatorios	262	300	87.33%
Optativos	14	0	0.00%
Totales	276	300	92.00%

REPORTE DE ASIGNATURAS

APROBADAS : 54
NO APROBADAS: 0
TOTAL : 54
PROMEDIO : 9.18

PRIMER SEMESTRE					
208 0241 06	GEOGRAFIA ECONOMICA	MB	OR 90-1	1353260	1102 1
208 0291 06	HISTORIA ECONOMICA GENERAL I	MB	OR 90-1	1353271	1102 1
208 0322 06	INTRODUCCION A LA ECONOMIA	MB	OR 90-1	1353299	1102 1
208 0361 06	ECONOMIA POLITICA I	MB	OR 90-1	1353310	1102 1
208 0368 04	CIES I	MB	OR 90-1	1353342	1102 1
208 0497 08	MATEMATICAS I	B	OR 90-1	1353369	1102 1
SEGUNDO SEMESTRE					
208 0292 06	HIST ECONOMICA GENERAL II	MB	OR 90-2	1401357	2202 1
208 0362 06	ECONOMIA POLITICA II	MB	OR 90-2	1401372	2201 1
208 0369 04	CIES II	B	OR 90-2	1401398	2202 1
208 0498 08	MATEMATICAS II	B	EX 90-2	1414821	EA01 1 1
208 0500 06	METODOLOGIA DE LAS C SOCIAL	MB	OR 90-2	1401428	2201 1
208 0755 06	SOCIOLOGIA Y POLITICA I	MB	OR 90-2	1401437	2202 1
TERCER SEMESTRE					
208 0293 06	HIST ECON DE AMERICA LATINA	MB	OR 91-1	1457315	1301 1
208 0363 06	ECONOMIA POLITICA III	MB	OR 91-1	1457352	1301 1
208 0370 04	CIES III	MB	OR 91-1	1457385	1301 1
208 0499 08	MATEMATICAS III	S	OR 91-1	1457414	1302 1
208 0756 06	SOCIOLOGIA Y POLITICA II	B	OR 91-1	1457421	1302 1
208 0814 06	TEORIA ECONOMICA I	MB	OR 91-1	1457427	1301 1
CUARTO SEMESTRE					
208 0004 04	ANALISIS D EDOS FINANCIEROS	S	OR 91-2	1500787	2401 1
208 0164 08	ESTADISTICA I	MB	OR 91-2	1500813	2401 1
208 0294 06	HIST ECONOMICA D MEXICO I	MB	OR 91-2	1500829	2401 1
208 0364 06	ECONOMIA POLITICA IV	MB	OR 91-2	1500840	2401 1
208 0371 04	CIES IV	MB	OR 91-2	1500863	2404 1
208 0815 06	TEORIA ECONOMICA II	S	OR 91-2	1500897	2401 1
QUINTO SEMESTRE					
208 0099 04	CONTABILIDAD SOCIAL	B	OR 92-1	1552689	1501 1
208 0165 08	ESTADISTICA II	MB	OR 92-1	1552702	1501 1
208 0295 04	HIST ECONOMICA D MEXICO II	MB	OR 92-1	1552742	1502 1
208 0365 06	ECONOMIA POLITICA V	MB	OR 92-1	1552768	1501 1
208 0372 04	CIES V	MB	OR 92-1	1552796	1503 1
208 0816 06	TEORIA ECONOMICA III	MB	OR 92-1	1552828	1502 1
SEXTO SEMESTRE					
208 0005 04	ANAL DEL CAMBIO SOC EN MEXCO	B	OR 92-2	1606348	2601 1
208 0131 04	DESARROLLO Y SUBDESLEO I	MB	OR 92-2	1606354	2601 1

Como se puede observar también se eliminó la fecha de la elaboración de la historia ya que para efectos prácticos es un dato poco usado dentro de la misma.

Sin embargo dentro de las pruebas no fue posible percatarse de un problema que no se podía ver en la estación de trabajo: la consulta de la información, en la mayoría de los casos, será hecha vía PC en red; algunas veces será consultada por otras estaciones de trabajo. Como se sabe la capacidad de desplegar texto en un monitor de PC es de 80 columnas⁷ (los renglones no importan ya que la aplicación permite el uso de las flechas de navegación del cursor). Si se observa con atención las historias estas rebasan dicho límite por lo que las historias se verán así:

⁷ Cabe señalar que los minitores VGA pueden desplegar más de 80 columnas en modo texto, sin embargo no todas las PC's conectadas a red tienen este tipo de monitor, por lo que es bueno estandarizarlo a 80 columnas.

Migración de la información al ambiente UNIX.

CUARTO SEMESTRE

208	0004	04	ANALISIS D EDOS FINANCIEROS	B OR 93-2	1710531
2452	1				
208	0164	08	ESTADISTICA I	B OR 93-2	1710548
2451	1				
208	0294	06	HIST ECONOMICA D MEXICO I	S OR 93-2	1710566
2451	1				
208	0364	06	ECONOMIA POLITICA IV	B OR 93-2	1710580
2452	1				
208	0371	04	CIES IV	B OR 93-2	1710599
2453	1				
208	0815	06	TEORIA ECONOMICA II	B OR 93-2	1710627
2451	1				

Por lo que el programa fue depurado para eliminar caracteres en blanco al principio y al fin de cada renglón de la historia, a continuación se muestra la salida de éste:

NOMBRE: ABUNDIZ VELAZQUEZ ISRAEL
 PLANTEL: E.N.E.P. ACATLAN ECONOMIA
 CARRERA: LIC EN ECONOMIA
 GENERACION: 93
 SEMESTRE: 94-1

No. CUENTA:9362639-4
 CLAVE:208
 CLAVE:21

 REPORTE DE AVANCE EN CREDITOS

	Cubiertos	Totales	%Avance
Obligatorios	72	300	24.00%
Optativos	0	0	0.00%
Totales	72	300	24.00%

REPORTE DE ASIGNATURAS

APROBADAS	: 13
NO_APROBADAS:	3
TOTAL	: 16
PROMEDIO	: 7.53

PRIMER SEMESTRE

208	0241	06	GEOGRAFIA ECONOMICA	B OR 93-1	1650718	1151	1
208	0291	06	HISTORIA ECONOMICA GENERAL I	S OR 93-1	1650726	1151	1
208	0322	06	INTRODUCCION A LA ECONOMIA	B OR 93-1	1650745	1151	1
208	0361	06	ECONOMIA POLITICA I	S OR 93-1	1650753	1151	1
208	0368	04	CIES I	B OR 93-1	1650775	1151	1
208	0497	08	MATEMATICAS I	NA EX 94-1	1775914	EA51	2 2

SEGUNDO SEMESTRE

208	0292	06	HIST ECONOMICA GENERAL II	NA EX 93-2	1718183	EA51	1 1
208	0362	06	ECONOMIA POLITICA II	S OR 93-2	1710573	2251	1
208	0369	04	CIES II	B OR 93-2	1710591	2251	1
208	0500	06	METODOLOGIA DE LAS C SOCIAL	B OR 93-2	1710615	2251	1
208	0755	06	SOCIOLOGIA Y POLITICA I	B OR 93-2	1710621	2251	1

TERCER SEMESTRE

208	0293	06	HIST ECON DE AMERICA LATINA	B OR 94-1	1765636	1352	1
208	0363	06	ECONOMIA POLITICA III	S OR 94-1	1765664	1351	1
208	0370	04	CIES III	B OR 94-1	1765691	1352	1
208	0756	06	SOCIOLOGIA Y POLITICA II	MB OR 94-1	1765717	1351	1
208	0814	06	TEORIA ECONOMICA I	NP EX 94-1	1775929	EA51	1 1

NOMBRE: ACEVEDO FERRER INGRID
 PLANTEL: E.N.E.P. ACATLAN ECONOMIA
 CARRERA: LIC EN ECONOMIA
 GENERACION: 92
 SEMESTRE: 94-1

No. CUENTA:8906263-4
 CLAVE:208
 CLAVE:21

Capitulo III Diseño de la propuesta.

REPORTE DE AVANCE EN CREDITOS				REPORTE DE ASIGNATURAS	
Obligatorios	Cubiertos	Totales	%Avance	APROBADAS	: 30
Optativos	0	0	0.00%	NO APROBADAS:	0
Totales	174	300	58.00%	TOTAL	: 30
				PROMEDIO	: 8.80

PRIMER SEMESTRE					
208 0241 06	GEOGRAFIA ECONOMICA	MB OR 92-1	1552721	1102	1
208 0291 06	HISTORIA ECONOMICA GENERAL I	MB OR 92-1	1552729	1102	1
208 0322 06	INTRODUCCION A LA ECONOMIA	MB OR 92-1	1552749	1102	1
208 0361 06	ECONOMIA POLITICA I	B OR 92-1	1552757	1102	1
208 0368 04	CIES I	MB OR 92-1	1552782	1102	1
208 0497 08	MATEMATICAS I	S OR 92-1	1552803	1102	1
SEGUNDO SEMESTRE					
208 0292 06	HIST ECONOMICA GENERAL II	B OR 92-2	1606372	2202	1
208 0362 06	ECONOMIA POLITICA II	MB OR 92-2	1606384	2202	1
208 0369 04	CIES II	B OR 92-2	1606398	2201	1
208 0498 08	MATEMATICAS II	B OR 92-2	1606420	2202	1
208 0500 06	METODOLOGIA DE LAS C SOCIAL	MB OR 92-2	1606426	2202	1
208 0755 06	SOCIOLOGIA Y POLITICA I	MB OR 92-2	1606431	2201	1
TERCER SEMESTRE					
208 0293 06	HIST ECON DE AMERICA LATINA	B OR 93-1	1650731	1302	1
208 0363 06	ECONOMIA POLITICA III	B OR 93-1	1650758	1302	1
208 0370 04	CIES III	MB OR 93-1	1650783	1352	1
208 0499 08	MATEMATICAS III	MB OR 93-1	1650800	1301	1
208 0756 06	SOCIOLOGIA Y POLITICA II	MB OR 93-1	1650806	1301	1
208 0814 06	TEORIA ECONOMICA I	MB OR 93-1	1650816	1352	1
CUARTO SEMESTRE					
208 0004 04	ANALISIS D EDOS FINANCIEROS	B OR 93-2	1710531	2452	1
208 0164 08	ESTADISTICA I	B OR 93-2	1710548	2451	1
208 0294 06	HIST ECONOMICA D MEXICO I	S OR 93-2	1710566	2451	1
208 0364 06	ECONOMIA POLITICA IV	B OR 93-2	1710580	2452	1
208 0371 04	CIES IV	B OR 93-2	1710599	2452	1
208 0815 06	TEORIA ECONOMICA II	B OR 93-2	1710627	2451	1
QUINTO SEMESTRE					
208 0099 04	CONTABILIDAD SOCIAL	B OR 94-1	1765590	1551	1
208 0165 08	ESTADISTICA II	B OR 94-1	1765602	1552	1
208 0295 04	HIST ECONOMICA D MEXICO II	MB OR 94-1	1765642	1552	1
208 0365 06	ECONOMIA POLITICA V	MB OR 94-1	1765669	1551	1
208 0372 04	CIES V	MB OR 94-1	1765698	1553	1
208 0816 06	TEORIA ECONOMICA III	B OR 94-1	1765729	1551	1

NOMBRE: ACEVEDO PEA OFELIA
 PLANTEL: E.N.E.P. ACATLAN ECONOMIA
 CARRERA: LIC EN ECONOMIA
 GENERACION: 90
 SEMESTRE: 94-1

No. CUENTA:9058174-8
 CLAVE:208
 CLAVE:21

REPORTE DE AVANCE EN CREDITOS				REPORTE DE ASIGNATURAS	
Obligatorios	Cubiertos	Totales	%Avance	APROBADAS	: 54
Optativos	14	0	0.00%	NO APROBADAS:	0
Totales	276	300	92.00%	TOTAL	: 54
				PROMEDIO	: 9.18

PRIMER SEMESTRE					
208 0241 06	GEOGRAFIA ECONOMICA	MB OR 90-1	1353260	1102	1
208 0291 06	HISTORIA ECONOMICA GENERAL I	MB OR 90-1	1353271	1102	1
208 0322 06	INTRODUCCION A LA ECONOMIA	MB OR 90-1	1353299	1102	1
208 0361 06	ECONOMIA POLITICA I	MB OR 90-1	1353310	1102	1
208 0368 04	CIES I	MB OR 90-1	1353342	1102	1
208 0497 08	MATEMATICAS I	B OR 90-1	1353369	1102	1
SEGUNDO SEMESTRE					
208 0292 06	HIST ECONOMICA GENERAL II	MB OR 90-2	1401357	2202	1
208 0362 06	ECONOMIA POLITICA II	MB OR 90-2	1401372	2201	1
208 0369 04	CIES II	B OR 90-2	1401398	2202	1

Migración de la información al ambiente UNIX.

208 0498 08 MATEMATICAS II	B EX 90-2 1414821 EA01 1 1
208 0500 06 METODOLOGIA DE LAS C SOCIAL	MB OR 90-2 1401428 2201 1
208 0755 06 SOCIOLOGIA Y POLITICA I	MB OR 90-2 1401437 2202 1
TERCER SEMESTRE	
208 0293 06 HIST ECON DE AMERICA LATINA	MB OR 91-1 1457315 1301 1
208 0363 06 ECONOMIA POLITICA III	MB OR 91-1 1457352 1301 1
208 0370 04 CIES III	MB OR 91-1 1457385 1301 1
208 0499 08 MATEMATICAS III	S OR 91-1 1457414 1302 1
208 0756 06 SOCIOLOGIA Y POLITICA II	B OR 91-1 1457421 1302 1
208 0814 06 TEORIA ECONOMICA I	MB OR 91-1 1457427 1301 1
CUARTO SEMESTRE	
208 0004 04 ANALISIS D EDOS FINANCIEROS	S OR 91-2 1500787 2401 1
208 0164 08 ESTADISTICA I	MB OR 91-2 1500813 2401 1
208 0294 06 HIST ECONOMICA D MEXICO I	MB OR 91-2 1500829 2401 1
208 0364 06 ECONOMIA POLITICA IV	MB OR 91-2 1500840 2401 1
208 0371 04 CIES IV	MB OR 91-2 1500863 2404 1
208 0815 06 TEORIA ECONOMICA II	S OR 91-2 1500897 2401 1
QUINTO SEMESTRE	
208 0099 04 CONTABILIDAD SOCIAL	B OR 92-1 1552689 1501 1
208 0165 08 ESTADISTICA II	MB OR 92-1 1552702 1501 1
208 0295 04 HIST ECONOMICA D MEXICO II	MB OR 92-1 1552742 1502 1
208 0365 06 ECONOMIA POLITICA V	MB OR 92-1 1552768 1501 1
208 0372 04 CIES V	MB OR 92-1 1552796 1503 1
208 0816 06 TEORIA ECONOMICA III	MB OR 92-1 1552828 1502 1
SEXTO SEMESTRE	
208 0005 04 ANAL DEL CAMBIO SOC EN MEXCO	B OR 92-2 1606348 2601 1
208 0131 04 DESARROLLO Y SUBDESLLLO I	MB OR 92-2 1606354 2601 1

El resultado: historias académicas que son vistas en cualquier monitor de PC conectado a red.

Con este último programa se cierran las pruebas de depuración de la información y solo resta la implementación para ser visualizadas y ser consultadas vía red.

5. Consulta vía RedUNAM

Finalizado el sistema, solo resta explicar la forma en que se accesa, ya sea desde una PC con capacidades gráficas o desde una estación de trabajo.

Como se explicó anteriormente, para tener acceso al sistema se tiene una clave pública denominada **sinac**, esta clave pública ejecuta un programa (script) de inicialización del sistema llamado **inicio**. Para no perder seguridad del sistema, dentro del archivo `/etc/passwd` de UNIX, se especifica que este es el *shell* por default, es decir, en vez de tener un intérprete de comandos y dentro de él un archivo de inicialización como el `.profile` o `.cshrc` ejecutamos directamente el script `inicio`, de esta forma si por algún motivo el usuario del sistema tiene algún problema y aborta, el usuario no queda dentro de una sesión en UNIX, sino que simplemente su conexión se cierra, finalizando la sesión. Este script es el siguiente:

La primera línea del programa especifica que se ejecutará un `cshell` como intérprete de los comandos que siguen a continuación.

```
#!/bin/csh -f
```

Como siguiente paso, definimos el tipo de terminal `vt100`, para despliegue de texto.

```
setenv TERM vt100
```

Definimos la tecla BackSpace como tecla para borrar, esto es debido a que el usuario está más acostumbrado a borrar con esta tecla que con cualquier otra:

```
stty erase '^H'
```

Fijamos la ruta que seguira el shell para ejecutar algún programa que se solicite: (todo el path está contenido en una sola línea):

```
set path=(/bin /usr/bin /usr/local/bin /usr/openwin/bin  
/usr/openwin/lib .)
```

En caso de que se posea capacidades gráficas para la presentación de Mosaic, definimos la ruta en donde se encuentran dichas librerías:

```
setenv LD_LIBRARY_PATH  
/usr/openwin/lib:/usr/lib:/opt/SUNWmotif/lib
```

Si existiera alguna interrupción por parte del usuario (tal y como un ^C o alguna otra combinación de teclas que haga que el sistema aborte), el programa salta hasta la etiqueta de fallo:

```
onintr goto fallo
```

Damos una pequeña interfaz de bienvenida:

```
inicio:  
clear  
echo ""  
echo ""  
echo ""  
echo ""  
echo "                BIENVENIDO"  
echo "                A      "  
echo "                SINAC  "  
echo ""  
echo ""  
echo ""  
echo "                SISTEMA DE INFORMACION ACADEMICA."  
echo ""  
echo ""  
echo ""  
echo ""
```

Pedimos al usuario su dirección IP y la almacenamos en la variable `dir`:

```
echo "Por favor teclee su direccion IP (Por ej.: 132.248.27.10) :"  
set dir = $<
```

Indicamos al usuario que especifique si su monitor tiene la capacidad gráfica X11 o no, esto con el objeto de arrancar Gopher para monitores sin capacidad gráfica o Mosaic para lo que poseen capacidades gráficas, almacenamos el resultado en la variable `resp`.

```
echo "su monitor presenta capacidades graficas (X11, XTerm, etc.)  
S/N/Q :"  
set resp = $<
```

Después procesamos las respuestas del usuario en el siguiente `switch` con el objeto de decidir si se ejecuta Gopher o Mosaic.

```
switch ($resp)
```

En el caso que tenga capacidades gráficas se define una variable de ambiente denominada `DISPLAY` esto con el objeto de que nuestra máquina que funge como servidor (`tzetzal`) tenga conocimiento de hacia donde debe de desplegar el ambiente gráfico, en este caso, la máquina del usuario. A continuación ejecutamos Mosaic con la opción `-home` es decir, con esta opción le indicamos cual será la primera pantalla que debe de visualizarse, en este caso `histo_hp.html` además direccionamos los mensajes que pudiera arrojar Mosaic hacia en `/dev/null`.

```
case [sS]:  
    setenv DISPLAY ${dir}:0.0  
    /opt/Mosaic/Mosaic-solaris -home  
http://tzetzal.dcaa.unam.mx/bases/sinac/histo_hp.html  
>& /dev/null  
    breaksw
```

En el caso de que no se tengan capacidades gráficas, se corre Gopher definimos nuevamente el tipo de terminal `vt100` para texto y lo ejecutamos:

```
case [nN]:
    setenv TERM vt100
    /home/gopher/gopher2.014/gopher/gopher tzetzal 1300
breaksw
```

En el caso que el usuario haya introducido Q (quit) el script salta a la etiqueta `fin`, finalizando el script.

```
case [qQ]:
    goto fin
breaksw
```

Si el usuario presiona cualquier otra tecla que no sean las definidas, entonces salta a la etiqueta `inicio`.

```
default:
    goto inicio
breaksw
endsw
```

Termina el script.
`goto fin`

Si hubo alguna interrupción:

```
fallo:
echo "El usuario ha interrumpido la sesion...."
fin:
```

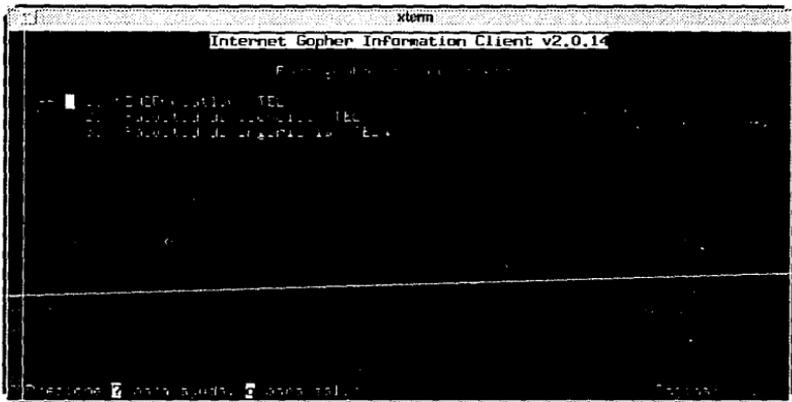
Como se puede ver este es un programa muy simple, pero que permite al usuario entrar a nuestro sistema. Una vez especificado esto, el usuario tiene acceso al sistema conectandose a `tzetzal.dcaa.unam.mx`, esto a través de `telnet`. Cuando se pida la clave del usuario se debe de teclear `sinac`, después se corra el script anterior y se accesa al sistema.

A continuación veremos el sistema final, primero la interfaz de texto con Gopher como front-end y WAIS como search engine.

INTERFAZ CUI

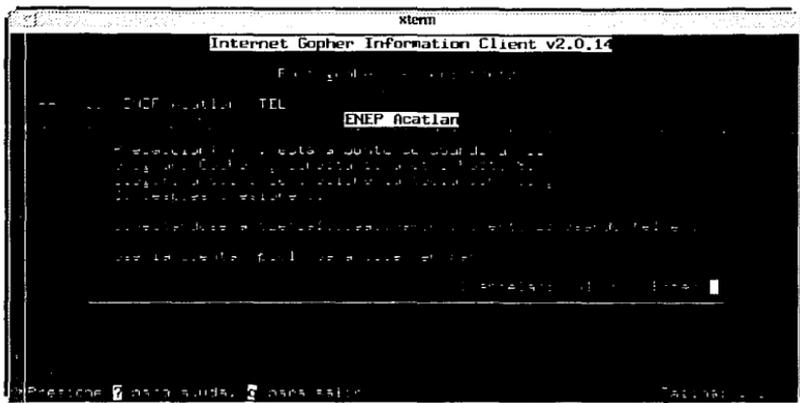
Antes debemos aclarar que las pantallas que se verán a continuación fueron utilizadas con el propósito de mostrar el sistema final, solo contienen algunas escuelas y facultades, no contienen toda la información que el sistema ofrece a los usuarios debido al carácter de la información que se maneja.

La primera pantalla que se visualiza en la siguiente:

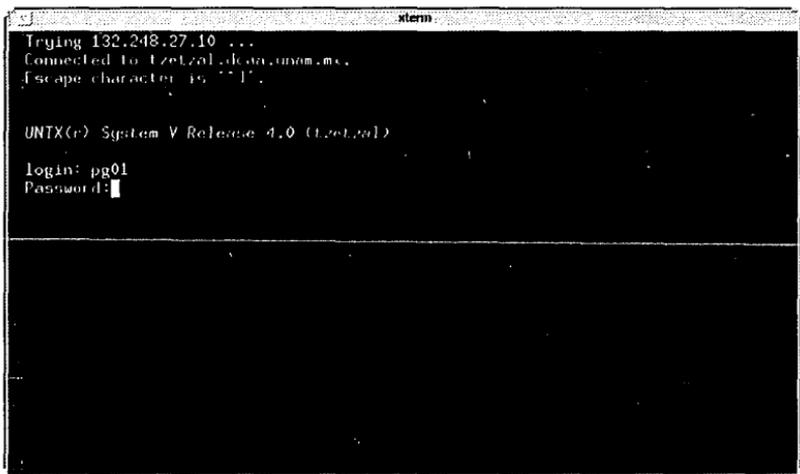


Como se puede observar se tienen tres opciones, al final de cada opción se puede ver la leyenda <TEL>, lo que significa que el acceso a la información se hará mediante una reconexión al sistema, vía telnet.

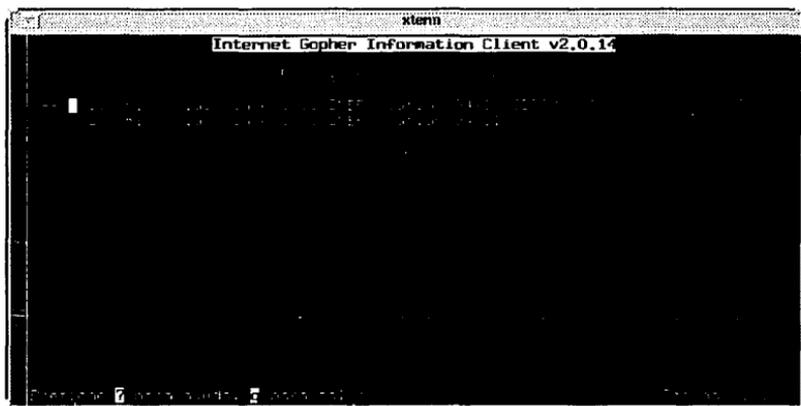
Seleccionando ENEP ACATLAN se visualiza lo siguiente:



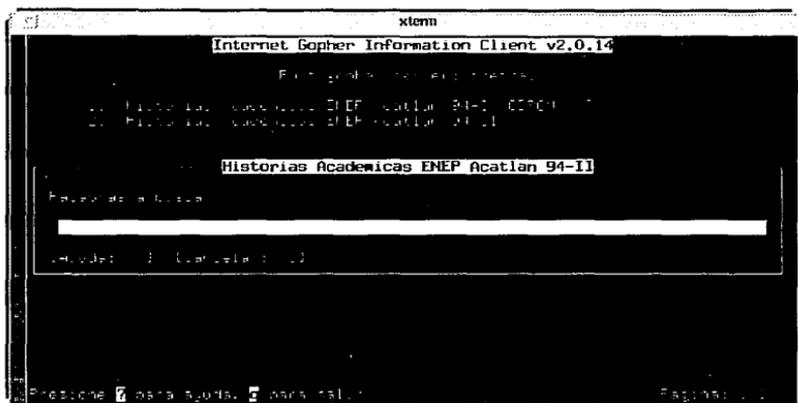
Lo que indica que se hará una reconexión a otra clave para tener acceso a la información de la ENEP, se especifica el nombre de la clave de acceso en la pantalla; el usuario debe saber el password, he aquí implementado un nivel de seguridad a nivel reconexión.



Una vez que el usuario ha tecleado el password correcto se tiene acceso a la información, en este caso el semestre actual (almacenado en disco) y el semestre anterior (almacenado en CDROM).



El usuario entonces puede seleccionar cualquier item, inmediatamente después se presenta una ventana en donde el usuario puede teclear alguna consulta.



En este punto el usuario puede optar por hacer consultas booleanas, exactas o por medio de algún metacaracter como puede ser el * o ? .

Por ejemplo, se puede efectuar una búsqueda exacta por número de cuenta, lo que tiene que hacer el usuario simplemente es teclear el número de cuenta de cualquier alumno y el sistema traerá la historia académica respectiva.

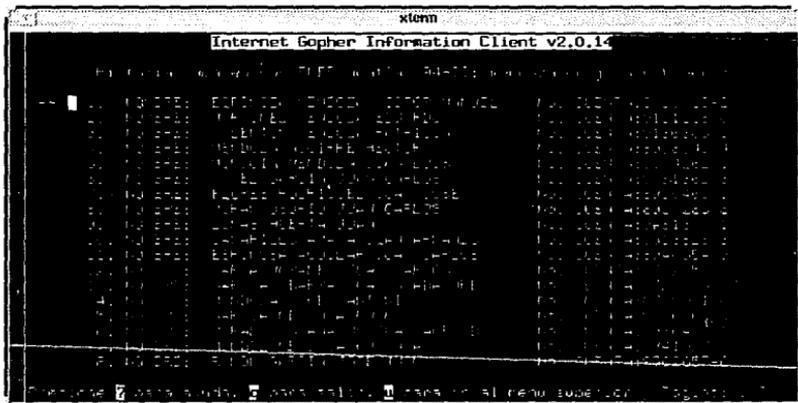


También se pueden hacer búsquedas booleanas, introduciendo los tres operadores lógicos básicos: and, or y not. La forma es la siguiente:

```
item1 <operador> item2 <operador> ....
```

Por ejemplo, la búsqueda de los alumnos que se apelliden Mendoza o se llame Juan pero que no sea Perez se haría de la siguiente forma:

```
MENDOZA or JUAN not PEREZ
```

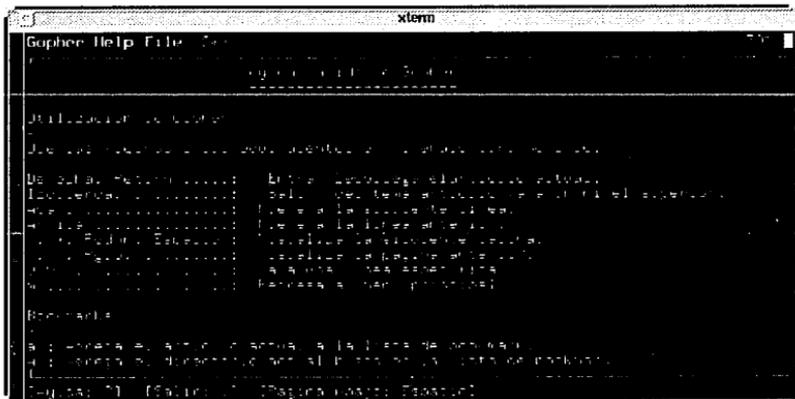


También búsquedas incluyendo metacaracteres como * el cual se sustituye por cualquier caracter. Por ejemplo todos los números de cuenta que comiencen con 8906*

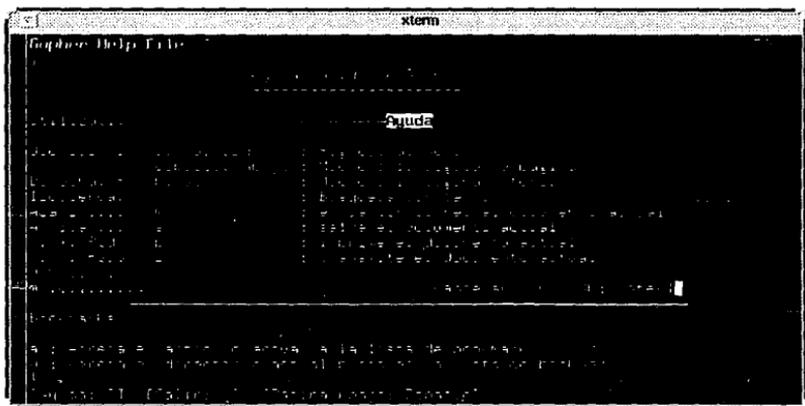


Al presionar <ENTER> en cualquiera de los ítems presentados, el sistema responderá trayendo la historia académica respectiva.

En cualquier momento se puede solicitar ayuda a Gopher presionando la tecla ? . La ayuda que presenta es la siguiente.



También se tiene una ayuda acerca de lo que se puede hacer con la historia, esto es, se puede imprimir, enviar por correo electrónico, transferirla hacia un archivo, etc.

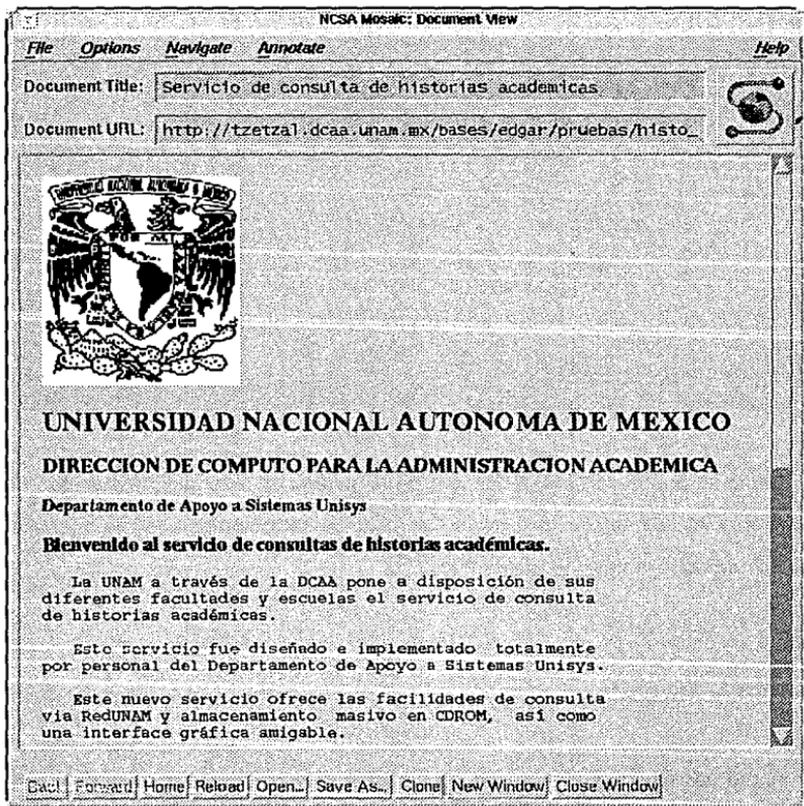


Como se puede ver, la interfaz que da al usuario es muy fácil de manejar y es amigable. Al salir de sesión regresará al menú de selección de escuela o facultad, al salir de esta pantalla, el sistema finaliza.

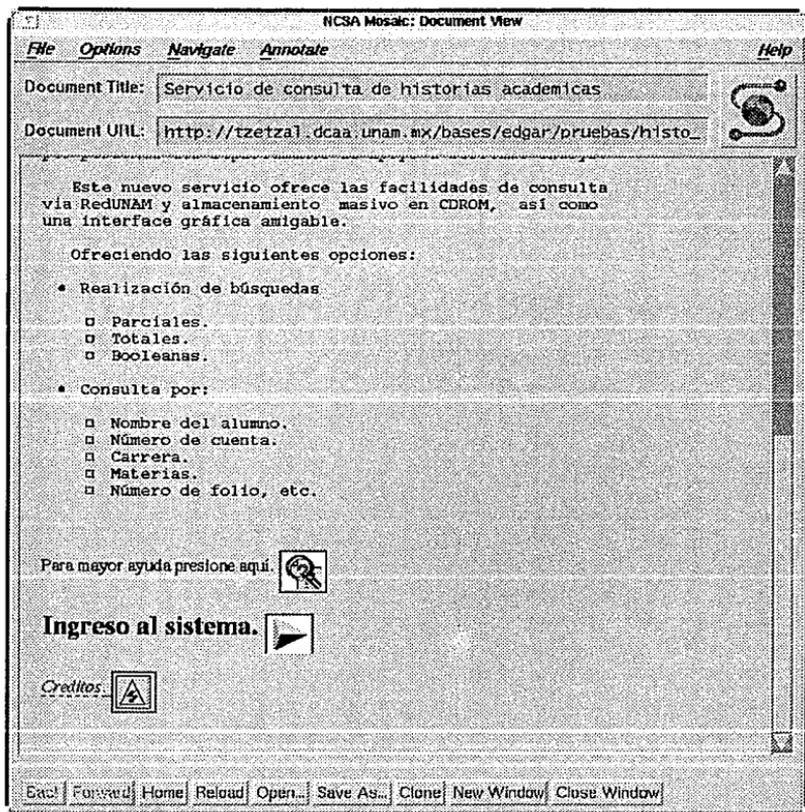
INTERFAZ GRÁFICA.

Ahora veremos la parte gráfica del sistema: la interfaz utilizando Mosaic como front-end y WAIS como search-engine.

Cada pantalla que se presenta corresponde al sistema final y se debe de recordar que cada una de ellas está programada en HTML. De entrada el sistema visualiza el *home page* del sistema.

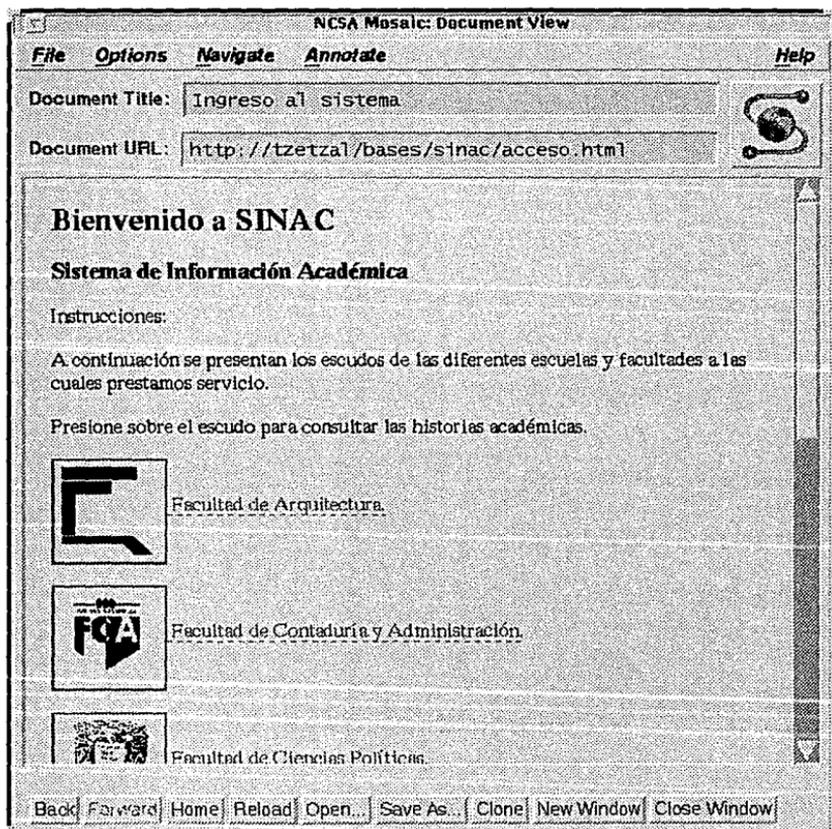


La parte inferior:



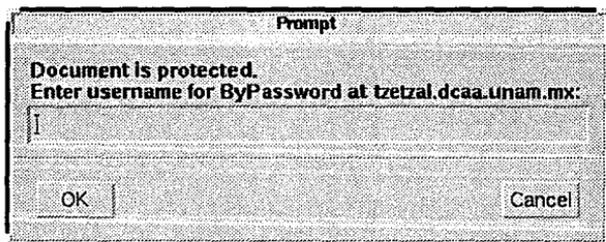
Como se puede observar, esta interfaz es mucho más amigable que la de Gopher, permite la presentación de imágenes, así como de hiperligas a otros documentos. El ambiente de Mosaic provee de mucho mayor número de funciones. Para una mayor explicación de cada una de las funciones de Mosaic, véase el apéndice A.

En la pantalla se tienen tres iconos, los cuales son *hiperligas* a tres documentos diferentes, uno de ayuda (muy simple), el acceso al sistema y los créditos. Presionando el de acceso al sistema se visualiza lo siguiente:

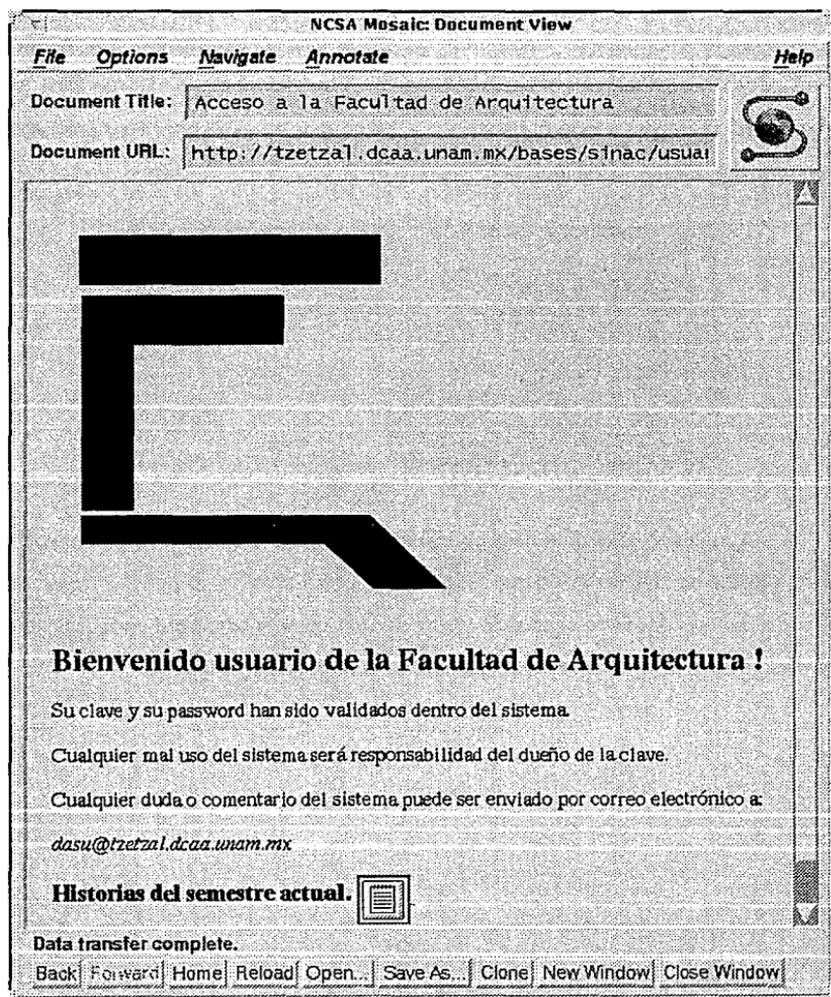


En esta pantalla se presentan todos los escudos de las diferentes escuelas y facultades, dichos escudos son hiperligas a los documentos llave para poder

consultar la información. Cada uno de ellos esta protegido por un archivo .htaccess, explicado en el capítulo anterior. Al presionar cualquiera de ellos, se busca dicho archivo y pide al usuario que introduzca su *username* y su *password* en una ventana como la siguiente:



Si el usuario se identifica correctamente, tiene acceso al sistema de consulta, sino Mosaic de la posibilidad de reintentar, de otra forma no se puede ir más allá de esa pantalla. Cuando el usuario teclea su username y password correctamente se visualiza la siguiente pantalla, que contiene el documento llave.



Como se puede observar en la parte inferior se tiene el ícono de consulta, al presionarlo se pueden hacer las consultas de forma igual que con Gopher anteriormente. En este ejemplo se busca por Perez:

NCSA Mosaic: Document View

File Options Navigate Annotate Help

Document Title: PEREZ (in Sem_actua1)

Document URL: wais://tztetza1.dcaa.unam.mx:1500/Sem_actua1?PEREZ

PEREZ

This is a searchable index. Enter search keywords:

Index Sem_actua1 contains the following 19 items relevant to 'PEREZ'. The first figure for each entry is its relative score, the second the number of lines in the item.

- 1000 25 NOMBRE: GONZALEZ PEREZ RAYMUNDO No. CUENTA:9461254-4
- 926 26 NOMBRE: TOPRES PEREZ SERGIO No. CUENTA:9461240-3
- 918 26 NOMBRE: PEREZ GUTIERREZ OSVALDO A No. CUENTA:8914609-7
- 918 26 NOMBRE: RIVERA PEREZ KEYNA ANGELICA No. CUENTA:9110754-5
- 901 26 NOMBRE: ROMERO PEREZ MAURICIO No. CUENTA:9452672-3
- 435 39 NOMBRE: LUNA PEREZ ANA LILIA No. CUENTA:8906810-0
- 431 40 NOMBRE: RODRIGUEZ PEREZ AGUSTIN No. CUENTA:8909741-8
- 280 54 NOMBRE: PEREZ RIVERA NORMA No. CUENTA:8909558-0
- 278 54 NOMBRE: CERROS PEREZ CARMEN No. CUENTA:8917660-1
- 269 54 NOMBRE: HERNANDEZ PEREZ JUAN MANUEL No. CUENTA:8909183-2
- 228 64 NOMBRE: PEREZ ESPAA OSCAR SAMUEL No. CUENTA:8605692-2
- 222 67 NOMBRE: MARTINEZ PEREZ JESUS No. CUENTA:8958139-1
- 216 69 NOMBRE: PEREZ MARTINEZ ALEJANDRO No. CUENTA:8718817-6
- 199 74 NOMBRE: GARCIA PEREZ DANIEL No. CUENTA:8202315-7
- 167 82 NOMBRE: LINO PEREZ CLAUDIA No. CUENTA:8958177-9
- 167 82 NOMBRE: SALONES PEREZ EMILIANO No. CUENTA:9718976-4

Back Forward Home Reload Open... Save As... Clone New Window Close Window

El sistema nos regresa todas las posibles ocurrencias de la palabra y para poder visualizar la historia respectiva, simplemente presionamos en el nombre y el sistema nos regresa la información respectiva:

File Options Navigate Annotate Help

Document Title: NOMBRE: LINO PEREZ CLAUDIA No. CUENTA:9058177-9

Document URL: wais://tzetzal.dcaa.unam.mx:1500/Sem_actual/TEXT/5966/1-tz

NOMBRE: LINO PEREZ CLAUDIA No. CUENTA:9058177-9
 PLANTEL: E.N.E.P. ACATLAN ECONOMIA CLAVE:208
 CARRERA: LIC EN ECONOMIA CLAVE: 21
 GENERACION: 90
 SEMESTRE: 94-1

REPORTE DE AVANCE EN CREDITOS				REPORTE DE ASIGNATURAS	
Obligatorios	Cubiertos	Totales	%Avance	APROBADAS	: 51
Optativos	0	0	0.00%	NO APROBADAS:	0
Totales	276	300	92.00%	TOTAL	: 51
				PROMEDIO	: 0.66

PRIMER SEMESTRE

208 0241 06 GEOGRAFIA ECONOMICA	MB OR 90-1 1353261 1102 1
208 0291 06 HISTORIA ECONOMICA GENERAL I	B OR 90-1 1353272 1102 1
208 0322 06 INTRODUCCION A LA ECONOMIA	S OR 90-1 1353300 1102 1
208 0361 06 ECONOMIA POLITICA I	S OR 90-1 1353311 1102 1
208 0368 04 CIES I	MB OR 90-1 1353343 1102 1
208 0497 08 MATEMATICAS I	S OR 90-1 1353370 1102 1

SEGUNDO SEMESTRE

208 0292 06 HIST ECONOMICA GENERAL II	B OR 90-2 1401357 2202 1
208 0362 06 ECONOMIA POLITICA II	MB OR 90-2 1401373 2201 1
208 0369 04 CIES II	MB OR 90-2 1401396 2201 1
208 0490 08 MATEMATICAS II	S OR 90-2 1401420 2201 1
208 0500 06 METODOLOGIA DE LAS C SOCIAL	MD OR 90-2 1401428 2201 1
208 0755 06 SOCIOLOGIA Y POLITICA I	B OR 90-2 1401434 2201 1

TERCER SEMESTRE

208 0293 06 HIST ECON DE AMERICA LATINA	B OR 91-1 1457316 1301 1
---	--------------------------

Back Forward Home Reload Open Save As Clone New Window Close Window

La gran ventaja de este sistema (gráfico) es que es totalmente intuitivo para el usuario, es decir, mediante el uso de imágenes, íconos e incluso alguna explicación con audio, el usuario por si solo comienza a descubrir todas las bondades del sistema.

Debemos recordar que todo esto fue accesado vía telnet, es decir, para poder acceder al sistema debemos de comunicarnos a tzetzal y entrar a la clave de **sina**c. También que todas las pantallas son hechas con HTML.

Como último punto se presentan las pantallas HTML de ingreso al sistema y el documento llave de la Facultad de Arquitectura.

```
<title>Ingreso al sistema</title>
<h1>Bienvenido a SINAC</h1>
<h2>Sistema de Informaci&oacute;n Acad&eacute;mica</h2>
Instrucciones: <p>
A continuaci&oacute;n se presentan los escudos de las diferentes
escuelas y facultades a las cuales prestamos servicio.<p>
Presione sobre el escudo para consultar las historias
acad&eacute;micas.
<p>
```

Aqui se define la hiperliga al documento llave acceso.html. Dentro de este subdirectorio se encuentra el archivo .htaccess que es el que controla el acceso.

```
<a
href="http://tzetzal.dcaa.unam.mx/bases/edgar/pruebas/fa/acceso.html">

Facultad de Arquitectura.</a><p>
```

```
<a
href="http://tzetzal.dcaa.unam.mx/bases/edgar/pruebas/fc/acceso.html">

Facultad de Ciencias.</a><p>
```

```
<a
href="http://tzetzal.dcaa.unam.mx/bases/edgar/pruebas/fca/acceso.html"
>

Facultad de Contadur&iacute;a y Administraci&oacute;n.</a><p>
```

```
<a
href="http://tzetzal.dcaa.unam.mx/bases/edgar/pruebas/fcp/acceso.html"
>

Facultad de Ciencias Pol&iacute;ticas.</a><p>
```

```
<a
href="http://tzetzal.dcaa.unam.mx/bases/edgar/pruebas/fd/acceso.html">

```

Facultad de Derecho.<p>

```
<a
href="http://tzetzal.dcaa.unam.mx/bases/edgar/pruebas/ffyl/acceso.html
">
```

```

Facultad de Filosof&iacute;a y Letras.</a><p>
```

```
<a
href="http://tzetzal.dcaa.unam.mx/bases/edgar/pruebas/fi/acceso.html">

Facultad de Ingenier&iacute;a.</a><p>
```

```
<a
href="http://tzetzal.dcaa.unam.mx/bases/edgar/pruebas/fm/acceso.html">

Facultad de Medicina.</a><p>
```

```
<a
href="http://tzetzal.dcaa.unam.mx/bases/edgar/pruebas/fq/acceso.html">

Facultad de Qu&iacute;mica.</a><p>
```

Este es el documento llave:

```
<title>Acceso a la Facultad de Arquitectura</title>
```

```

```

```
<h1>Bienvenido usuario de la Facultad de Arquitectura !</h1>
```

Su clave y su password han sido validados dentro del sistema.<p>

Cualquier mal uso del sistema será responsabilidad del dueño de la clave.<p>

Cualquier duda o comentario del sistema puede ser enviado por correo electrónico a:<p>

```
<i>dasu@tzetzal.dcaa.unam.mx</i>
```

```
<p>
```

Aquí es donde se da la liga hacia WAIS, como search-engine:

```
<h2>
```

Historias del semestre actual.

```
<a href="wais://tzetzal.dcaa.unam.mx:1500/Sem_actual">
```

```

```

```
</a>
```

```
</h2>
```

6.

Perspectivas de desarrollo.

Las perspectivas de desarrollo de SINAC presentadas no sólo se refieren al sistema en sí, sino al conjunto de herramientas utilizadas para su desarrollo. Las aplicaciones resultantes son muy diversas y dan solución a toda una gama de problemas entre los cuales se encuentra la automatización de la Administración Escolar.

Las herramientas utilizadas en SINAC, como son Mosaic, Gopher y Wais además de ser un estándar dentro de una amplia comunidad como lo es la Internet, presentan características que las hacen únicas para la consulta a través de la red de grandes volúmenes de información, en este caso las historias académicas. Podemos partir de esta premisa para dar algunas ideas para la implementación de nuevos sistemas basados en estas herramientas, como pueden ser:

- Kioscos de información para la comunidad Universitaria, los cuales proporcionarán datos relacionados con las actividades dentro de cada facultad o escuela así como los servicios que éstas prestan.
- Un directorio telefónico de cada dependencia de la DGSCA que incluya además de la información que normalmente proporciona este servicio, la foto de la persona, así como hiperligas a documentos en donde se muestre las actividades que esta desarrolla.

- Un sistema de información de las actividades que desarrolla DGSCA apoyada por videos, sonidos e hiperligas a documentos particulares de cada dependencia, además de los cursos que se imparten en ellas, siendo así un sistema integral, ya que cada dependencia se puede hacer cargo de su parte y a través de Mosaic realizar la fusión de éstas.
- Una guía de carreras en donde el usuario irá obteniendo desde una visión general hasta una particular de las carreras que son de su interés, todo esto apoyado en multimedios, gracias a Mosaic y Wais.
- Implementación de un sistema de apoyo a las actividades académicas al hacer accesible a través de la red documentos como podrían ser capítulos de libros, artículos de revistas, publicaciones varias que sirvan como complemento a lo expuesto en clase por los profesores.

La ventaja de utilizar estas herramientas es que atrás de ellas se encuentra un gran equipo de trabajo formado por los miembros de la comunidad de Internet que engloban a las mejores Universidades a nivel mundial y por ende a los más prestigiados desarrolladores de software. Al ser un software de dominio público y ser ampliamente utilizado por todos ellos, implica que exista una realimentación constante en cuanto a mejoras y nuevas ideas. Además de ser adaptable a las necesidades de la aplicación en particular ya que se puede contar con los programas fuentes para su modificación.

Lo anteriormente expuesto tiene la única finalidad de presentar sólo algunas ideas en las que nosotros vemos un posible desarrollo (aunque algunas de ellas, al término de esta tesis, estén en plena fase de implementación), más no es el

objetivo dar una descripción a fondo de cada una de ellas, ya que de cada una puede generar temas de tesis. También cabe destacar que no son las únicas y principales sino las que presentan un desarrollo a corto plazo.

Gopher y Wais son herramientas que están siendo implementadas con gran éxito y eficiencia en la Universidad a través de servicios como la consulta del herbario nacional, síntesis de artículos de algunos de los principales diarios del país, consulta de actividades culturales dentro de la Universidad, entre otros. Este servicio fue el primero en su tipo en desarrollarse dentro de la Universidad y constituyó la base para generar nuevas ideas sobre la utilización de la infraestructura de RedUNAM. Siguiendo esta misma línea pensamos en implementar nuevos servicios no sólo con las herramientas que ya se dominan, sino fomentando el uso de nuevas y poco explotadas dentro de la Universidad como es el caso de Mosaic.

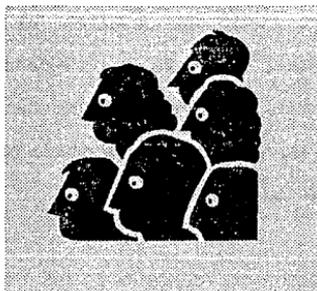
En el caso concreto de SINAC, podemos decir que las perspectivas de desarrollo son muy amplias. Este sistema entrará a su etapa de producción a partir del semestre 95-I. En una primera instancia tendrá un modelo centralizado de almacenamiento de información, con una consulta distribuida. Dicho almacenamiento se concentrará en la DCAA, específicamente en el equipo SUN.

La siguiente etapa contempla la instalación de un *jukebox* de CD-ROM's, gracias al cual se podrá tener acceso a la información de diferentes semestres sin intervención de un operador. Este equipo permite tener montados un número mayor de CD's y accederlos de una forma independiente para su consulta.

Se tiene prevista la compra de uno que cuente con 10 platos para CD-ROM's y una unidad lectora, de esta forma se pueden consultar historias académicas correspondientes a 5 años incluyendo el más reciente.

La última etapa del sistema será la descentralización del almacenamiento masivo, ya que cada dependencia usuaria contará con un equipo capaz de consultar la información que se encuentra en un CD-ROM, además que el sistema de acceso estará instalado de manera local. A cargo de la DCAA sólo quedará la producción de los CD-ROM's correspondientes a cada facultad y escuela, quedando DGAE como responsable de la distribución. Un punto que cabe resaltar es que **la responsabilidad del uso de la información quedará a cargo de cada escuela o facultad.**

En conclusión podemos decir que SINAC es uno de los sistemas que realmente constituye una solución automatizada para la administración escolar a corto y largo plazo, reemplazando así los viejos esquemas que se venían siguiendo desde hace varios años.



Conclusiones.

La Universidad ha buscado como parte de sus objetivos el mantenerse a la vanguardia de conocimientos y nuevas tecnologías. Este objetivo no se encuentra limitado al aspecto de la investigación y lo académico sino que busca darle una aplicación práctica a estos conocimientos dentro de los requerimientos de la Institución.

Los procesos administrativos de la UNAM consumen una gran cantidad de tiempo y recursos en llevarse a cabo, y debido a la importancia que tienen para el funcionamiento de la institución se ha buscado integrarlos a un proceso de modernización. Como parte de una dependencia que apoya a la administración escolar se buscó incorporar uno de sus procesos a los nuevos esquemas de automatización y modernización de procedimientos.

Uno de estos procesos, el almacenamiento de historias académicas, cumplía con todos los requisitos para ser implementado con una de las tecnologías más

recientes de almacenamiento de información: la tecnología de **CD-R**. Además del bajo costo que representa cada unidad, y de la existencia de un estándar bien establecido que asegura la compatibilidad con el hardware y software existente, la grabación de la información en este medio permite su consulta desde cualquier tipo de equipo de cómputo como lo sería una computadora personal.

Se eligió esta solución con base en el análisis realizado de las diferentes alternativas con las que se contaba. Los puntos que se consideraron para su elección fueron la tendencia existente en el mercado mundial actual, el cual se dirige hacia la utilización de esta tecnología debido a la existencia de un estándar en su fabricación y formato de grabación que asegura su compatibilidad con el hardware existente para los CD-ROM. Además en los últimos años se ha estado presentando una disminución del costo de las unidades de CD que ha venido a incrementar de una manera importante la plataforma instalada. Esto nos asegura que no se requiere de instalaciones especiales y costos elevados para la lectura de la información.

Este sistema no sólo involucró la utilización de la tecnología de CD-R sino que se implementó todo un sistema para realizar de una manera eficiente la búsqueda y recuperación de información. También en esta parte del proyecto se presentaron varias alternativas de software, las cuales abarcaron desde soluciones comerciales hasta software de dominio público. Entre las soluciones comerciales se encontraba el manejador de bases de datos SYBASE, que a pesar de ser utilizado actualmente en la UNAM en otros procesos administrativos no era una herramienta adaptable a este problema. Es por ello que se eligió un software de dominio público que además de ser gratuito y tener disponibles sus programas fuentes, constituye un estándar ampliamente difundido en la comunidad Internet.

Durante el desarrollo del presente trabajo adquirimos una visión más amplia del alcance de la tecnología y herramientas de software empleadas en éste.

Dado que todo lo que se utilizó está diseñado para aplicaciones en general y para ninguna en particular, este sistema es posible implementarlo como solución en gran cantidad de problemas actuales tanto de la Universidad como de cualquier institución pública o privada sin fines de lucro, esto último por las condiciones que presentan los dueños de los derechos de autor del software público utilizado (WAIS, Gopher y Mosaic).

Ahora bien existe ya la aplicación de varias de estas herramientas como son el servicio de Gopher implementado en la máquina `condor.dgsca.unam.mx` que se encuentra en la Dirección de General Servicios de Cómputo Académico de la UNAM, o como los CD's con el catálogo bibliográfico de la UNAM, etc. Pero nuestro trabajo será una buena experiencia para poder mejorar estos servicios generando nuevas ideas y haciéndolos más completos.

Por otra parte es posible implementar sólo alguna de las partes que conforman al sistema de acuerdo al problema planteado, por ejemplo, se podría requerir sólo la fase de indexamiento de la información (por medio de WAIS) y la presentación ya sea en tipo texto (por medio de Gopher) o tipo gráfico (por medio de Mosaic) sin grabar la información en CD si así se necesita; o al contrario tal vez sólo se necesite su respaldo en un medio masivo de almacenamiento como lo es el CD-ROM.

En cuanto a lo que se refiere específicamente a nuestro sistema las posibilidades de desarrollo ya se están llevando a cabo en estos momentos ya

que se requiere que los servicios de Gopher y de Mosaic sean más aprovechados teniendo acceso a información que se encuentre en otras dependencias universitarias o mundiales. En primera instancia se está ya en la planeación para incorporar una guía sobre todos los servicios de cómputo que presta la UNAM, así como un auxilio para todo lo referente a los trámites del alumnado de primer ingreso; y a mediano plazo un sistema integral de información académica de la UNAM que incluya desde horarios de los alumnos, reposición de historias académicas, de comprobantes de inscripción, información sobre planes de estudio, adquisición de manuales y notas, etc.

Al finalizar el proyecto y el presente trabajo de tesis pudimos darnos cuenta de varios aspectos que consideramos importantes mencionarlos.

El primero de ellos es la realización de un trabajo de tesis que no sólo signifique la terminación de nuestros estudios profesionales, sino que también beneficie de alguna forma a la Universidad dando un enfoque característico de nuestra profesión: la aplicación de conocimientos para la solución de un problema real que beneficie al mayor número de gente posible, al menor costo y de la forma más eficiente.

El segundo es la aplicación de una manera práctica de los conocimientos adquiridos a lo largo de nuestra carrera. Si bien el plan de estudios de ésta es ambicioso y en algunos puntos completo, no siempre es posible llevarlo a cabo por completo, además de que en muchos casos no existe una relación estrecha entre lo aprendido en el aula y la solución de problemas reales. Aunque cabe resaltar que fue precisamente el esfuerzo de la Universidad en cambiar ésto lo que nos llevó a formar parte de uno de sus planes de becas que involucra al

alumnado con la solución de problemas reales dentro de la Universidad. Estos planes dan una visión todavía mayor de los alcances de nuestra carrera y permiten vincular los conocimientos teóricos con las aplicaciones reales; en particular agradecemos al plan de becas de la Dirección de Cómputo para la Administración Académica por los conocimientos adquiridos y por la oportunidad de desarrollarnos profesionalmente.

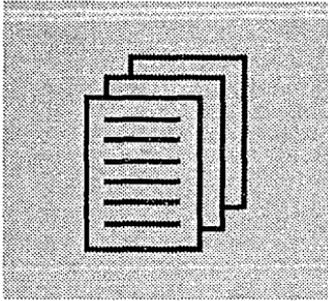
Tercero, conocer algunas de las tecnologías de punta que actualmente se están definiendo como estándares en el almacenamiento de la información, como pueden ser el CD-ROM; de manejo de la misma (como el proyecto WWW y programas como Gopher y Wais) e implementación de sistemas bajo ambiente UNIX, los cuales pueden funcionar de manera eficiente como un modelo cliente/servidor.

Cabe aclarar que para la realización de este trabajo se hizo una extensa investigación de nuevas tecnologías, modelos de sistemas, etc. que pudieran resolver de manera eficiente y rápida el problema planteado; esto implicó la consulta de mucha bibliografía: libros, revistas y manuales; pláticas con personal de DGSCA para tener puntos de vista diferentes del problema; sin embargo una de las fuentes de mayor información fue la red Internet.

Podemos decir que la mayoría de la bibliografía proviene de ella, debido a que la tecnología empleada es muy reciente como para buscar referencias bibliográficas. Debido a ello consideramos la importancia de darle una mayor difusión a los alumnos de nuestra carrera para que puedan aprovecharla al máximo.

Conclusiones

Para finalizar diremos que la realización de este trabajo representó para nosotros una gran experiencia y satisfacción debido a que se encuentra ya en funcionamiento y que ayudó a solucionar un problema de nuestra *alma mater*, la **Universidad Nacional Autónoma de México**.



Apéndice I

Make y Makefiles.

Make y Makefiles.

El objetivo de este Apéndice es el de dar una breve introducción al uso de la utilería **make** así como una revisión breve de los archivos **Makefile** utilizados en la compilación de WAIS y GOPHER, herramientas utilizadas en SINAC.

Make ejecuta una serie de comandos asociados a un **objetivo**, es decir, con cada programa que se quiere hacer ejecutable. El **Makefile** es un archivo que contiene la forma en que el **objetivo** tiene que ser logrado con respecto a aquellos archivos de los cuales depende (generalmente programas en C) a los cuales denominamos **dependencias**. A su vez cada **dependencia** es un **objetivo** por lo que este también tendrá **dependencias** y así sucesivamente. Así, los **objetivos**, **dependencias** y **sub-dependencias** forman un **árbol** en el que **make** se mueve decidiendo que **objetivo** tiene que ser reconstruido (recompilado) y cual no.

Para poder construir un **objetivo**, **make** ejecuta una serie de comandos, llamados **reglas**. Estas **reglas** pueden ser listadas explícitamente dentro del **Makefile**. Si un archivo **Makefile** no es especificado con la opción **-f** del **make** este verifica si existe un archivo llamado **Makefile**, si este existe usa las **reglas** contenidas en él. Si tampoco se le especifica un **objetivo**, **make** utiliza el primer objetivo definido en el **Makefile**.

Make supone que sólo se harán cambios a los archivos que se están procesando durante la corrida del mismo. Si un archivo fuente es cambiado durante la corrida del **make**, el fuente resultante puede tener algún tipo de inconsistencia.

El formato básico del *objetivo* se muestra a continuación:

```
objetivo1 [objetivo2 ...] : [dependencia]
[ comando ]
```

En la primera línea, la lista de los nombres de los *objetivos* (separados por espacio en blanco) termina con dos puntos (:). A continuación viene una lista de dependencias, las cuales pueden ser opcionales. Si se tienen muchos *objetivos* listados cada uno de ellos será tratado independientemente usando las reglas que se tengan.

Las siguientes líneas son *comandos* y éstos deben de empezar con un caracter TAB, un error muy común es emplear espacios en vez del caracter TAB. Para incluir comentarios de debe de poner el caracter # y éste terminará hasta que se encuentre un caracter CR (carriage return).

Para ejemplificar su uso veamos un ejemplo del contenido de un archivo

Makefile:

```
prueba:
    ls prueba
    touch test
```

Cuando se ejecuta **make** en la línea de comandos, sin argumentos, busca un archivo llamado *Makefile*. La salida será la siguiente:

```
tzetzal:1> make
ls prueba
prueba not found
```

```
touch prueba
tzetzal:2> ls prueba
prueba
```

Debido a que el archivo `prueba` no estaba presente se ejecuta la *regla* que se tiene descrita, en este caso `touch1 prueba`. Si corremos ahora `make` nos dira lo siguiente:

```
tzetzal:3> make
'prueba' is up to date.
```

Es decir, nuestro archivo ya fue actualizado.

Los cambios de linea dentro de una regla son significativos para cada comando que se ejecute en esa regla. Esto quiere decir que si se tiene:

```
prueba:
    cd /tmp
    pwd
```

se comportará de una forma que no se espera, en este caso no hará el cambio al directorio `/tmp` y solo ejecutará el `pwd`. Para evitar esto, nos valemos de un `;` para especificar una secuencia de comandos a ejecutar:

```
prueba:
    cd /tmp ; pwd
```

¹ Touch es un comando de Unix que permite crear un nuevo archivo con extensión de cero bytes.

Podemos poner un backslash (\) para indicar que la línea continua en el renglón de abajo:

prueba:

```
cd /tmp ; \  
pwd
```

Una vez que `make` comienza a procesar los *objetivos* busca las dependencias asociadas a ellos. Por ejemplo:

```
batch: a b  
touch batch  
b:  
touch b  
a:  
touch a  
c:  
echo "Nunca veras este mensaje."
```

`make` comienza con el *objetivo* `batch`, como este tiene dependencias que no han sido verificadas difiere a `batch` hasta que todas estas dependencias hayan sido procesadas. El proceso es el siguiente: se tiene a `batch` pero como este tiene dependencias (llamadas a y b) `make` sigue con la dependencia a. Debido a que esta no tienen *subdependencias* ejecuta `touch a` y regresa a `batch`, como este tiene una segunda dependencia (b) ejecuta `touch b`. Finalmente, como no se tienen más dependencias ejecuta `touch batch` finalizando con esta última acción. También es válido tener dependencias con una regla nula, esto con el

propósito de ejecutar siempre la regla especificada, comúnmente a esto se le denomina "forzar una dependencia" (FORCE).

```
basura: FORCE
    echo "Siempre me verás"
FORCE:
```

Al ejecutar:

```
tzetzal:1> touch basura
tzetzal:2> make basura
echo "Siempre me verás"
Siempre me verás
```

Se debe de notar que al `make` podemos pasarle como parámetro un *objetivo* el cual queremos que se cumpla, en el caso anterior "basura".

Si alguno de los objetivos no se encuentra dentro del *Makefile*, el `make` marcará un error: `make: Fatal error: Don't know how to make target 'xxxx'`. En este caso `xxxx` es el nombre del objetivo llamado.

La utilería `make` nos permite definir **macros** y poder sustituir estas en cualquier parte de nuestro archivo *Makefile*. Por ejemplo si queremos compilar un programa con la opción `-ansi` del `gcc`, podemos hacer una referencia a una macro dentro de nuestro *Makefile* de la siguiente forma:

```
funcion: funcion.c
    gcc $(CFLAGS) -o funcion funcion.c
```

Entonces la referencia a la macro CFLAGS será sustituida por un valor, el que nosotros le indiquemos:

```
tzetzal:1> rm funcion
tzetzal:2> make funcion "CFLAGS= -ansi"
gcc -ansi -o funcion funcion.c
```

Estas definiciones también las podemos hacer dentro de nuestro *Makefile* de la siguiente manera:

```
CFLAGS= -ansi
funcion: funcion.c
    gcc $(CFLAGS) -o funcion funcion.c
```

Una vez que tenemos las ideas básicas del funcionamiento del make podemos utilizarlo para compilar nuestros programas, a manera de ejemplo se tiene la compilación del siguiente programa que tiene sus fuentes en dos archivos diferentes:

```
# Ejemplo de la utilización del make
funcion: main.o data.o
    gcc -O funcion main.o data.o
main.o: main.c
    gcc -O -c main.c
```

```

data.o: data.c
    gcc -O -c data.c
clean:
    rm funcion main.o data.o

```

Con este pequeño ejemplo podemos ver lo fácil que es escribir un Makefile teniendo en cuenta cual es el objetivo y sus dependencias y reglas asociadas.

Además se tienen algunos caracteres que tienen un significado especial para make, como lo son:

- \$@

El nombre del *objetivo* actual.

- \$?

Lista de *dependencias*.

- \$*

El nombre base del *objetivo* actual (sin sufijo).

MAKEFILE de la herramienta WAIS.

En este punto se da información de todas las revisiones hechas al programa:

```

#
# Top level make of the WAIS system
#   brewster 2/91
#   jonathan 6/91
# $Log: Makefile-release,v $
# Revision 1.8  92/05/07  15:54:08  jonathan
# Updated for release.

```

Make y Makefiles.

```
#
# Revision 1.7  92/03/26  18:29:59  jonathan
# Fixed some broken make lines.
#
# Revision 1.6  92/03/07  19:45:01  jonathan
# Added recommendation for IBM.
#
# Revision 1.5  92/02/27  10:07:24  jonathan
# got rid of automatic setting of TOP.  Used Simon's
# approach
# instead.
# Revision 1.4  92/02/13  12:27:53  jonathan
# Removed references to seeker.
#
# Revision 1.3  92/02/13  12:05:17  jonathan
# Removed release targets.
#
# Revision 1.2  92/02/13  11:57:56  jonathan
# Added $Log for RCS.
```

A continuación se dan todas las definiciones de macros que son necesarios para la compilación y ligado:

```
# common customizations:
# see the CFLAGS variable for pointers.
# to do:
# create the scripts, install the pointer to this version
# if it is the newest.
#
# SGIs want this uncommented
# SHELL=/bin/sh

RELEASE = freeWAIS-0.3
```

Aquí se define RM, una macro que contiene el comando rm de UNIX que nos permite borrar archivos, la opción -f indica que al borrar archivos no despliegue ninguna información.

```
RM = /bin/rm -f

# on SGIs set this to true
# RANLIB = true
```

RANLIB es una macro que define el lugar en donde se encuentra el programa ranlib que efectua actualización de bibliotecas.

```
RANLIB = /home/gopher/modif/ranlib
# on IBM RS6000 this should be c89.
# CC = cc
```

Aquí definimos nuestro compilador de "C", en este caso gcc:

```
CC = gcc

# set this for your site. This syntax only works in SunOS
# for other UNIX-like OS's set this to this directory.
# TOP:sh = pwd
# or fill in the blank for other OS's comment-me:
# @echo You must set "\${TOP}" to point to the freeWAIS
# src directory
TOP:sh = pwd
SUPDIR = ${TOP}/ir
```

A continuación se presentan las opciones para el compilador dependiento de la arquitectura del equipo y el sistema operativo.

```
# compiler specific stuff
#
# for old BSD add -DBSD
# for newer BSD that needs to use <sys/dir.h>, add -DBSD43
# for System V add -DSYSV
# for XENIX add -M3e -Zi
# USG for Unix Diredt in lib
# for SGIs running IRIX 4.0.1, add -cckr
# for NeXTSTEP add -DNEXT and -posix
# for Linux add -DLINUX
#
# For a little better security in the server, add -
#DSECURE_SERVER
# this sets the server user id to -u argument after
# startup.
# for relevance feedback in the search engine, add
# -DRELEVANCE_FEEDBACK
#
# dgg additions
# LITERAL == waissserver, search for "literal strings"
# BOOLEANS == waissserver, search with boolean AND, NOT
```

```

#operators
# PARTIALWORD == waissserver, search for partial words, hum*
# matches human, hummingbird, ...
# BIO == waisindex, waissserver changes including symbol
# indexing & search & bio data formats
# -DTELL_USER lets the server know who you are at connect
# time
# -DUSE_SYSLOG if you want logging to be done with syslog
# rather than fprintf
# -DNEED_VSYSLOG if your C library does not have a
# vsyslog()function
#   in it (and you defined USE_SYSLOG)
# -DDUMPCORE will force the waissserver to dump the core
# when aborting otherwise the core will not be dumped
# -DEND_MERGE if you want to merge the index files at the
# end of an index process otherwise they are merged as we go
# along
# -DSTEM_WORDS to stem words during indexing and queries
#
# Note - the default Porter Stemmer removes trailing e's
# from words - variable becomes variabl - this can impact
# the use of literals in searches!!!!!!!!!!!!
#
# -DLIST_STEMS to show stemmed words in server log and
# indexer output
# -DSOLARIS for SunOS 5.2 (Solaris 2.2) machines.
#
# Use this version of CFLAGS for SGIs with gcc
# CFLAGS = -I$(SUPDIR) -DTELL_USER -DSECURE_SERVER
# -DRELEVANCE_FEEDBACK -DUSG -DBOOLEANS -DPARTIALWORD -
# DLITERAL
# -DSOUND -DBIBDB
#
# Use this version of CFLAGS for DECstation with gcc
# CFLAGS = -ansi -I$(SUPDIR) -DTELL_USER -DUSG -
# DSECURE_SERVER
# -DRELEVANCE_FEEDBACK -DBOOLEANS -DPARTIALWORD -DLITERAL -
# DSOUND
# -DBIBDB -DULTRIX
#
# Use this version of CFLAGS for DECstation with cc
# CFLAGS = -I$(SUPDIR) -DTELL_USER -DUSG -DSECURE_SERVER
# -DRELEVANCE_FEEDBACK -DBOOLEANS -DPARTIALWORD -DLITERAL -
# DSOUND
# -DBIBDB -DULTRIX
#
# Use this version of CFLAGS for DEC Alpha with gcc
# CFLAGS = -ansi -I$(SUPDIR) -DTELL_USER -DUSG -
# DSECURE_SERVER
# -DRELEVANCE_FEEDBACK -DBOOLEANS -DPARTIALWORD -DLITERAL -

```

```

# DSOUND
# -DBIBDB
#
# Use this version of CFLAGS on Sun with gcc (not Solaris)
# CFLAGS = -ansi -I$(SUPDIR) -DTELL_USER -DUSG -
# DSECURE_SERVER
# -DRELEVANCE_FEEDBACK -DBOOLEANS -DPARTIALWORD -DLITERAL -
# DSOUND
# -DBIBDB
#
# Use this version of CFLAGS for Linux with gcc
# CFLAGS = -O2 -ansi -m486 -fwritable-strings -I$(SUPDIR)
# -DTELL_USER -DUSG -DSECURE_SERVER -DRELEVANCE_FEEDBACK
# -DBOOLEANS -DPARTIALWORD -DLITERAL -DSOUND -DBIBDB -
# DLINUX

```

Una vez vistas todas las opciones posibles para la compilación y las que se requieren para nuestro sistema (Solaris), las definimos a continuación:

```

CFLAGS = -ansi -lsocket -lnsl -lgen -I$(SUPDIR) -DTELL_USER -
DUSG -DSECURE_SERVER -DRELEVANCE_FEEDBACK -DBOOLEANS
-DPARTIALWORD -DLITERAL -DSOUND -DBIBDB -DSYSV -DSOLARIS
-mfpu -mno-epilogue

# Solaris should use
#LIB = -ltcpip -lsocket -lnsl
# SGIs should use this
#LIB= -lmalloc
#LIB = -ltcpip -lsocket -lnsl

# There are different versions of curses which could be
# used, so
# set the
# library here
# CURSESLIB = -lcurses
# Solaris machines don't use -k
# MFLAGS = -k

```

Se define otra macro llamada MAKE, debido a que dentro de cada subdirectorio de WAIS existen otros *Makefile* específicos para los programas que se encuentran en ellos.

```
MAKE = make $(MFLAGS)
```

```
#default: config.h lib ir ui bin doc x tags
default: config.h lib ir ui bin x check
    @echo "Welcome to WAIS"
```

En este caso, el objetivo config.h se vale de un programa llamado config que se encuentra en el directorio en uso (.). La salida de este se almacena mediante un direccionamiento hacia en archivo config.h.

```
config.h: config
    ./config >config.h

config: config.c
    $(CC) $(CFLAGS) -o config config.c

lib::
    cd lib;
    $(MAKE) CC=$(CC) CFLAGS="$(CFLAGS)" RANLIB=$(RANLIB)
TOP=$(TOP)

ir:
    cd ir;
    $(MAKE) CC=$(CC) CFLAGS="$(CFLAGS)" RANLIB=$(RANLIB)
TOP=$(TOP)

waisindex:
    cd ir;
    $(MAKE) waisindex CC=$(CC) CFLAGS="$(CFLAGS)"
RANLIB=$(RANLIB)\ TOP=$(TOP)

waisserver:
    cd ir;
    $(MAKE) waisserver CC=$(CC)
CFLAGS="$(CFLAGS)" RANLIB=$(RANLIB)\
    TOP=$(TOP)

ui:
    cd ui;
    $(MAKE) CC=$(CC) CFLAGS="$(CFLAGS)" TOP=$(TOP) \
    CURSES_LIB=$(CURSES_LIB)

# NeXT and ULTRIX don't have an env command, so this
# doesn't work.
# try going to the x directory and just doing a make -k
# you may have to edit the CFLAGS in the Makefile yourself.

# Trata de ejecutar un programa para ambiente gráfico, como
```

```
# no se necesita se eliminó para hacer la compilación más
# rápida.
# cd x; xmkmf; make depend; make

# x:
# cd x; make depend; make
# cd x; (env TOP=$(TOP) CC=$(CC) CFLAGS="$(CFLAGS)"
# MAKE="$(MAKE)" ./makex.sh)

bin:
    cd bin; $(MAKE) CC=$(CC) CFLAGS="$(CFLAGS)" TOP=$(TOP)

test:
    @echo $(MAKE) CC=$(CC) CFLAGS="$(CFLAGS)" TOP=$(TOP)

check:
    cd wais-test; $(MAKE)

# Remove objects and library.
clean:
    $(RM) *%
    $(RM) *~
    $(RM) \#*\#
    $(RM) core
    $(RM) TAGS
    $(RM) -r SearchLog
    $(RM) wais-sources/wais-docs*
    $(RM) config config.h
    cd lib; make $@
    cd ir; make $@
    cd ui; make $@
    cd bin; make $@
    cd wais-test; make $@
    cd x; make $@

rlocks:
    $@
    cd lib; $@
    cd ir; $@
    cd ui; $@
    cd bin; $@
    cd doc; $@
    cd x; $@
```

Una vez analizado el caso de WAIS, veremos a continuación el de GOPHER y su Makefile.

El archivo que se muestra a continuación se llama *Makefile.conf* y su objetivo es el de configurar todas las opciones de librerías, includes y parámetros del compilador. Esta es una buena medida para evitar modificaciones al archivo *Makefile* que es el que make busca.

```
#-----
# This is where most of the configuration parameters are
# defined also see conf.h
#
#*****#
lindner
# 3.12
# 1994/03/08 15:54:06
#
#/home/mudhoney/GopherSrc/CVS/gopher+/Makefile.config.dist
#,v
# Exp
#
# Paul Lindner, University of Minnesota CIS.
#
# Copyright 1991, 1992 by the Regents of the University of
# Minnesota
# see the file "Copyright" in the distribution for
# conditions of
# use.
#*****#
# MODULE: Makefile.config.dist
# Parameters for the gopher distribution
#*****#

#-----
# Your favorite C compiler
#
# Note that sun international users should use /usr/5bin/cc
# instead of cc
#
# Sco's cc compiler gives lots of problems that gcc will
# fix, and gcc is now reasonably easy to get running under
#SCO. Using this removes the need for -UM_XENIX -DSCO_UNIX as
#used for gopher1.1 note that if you use gcc, you'll also
#need -lintl in SCOLIBS
```

```

#
CC = gcc

#-----
# Optimization level.
#
OPT=-g
#OPT=-O

#-----
# System Selection, note that you won't have to edit
# unless you have compilation problems.
#
# Add -DUSG          for System V
#   -DBSD           for BSD
#   -DNO_WAITPID    if you have wait3 instead of waitpid()
#   -DUSE_FLOCK     if you have flock instead of fcntl()
# locking
GSYSTYPE=
#-----
# The ranlib command on your system. A/UX (and probably
# other sysv's should change this to "touch"
# Known to need changing on: A/UX, SCO3.2.4, IRIX

RANLIB = /home/gopher/modif/ranlib

#-----
# The install command on your system. OSF/1 should change
# this to install bsd
#
# SCO ODT systems should change this to bsdinst
# other SCO systems can duplicate this functionality easily
# dont use SCO's "install" program it behaves totally
# differently IBM AIX systems should change this to
#/usr/ucb/install

INSTALL = install -c

#-----
# Where shall we install stuff?
#
PREFIX          = /usr/local
CLIENTDIR      = $(PREFIX)/bin
CLIENTLIB      = $(PREFIX)/lib
SERVERDIR      = $(PREFIX)/etc
# On SCO manuals are in /usr/man but its easiest to do a
# symbolic link from /usr/local/man to /usr/man for this
# and other packages
MAN1DIR       = $(PREFIX)/man/man1

```

Make y Makefiles.

MAN5DIR = \$(PREFIX)/man/man5
MAN8DIR = \$(PREFIX)/man/man8

```
#-----  
# DEBUGGING control...  
#  
# Comment this to make a slimmer executable...  
  
DEBUGGING = -DDEBUGGING  
  
#-----#  
Optional server features.  
#  
# Add -DADD_DATE_AND_TIME to add dates and times to the  
# gopher titles  
#  
# Add -DLOADRESTRICT if you want to restrict access based  
# on load avg.  
# (Note you'll need to add -lkvm in SERVERLIBS) Note  
# also that this has only been tested under SunOS 4.1.1  
# Add -DBIO if you're using the biology portion of Don  
# Gilbert's modified  
# wais8b5 that supports boolean and phrase searching.  
# PLEASE NOTE: the -DBIO option is NOT needed nor  
# recommended for use of the boolean and phrase searching  
# portion of this modified wais, just the symbol searching.  
# But, you must compile wais and gopher with the same  
# option setting (-DBIO or NOT).  
# The source can be gotten from:  
# ftp.bio.indiana.edu  
#  
# Add -DDL and define DLPATH and DLOBS if you want to  
# include support for Tim Cook's 'dl' databases You will also  
# have to have a working copy the program with source code in  
# the DLPATH directory. The files getdesc.o and enddesc.o  
# must be there.  
# Source for dl can be gotten from:  
# admin.viccol.edu.au  
# pub/dl/describe-1.8.tar.Z or higher...  
#  
# Add -DUMNDES if you'd like to try out the Admit1 protocol  
# extension  
#  
# Add -DCAPFILES if you want compatibility with the older  
# .cap directory  
#  
# Add -DSETPROCTITLE if you want to set the process title  
# displayed by the 'ps' command (only works on bsdish  
# systems...)  
  
SERVEROPTS = -DSETPROCTITLE -DCAPFILES
```

```

#-DUMNDES -DBIO -DDL -DLOADRESTRICT
#DLPATH = /home/mudhoney/lindner/src/describe
#DLOBS = $(DLPATH)/getdesc.o $(DLPATH)/enddesc.o

#-----
# Optional client features.
#
# Add -DNOMAIL if you don't want remote users mailing
# documents
#
# Add -DAUTOEXITONU if you want to treat q and u as the
# same, and automatically exit from the top menu - usefull if
# Gopher called from another app.

CLIENTOPTS = #-DNOMAIL -DREMOTEUSER -DCLIENT_LOGGER

#-----
# Libraries for clients and servers
#   Ultrix needs -lcursesX instead of -lcurses
#
#-----
# Libraries... Uncomment out SEQLIBS if compiling on
# sequent Dynix,
#          "          " PTXLIBS if compiling on
# sequent Dynix/ptx,
#          "          " UMAXLIBS if compiling under
# UMAX,
#          "          " SCOLIBS if compiling under
# SCO Unix.
#          "          " AUXLIBS if compiling under
# A/UX
#          "          " INTERACTIVELIBS if compiling
# under Interactive
#
# Note: SCOLIBS needs -lintl if using gcc to compile in
# order to find strftime
#
#UMAXLIBS = -lresolv
#LOADLIBS = -lkvm
#SEQLIBS   = -lseq
#PTXLIBS   = -lseq -lsocket -linet -lnsl
#SCOLIBS   = -lsocket -lintl
SVR4LIBS   = -lsocket -lnsl -lgen
#AUXLIBS   = -lmalloc
#INTERACTIVELIBS= -linet
#DGUXLIBS  = -lnsl

OTHERLIBS = $(UMAXLIBS) $(SEQLIBS) $(PTXLIBS) $(SCOLIBS) \
            $(SVR4LIBS) $(AUXLIBS) $(DGUXLIBS)

```

Make y Makefiles.

```
CLIENTLIBS      = -lcurses -ltermcap -lgopher $(OTHERLIBS)
SERVERLIBS      = -lm -lgopher $(OTHERLIBS) $(LOADLIBS)

#
# Uncomment out this line to use shared libraries on Sun
# systems
#

SHAREDLIB = sun

#-----
# If your hostname command returns the Fully Qualified
# Domain Name (i.e. it looks like foo.bar.edu and not just
# foo) then make the domain name a null string.  Otherwise
# put in the rest of your domain name that `hostname` doesn't
# return.
# Set to Null on SCO3.2.4

DOMAIN          = .dcaa.unam.mx

#-----
# SERVERDIR is the default directory for gopherd.  It can
# be overridden on the command line SERVERPORT is the default
# port for gopherd.  It too can be overridden on the command
# line.
#

SERVERDATA      = /home/gopher/gopherd
SERVERPORT      = 70

#-----
# Compatibility defines
#
# If you don't have the strstr() function call then add
# -DNOSTRSTR
#
# Most of these are automatically defined via the built in
# compiler definitions.  Don't worry about them unless you
# have problems

COMPAT = # -DNOSTRSTR # -DNO_STRDUP # -DNO_BZERO # -
DNO_TMPNAM
          # -DNO_VFORK

#-----
# Stuff that follows shouldn't be changed

OBJINCLUDES    = -I../object
OTHERINCLUDES  = -I.. -I../ir -I../ui
INCLUDES       = $(OBJINCLUDES) $(OTHERINCLUDES)
```

```
LDFLAGS = -L../object
SHELL   = /bin/sh
```

Una vez configurado el *makefile.conf*, veamos el *Makefile* :

```
*****
# lindner
# 3.12
# 1994/01/07 20:52:01
# /home/mudhoney/GopherSrc/CVS/gopher+/Makefile,v
# Exp
#
# Paul Lindner, University of Minnesota CIS.
#
# Copyright 1991, 1992 by the Regents of the University of
# Minnesota see the file "Copyright" in the distribution for
# conditions of use.
*****
# MODULE: Makefile
# Makefile for entire gopher distribution.
*****
```

Aquí se incluye la parte de configuración anterior :

```
include Makefile.config

all: objects server client gopherfilter
    @echo "Welcome to Gopher"

server: gopherd/gopherd
client: gopher/gopher
gopherfilter: gophfilt/gophfilt
objects: object/libgopher.a
```

Esta parte del *Makefile* es crucial, ya que se define el **search engine** a utilizar, en este caso se tiene a WAIS para tal efecto, por lo que se tiene un pequeño *script* (archivo de comandos de Unix) el cual busca todas las posibilidades de search engine que tiene el sistema, para el caso de WAIS se deben de crear tres ligas, es decir, apuntadores a archivos que se encuentran en los directorios de

WAIS, estos son ui, ir, bin. El script busca por "ui" por lo que identifica a WAIS (el script esta escrito en Bourne shell):

```
gopherd/gopherd: object/libgopher.a gopherd/*.c gopherd/*.h \
    conf.h
    @echo "Making server"
    @if [ -f /sdmach -a -d ./ui ]; then \
        (echo "Making server with NeXT and WAIS searching"; \
        cd gopherd; $(MAKE) $(MFLAGS) "SEARCH=-DNEXTSEARCH
        -DWAISSEARCH" nextwais); \
    elif [ -f /sdmach ]; then \
        (echo "Making server with NeXT searching"; \
        cd gopherd; $(MAKE) $(MFLAGS) "SEARCH=-DNEXTSEARCH"
next); \
    elif [ -d ./ui ]; then \
        (echo "Making server with WAIS searching"; \
        cd gopherd; $(MAKE) $(MFLAGS) "SEARCH=-DWAISSEARCH"
wais); \
    else (echo "Making data server"; cd gopherd; $(MAKE)
$(MFLAGS)
none) \
    fi

# gopher is dependant on conf.h - (as are other things)
gopher/gopher: object/libgopher.a conf.h gopher/*c gopher/*h
    @echo "Making client"
    (cd gopher; $(MAKE) $(MFLAGS))

gophfilt/gophfilt: object/libgopher.a conf.h gophfilt/*c
    @echo "Making gophfilt"
    (cd gophfilt; $(MAKE) $(MFLAGS))

object/libgopher.a: object/*c object/*h
    @echo "Making Objects"
    (cd object; $(MAKE) $(MFLAGS))

install:
    (cd object; $(MAKE) $(MFLAGS) install);
    (cd gopher; $(MAKE) $(MFLAGS) install);
    (cd gopherd; $(MAKE) $(MFLAGS) install);
    (cd gophfilt; $(MAKE) $(MFLAGS) install);
    (cd doc; $(MAKE) $(MFLAGS) install)
```

El caracter "-" antes del comando `rm` le indica a `make` que aunque existan errores (como puede ser que el archivo no se encuentre) estos se procesen sin abortar la corrida:

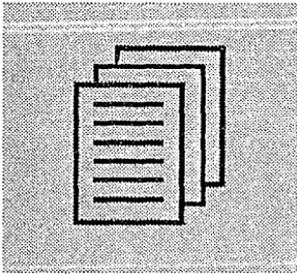
```
clean:
(cd gopher; $(MAKE) $(MFLAGS) clean)
(cd gopherd; $(MAKE) $(MFLAGS) clean)
(cd object; $(MAKE) $(MFLAGS) clean)
(cd gophfilt; $(MAKE) $(MFLAGS) clean)
(cd doc; $(MAKE) $(MFLAGS) clean)
-rm -f "examples/Sample Directory/wais-index/index.*"

spotless: clean
-rm -f *~

archive: spotless
-rm -f ir ui bin MANIFEST
touch MANIFEST
cp Makefile.config.dist Makefile.config
HERE=`basename $$PWD`; \
echo $$HERE ; \
    cd .. ; \
    find $$HERE -type f -print |grep -v '.o$$' |grep -v '-$$'
|grep -v 'CVS' >$$HERE/MANIFEST ; \
/usr/gnu/bin/tar -T $$HERE/MANIFEST -czvf $$HERE.tar.Z ;\
zip -@ $$HERE.zip <$$HERE/MANIFEST
@echo "Did you remember to update the patchlevel.h
Paul?..."
@echo "Hmmm... Hmmm...."

quicktar:
HERE=`basename $$PWD`; \
    cd .. ; \
/usr/gnu/bin/tar -T $$HERE/MANIFEST -czvf $$HERE.tar.Z ;
```

Con esto se termina la parte del `make` y de sus archivos *Makefile* para cada una de las aplicaciones correspondientes.



Apéndice II

Manual del usuario

SINAC

Sistema de Información Académica.

- **Introducción.**

El Sistema de Información Académica (SINAC) tiene como meta facilitar la consulta de las historias académicas de la UNAM a los usuarios que así lo requieran, como es el caso de DGAE (Dirección General de Administración Escolar) y las oficinas de Servicios Escolares de cada Facultad.

Esto lo hace mediante dos interfases para el usuario: una interfaz orientada a caracter mediante la cual cualquier usuario que pueda conectarse a RedUNAM puede tener acceso a ella y una interfaz gráfica, mediante la cual el usuario tiene un ambiente mucho más agradable e intuitivo, ya que mediante el uso de íconos podemos acceder a la información. Para ello, la máquina del usuario además de poderse conectar a RedUNAM, debe contar con capacidades gráficas o poseer Mosaic para Windows, en caso que no se tenga, se puede instalar éste de una forma muy sencilla.

Este sistema fue **totalmente** implementado en la ***Dirección de Cómputo para la Administración Académica*** por el personal y becarios que laboran en la ***Coordinación Técnica.***

- ¿ Cómo se puede acceder el sistema ?

Para poder acceder a nuestro sistema, debemos conectarnos al servidor UNIX de la dependencia el cual es tzetza1 de la siguiente forma:

1) Debemos establecer una sesión telnet, es decir una sesión remota con nuestro servidor UNIX como se muestra a continuación:

```
LANWP:  tnvt220 tzetza1.dcaa.unam.mx      ó
        tnvt220 132.248.27.10
```

```
UNIX:   telnet tzetza1.dcaa.unam.mx      ó
        telnet 132.248.27.10
```

Programa genérico:

```
tn tzetza1.dcaa.unam.mx      ó
tn 132.248.27.10
```

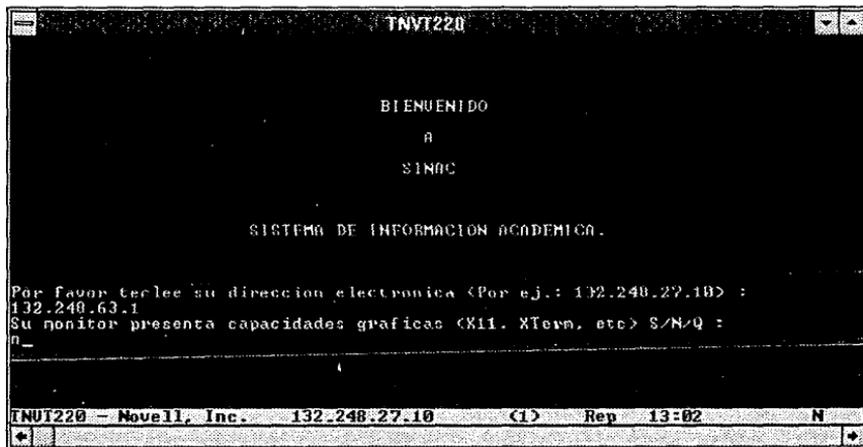
Una vez que el programa se ha conectado a tzetza1, el sistema responderá de la siguiente forma:

```
login: _
```

En este punto el sistema pide una clave de acceso, se debe teclear `sinac` y dar <ENTER>.

```
login: sinac
```

El sistema responde dando un mensaje de bienvenida:



Se pide que teclee la dirección IP de la máquina del usuario tal y como se muestra en el ejemplo, después se pregunta si su monitor presenta capacidades gráficas para el despliegue de la interfaz GUI.

Esta parte del manual solo se enfoca a la parte orientada a caracter CUI. Por lo que se recomienda teclear 'N' cuando el sistema pregunta por capacidades gráficas.

Después de unos segundos se visualiza la siguiente pantalla:



Esta es la primera pantalla de nuestro sistema, muestra un menú donde se observan las diferentes escuelas y facultades, hasta este momento solo contamos con la información de la ENEP Acatlán de la carrera de Economía para propósitos de prueba, sin embargo se piensa incluir la información del semestre anterior para comenzar la producción de CD-ROM's para su consulta centralizada como primera etapa del proyecto y después descentralizada a manera de que cada Escuela y Facultad sea responsable de su información; esta deberá pedirse a DGAE.

Debido a ello seleccionamos la ENEP Acatlán, las demás opciones no presentan información alguna, sin embargo da una idea del sistema cuando este con toda la información.

Al seleccionar esta escuela se verá la siguiente pantalla:

```

Internet Gopher Information Client v2.0.14
Root gopher server: tze2tal

--> 1. ENEP ficatlan <TEL>
      ficatlan

Precaucion!!!!, esta a punto de abandonar el
programa Gopher y conectarse a otro host. Si
llegara a pasarse presione la tecla control y
l. despues presione q.

Conectandose a tze2tal.deca.unam.mx, puerto 21 usando telnet.
Use la cuenta "pg01" para poder entrar.

(Cancelar: ^G) OK: Enter

Presione ? para ayuda, ^ para salir. Pagina: 1/1
INUT220 - Novell, Inc. 132.248.27.10 <1> Rep 13:03

```

El mensaje que se ve indica que se efectuara una reconexión al sistema, ésto con el objeto de implementar un nivel de seguridad de la información, ya que no todos los usuarios deben tener acceso a la misma.

Se debe de observar que en la ventana se presenta información acerca de que clave se debe de usar cuando se efectue la reconexión, en este caso es pg01. Al presionar cualquier tecla se presentará la siguiente pantalla y estas claves se deben de teclear, como se muestra:

```

TNVT220
Trying 132.248.27.10 ...
Connected to tzetzal@caatunam.mx.
Escape character is '^['.

UNIX(r) System U Release 4.0 (tzetzal)

login: pg01
Password: _

TNVT220 - Novell, Inc. 132.248.27.10 <1> Rep 13:07

```

Al teclear correctamente las claves, entramos a la ventana principal de nuestro sistema, en este caso estamos listos para consultar la información.

```

TNVT220
Internet Gopher Information Client v2.0.14

Root gopher server: tzetzal

--> 1. Historias Academicas ENEP Acatlan 94-I <CDROM> <?>
    2. Historias Academicas ENEP Acatlan 94-II <?>

Presione ? para ayuda, ! para salir
TNVT220 - Novell, Inc. 132.248.27.10 <1> Rep 13:07 Pagina: 1/1

```

Como se ve, se tienen dos opciones, las cuales corresponden a las historias académicas de la ENEP Acatlán de la carrera de Economía. La primera opción corresponde a las historias académicas del semestre 94-II, las cuales se encuentran almacenadas en disco y las del semestre 94-I, las cuales se encuentran almacenadas en CD-ROM. Estas dos opciones contienen la misma información y solo se tienen así para propósitos de demostración. Cabe señalar que la información almacenada en CD-ROM se tiene centralizada en el servidor, es decir que el usuario esta accedando la información que se encuentra en éste y no en su estación de trabajo; en un período breve de tiempo, cada Escuela y Facultad tendrá este sistema con lo que la información se tendrá descentralizada y se podrá consultar desde su estación de trabajo.

Para acceder a la información, simplemente damos <ENTER> a la opción que deseemos, al darlo se verá lo siguiente:

```

TINY220
Internet Gopher Information Client v02.0.14
Root gopher server: tnet2al
--> 1. Historias Academicas ENEP Acatlan 94-I <CDROM> <?>
    2. Historias Academicas ENEP Acatlan 94-II <?>

Historias Academicas ENEP Acatlan 94-I <CDROM>
Palabras a buscar
[ ]
Ayuda: ^_I | Cancelar: ^CI

Presione ? para ayuda, q para salir
TINY220 -- Novell, Inc. 132.248.27.10 <1> Rep 13:34 Pagina: 1/1

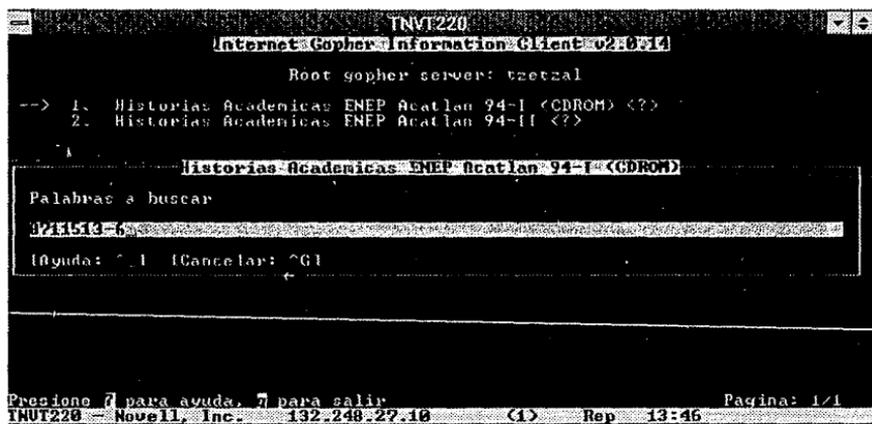
```

En este punto el sistema pide que el usuario teclee la información que se desea consultar. El sistema tiene la capacidad de realizar búsquedas booleanas, exactas y mediante algún metacaracter. Los siguientes ejemplos utilizarán estas tres formas.

Por ejemplo, una búsqueda exacta puede ser el número de cuenta de alguna persona, por lo que tecleamos:

8711513-6

como se muestra a continuación:



```
Internet Gopher Information Client v2.0.0.14
Root gopher server: tztztl
--> 1. Historias Academicas ENEP Acatlan 94-I <CDROM> <?>
    2. Historias Academicas ENEP Acatlan 94-II <?>

Historias Academicas ENEP Acatlan 94-I <CDROM>
Palabras a buscar
8711513-6
Ayuda: ^I | Cancelar: ^G|

Precione ? para ayuda, q para salir
Internet - Novell, Inc. 192.248.27.10 (1) Rep 13:46 Pagina: 1/1
```

Al dar <ENTER> obtenemos la información deseada, el alumno cuyo número de cuenta fue el solicitado:

```

TNU220
Internet Gopher Information Client 02-0-14
Historias Academicas ENEP Acatlan 94-I <CDROM>: 8711513-6
-->_1. NOMBRE: GONZALEZ REYES MARIO          No. CUENTA:8711513-6

Presione 7 para ayuda, 7 para salir, 0 para ir al menu superior Pagina: 1/1
TNU220 - Novell, Inc. 132.248.27.10 <1> Rep 13:50

```

Si damos <ENTER> la historia académica del alumno será presentada de la siguiente forma:

```

TNU220
NOMBRE: GONZALEZ REYES MARIO          No. CUENTA:8711513-6 <1> 78%
NOMBRE: GONZALEZ REYES MARIO          No. CUENTA:8711513-6
PLANTEL: E.N.E.P. ACATLAN ECONOMIA    CLAVE:208
CARRERA: LIC EN ECONOMIA              CLAVE:21
GENERACION: 94
SEMESTRE: 94-1

REPORTE DE AVANCE EN CREDITOS          REPORTE DE ASIGNATURAS
-----
Obligatorios   Cubiertos  Totales  %Avance  APROBADAS   : 0
0              0         0        0.00%    NO APROBADAS: 5
Optativos     0         0        0.00%    TOTAL       : 5
Totales       0         300      0.00%    PROMEDIO    : 0.00

PRIMER SEMESTRE
208 0241 06 GEOGRAFIA ECONOMICA        NP OR 94-1 1265619 1151 1
208 0291 06 HISTORIA ECONOMICA GENERAL I NP OR 94-1 1265627 1151 1

[Ayuda: ?] [Salir: ul] [Pagina Abajo: Espacio]
TNU220 - Novell, Inc. 132.248.27.10 <1> Rep 13:50

```

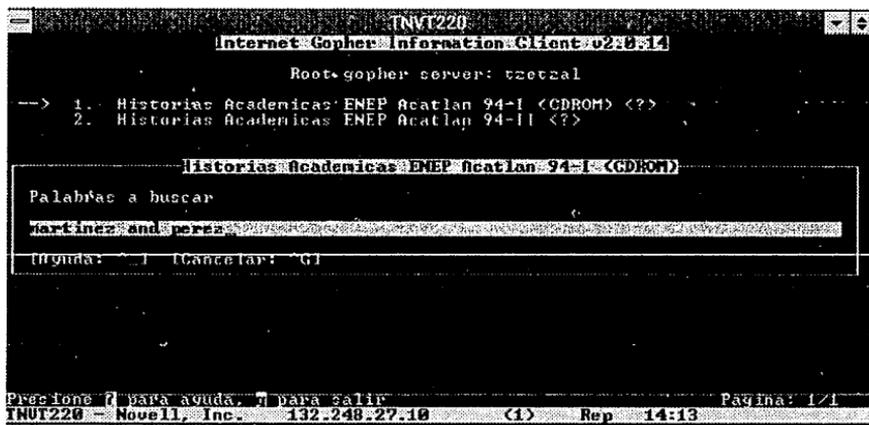
Como indica podemos navegar en ella y observar la historia académica total del alumno. El usuario tiene incluso la capacidad de imprimir la historia

académica desde su estación de trabajo a una impresora local conectada a esta. Esto se logra presionando la tecla p, para mayor referencia y ayuda el usuario puede pedir la ayuda presionando ?

Para una búsqueda booleana nos valemos de los operadores lógicos and, or y not y podemos hacer nuestra búsqueda tan compleja como queramos. Por ejemplo:

```
martinez and perez
```

de la siguiente forma:



```
Internet Gopher Information Client v2.0.14
Root gopher server: tzetzal
--> 1. Historias Academicas ENEP Acatlan 94-1 <CDROM> <?>
    2. Historias Academicas ENEP Acatlan 94-11 <?>

Historias Academicas ENEP Acatlan 94-1 <CDROM>
Palabras a buscar
martinez and perez
Ayuda: ? (Cancelar: C)
```

Presione ? para ayuda, q para salir. Pagina: 1/1
INUI220 - Novell, Inc. 132.248.27.10 (1) Rep 14:13

Mostrará un alumno que se llame Martínez y Pérez.

Una búsqueda con metacaracteres puede ser con *, el cual sustituye cualquier cadena, por ejemplo queremos saber que alumnos tienen el número de cuenta que comienza con 87:

```

INUT220
Internet Gopher Information Client v2.0.14

Root gopher server: tztatzal

--> 1. Historias Academicas ENEP Acatlan 94-I <CDROM> <?>
    2. Historias Academicas ENEP Acatlan 94-II <?>

Historias Academicas ENEP Acatlan 94-I <CDROM>

Palabras a buscar
87*

[ayuda: ^_] [Cancelar: ^G]

Presione ? para ayuda, q para salir, q para ir al menu superior Pagina: 1/1
INUT220 - Novell, Inc. 132.248.22.10 <1> Rep 14:16

```

Entonces despliega:

```

INUT220
Internet Gopher Information Client v2.0.14

Historias Academicas ENEP Acatlan 94-I <CDROM>: 87*

--> 1. NOMBRE: GONZALEZ REYES MARIO No. CUENTA: 87011513-6
    2. NOMBRE: AVILA SOLIS CARLOS No. CUENTA: 8704679-5
    3. NOMBRE: GONZALEZ-RODRIGUEZ FELIPE No. CUENTA: 8701346-5
    4. NOMBRE: ISLAS GARCIA JUSTO No. CUENTA: 8701435-2
    5. NOMBRE: GONZALEZ ARREOLA OLGA No. CUENTA: 8702467-8
    6. NOMBRE: SEQUERA RAMIREZ GABRIEL GALI No. CUENTA: 8737848-3
    7. NOMBRE: SAUCEDO ALVAREZ ELIA CONCEPCION No. CUENTA: 8703004-3
    8. NOMBRE: SALAZAR ROMERO ABRAHAM No. CUENTA: 8712019-6
    9. NOMBRE: ROMERO DURAN OSCAR No. CUENTA: 8702961-1
    10. NOMBRE: MENDOZA LOPEZ ANTONIO No. CUENTA: 8709031-0
    11. NOMBRE: RODRIGUEZ MUÑOZ MYRIAM No. CUENTA: 8709340-7
    12. NOMBRE: ARELLANO-BENITEZ MARIA ANTONIA No. CUENTA: 8709742-9
    13. NOMBRE: ANGUIANO NEGRET PATRICIA No. CUENTA: 8703260-8
    14. NOMBRE: CASIMIRO GARCIA RAUL No. CUENTA: 8700413-3
    15. NOMBRE: BALTAZAR BERNAL MIGUEL ANGEL No. CUENTA: 8706105-7
    16. NOMBRE: GARCIA IBARRA JUAN GUADALUPE No. CUENTA: 8728276-2
    17. NOMBRE: CRUZ LOPEZ MOISES No. CUENTA: 8753565-9
    18. NOMBRE: VALDEZ MEDINA SERGIO No. CUENTA: 8703087-9

Presione ? para ayuda, q para salir, q para ir al menu superior Pagina: 1/6
INUT220 - Novell, Inc. 132.248.22.10 <1> Rep 14:16

```

Como se puede observar se pueden hacer muchas combinaciones con estos tipos de búsqueda, pueden ser muy complejas o muy simples.

A continuación se muestra un cuadro que hace un resumen de las posibles búsquedas:

Metacaracter: *

Función: Sustituye cualquier cadena.

Ejemplo: Todos los alumnos que tengan número de cuenta que comience con
876: 876*

Operadores Booleanos: and, or, not

Función: Efectúa operaciones lógicas con la información.

Ejemplo: Todos los alumnos que se llamen Juan o Carlos y que se apelliden
García:

(juan or carlos) and garcia

... y su variantes:

(juan or carlos) and garcia not diaz

(juan not carlos) and garcia

(juan not (carlos or jose)) and garcia

Búsquedas exactas.

Ejemplo: el alumno que tenga el número de cuenta 8711513-6

Para salir del sistema se presiona q lo que nos regresará al menú de conexiones, al dar q nuevamente salimos del sistema.

- Ambiente gráfico.

Para tener acceso al ambiente gráfico de nuestro sistema se puede proceder de dos formas:

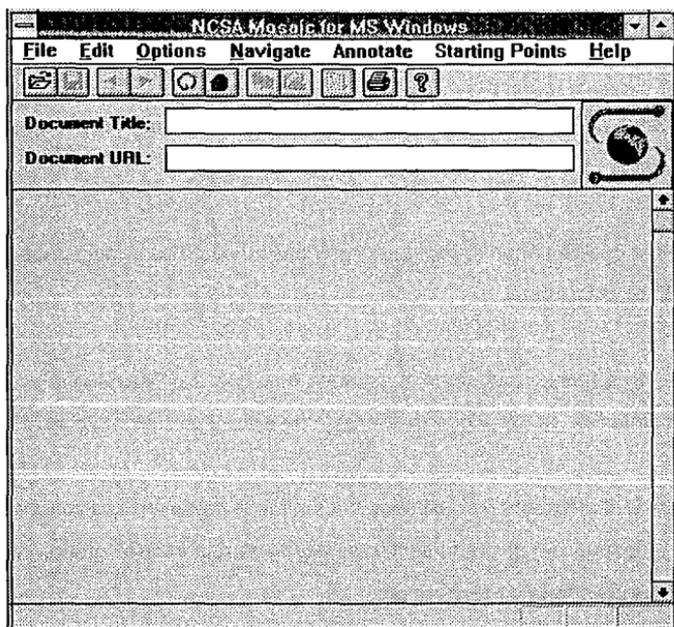
- 1) Indicar que se tienen capacidades gráficas cuando nos conectamos con `telnet a tzetzal`.

Si el monitor que tiene el usuario no es capaz de desplegar la interfaz, se despliega un mensaje de error y se cierra la conexión.

Si el programa no despliega ningún mensaje de error, entonces después de un cierto tiempo se desplegará la interfaz gráfica con una portada de bienvenida.

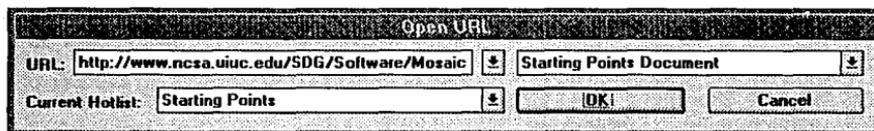
- 2) Conectarse mediante el programa MOSAIC para Windows.

La interfaz de Mosaic para Windows es la siguiente:



Para poder acceder al sistema debemos indicar el punto de partida para que el programa trate de conectarse y presentar la portada de bienvenida. La forma en que indicamos ésto es mediante un *URL*, es decir un *Unified Resource Locator*, ésto se logra de la siguiente manera:

Abrimos el menú de **f**ile de Mosaic y seleccionamos **o**pen URL, se visualiza la siguiente pantalla:

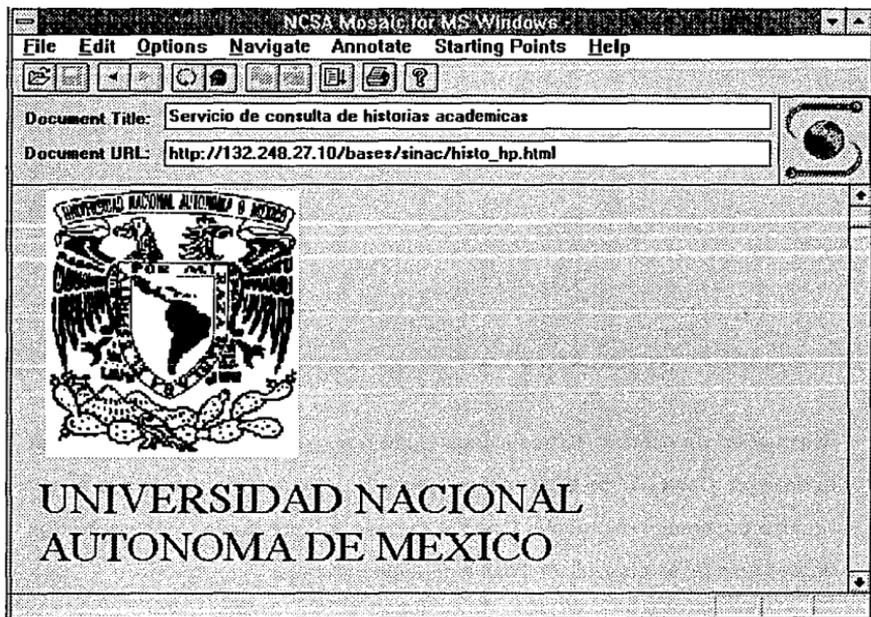


Borramos el texto que aparece en URL y tecleamos lo siguiente:

`http://tztetzal.dcaa.unam.mx/bases/sinac/histo_hp.html`

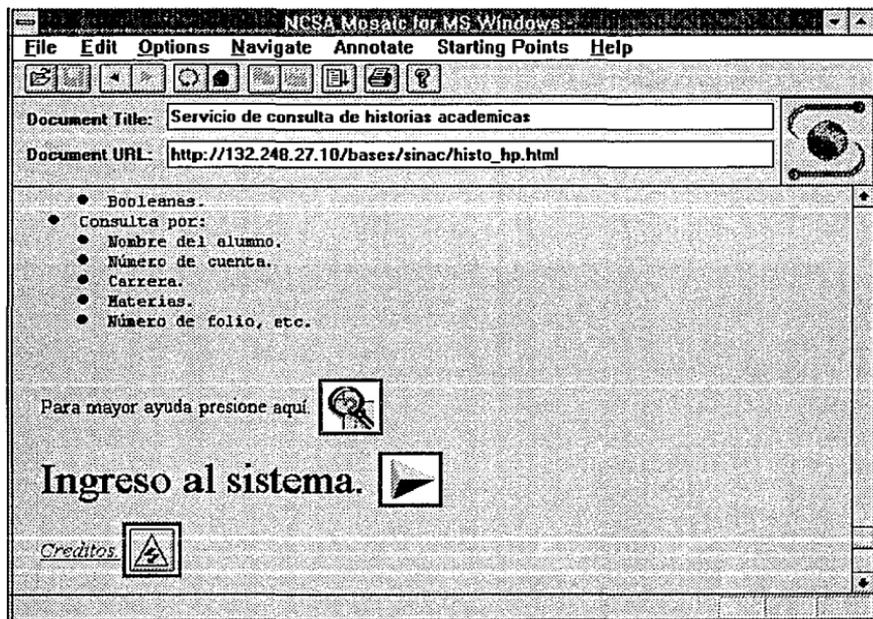
Presionamos el botón de OK.

Después de un momento se presentará la interfaz gráfica, ésta se muestra a continuación:



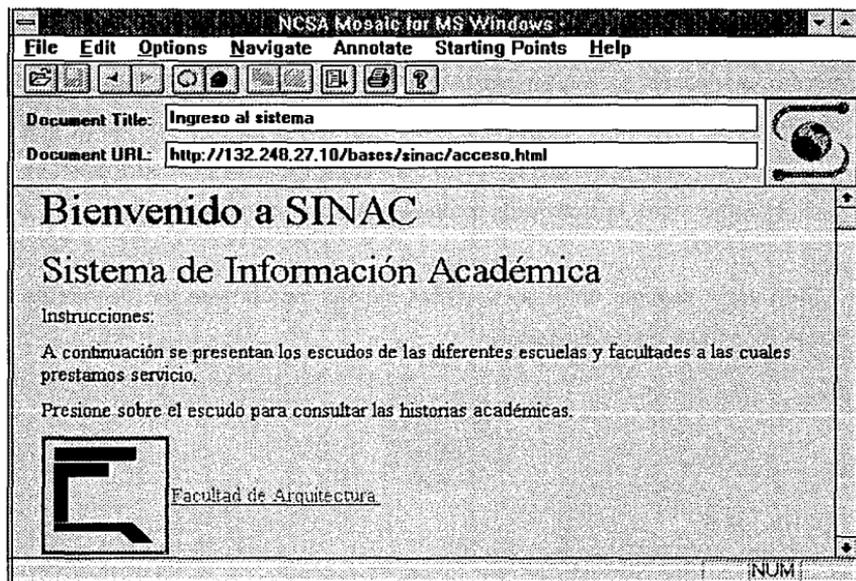
Hasta este punto, se está ya conectado al sistema. De aquí en adelante la explicación del sistema es la misma si el usuario se conecta con un telnet o si utiliza Mosaic para Windows.

Si hacemos un *scroll* de la pantalla veremos lo siguiente:



Se pueden ver claramente tres íconos: Información, Ingreso al sistema y Créditos. Cada uno de ellos es una liga a otros documentos, por lo que sí con el mouse presionamos el icono tenemos acceso al documento. Igualmente, las palabras resaltadas en azul son ligas a otros documentos.

Para acceder al sistema presionamos Ingreso al sistema, y se presenta la siguiente ventana:



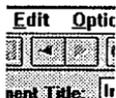
Simplemente presionando sobre el escudo de la Facultad o Escuela estamos en posibilidad de acceder a las historias académicas de éstas. En este momento solo tenemos las historias académicas de la ENEP Acatlán (como se había indicado anteriormente) pero esta vez se tienen en el escudo de la Facultad de Arquitectura con propósitos de demostración.

Al presionar el escudo de la Facultad de Arquitectura el sistema preguntará por el nombre del usuario así como un password, esto con el objetivo de presentar un

nivel de seguridad. Una vez que el sistema valida estos nombres se accesa a las consultas.

Después de ésto se presentará una pantalla de bienvenida y al final de ésta tenemos un ícono que nos permite tener acceso a las historias académicas. El número de íconos se incrementará una vez que tengamos la información semestral de las historias. Al presionarlo estamos en posibilidad de hacer una petición de la misma forma que se indicó para la interfaz orientada a caracter, es decir, podemos hacer búsquedas booleanas, exactas y por metacaracter.

Para poder navegar entre las pantallas, Mosaic nos provee de "flechas de navegación" que son las siguientes:



Con ellas podemos regresar o adelantar hojas las cuales ya hemos accedido.

Dudas o comentarios: favor a comunicarse al (62)236-75 ó 236-87

Coordinación Técnica de Redes e Interoperabilidad.

Con:

Gabriel Iván Montes de Oca Mendoza

ivan@tztetzal.dcaa.unam.mx

Ana Cecilia Padilla Chávez

cecilia@tztetzal.dcaa.unam.mx

Edgar Valencia Figueroa

edgar@tztetzal.dcaa.unam.mx



Glosario

ALGOL

ALGOrithmic Language. Lenguaje de programación estructurado que es el nativo de los equipo Series A de Unisys.

ANSI

American National Standards Institute. Grupo que define los estándares de EU para la industria del procesamiento de la información. ANSI participa en la definición de estándares para protocolos de redes.

APLICACIÓN

Programa que es creado para ayudar al usuario en una tarea específica.

BACKUP

Subsistema de control, manejo y operación de impresiones del sistema operativo MCP.

BATCH

Procesamiento de tareas en forma concurrente y sin intervención del usuario.

BBS

Boulettin Board Service o Boletín electrónico es una computadora preparada para recibir llamadas y actuar como un **host**. Los BBS permiten comunicarse entre sí a los usuarios mediante mensajes, así como intercambiar archivos.

BUFFER

Area reservada de memoria que en conjunto se comporta como una sola unidad.

CANDE

Command AND Editor. Sistema que permite la edición de archivos y ejecución de comandos del sistema operativo MCP.

CD-ROM

Compact Disc - Read Only Memory. Dispositivo de almacenamiento óptico de información. Similar en sus características a los discos compactos de audio.

CINTA MAGNÉTICA

Dispositivos de almacenamiento magnético secuencial de información. Generalmente utilizado para respaldar grandes volúmenes de datos.

CLIENTE

Programa de computadora que utiliza los recursos dados por otra máquina o programa a través de la red.

CUI

Character User Interface. Interfaz de presentación de datos orientada al despliegue únicamente de caracteres.

DAEMON

Programa residente en memoria que ejecuta algún servicio o acción muy específica y que solo es aplicable a los sistemas operativos UNIX.

DESCRAMBLING

Proceso llevado a cabo durante la lectura de la información grabada en el CD-ROM. Al grabar la información se deben de mezclar los bits de datos con bits especiales para igualar la longitud de los registros. El proceso de quitar estos bits especiales es el que se denomina **descrambling**.

DOMINIO

Parte de la jerarquía de nombres de sistemas. Sintácticamente un nombre de un dominio consiste en una secuencia de nombres u otras palabras separadas por puntos.

DUMPALL

Utilería que genera impresiones de archivos, controla el vaciado de cintas y permite a los archivos ser copiados entre medios diferentes, como puede ser de disco a cinta, entre otros. Esta utilería permite desplegar los archivos con diferentes formatos y es característica de los equipos Serie A de Unisys.

EBCDIC

Código expandido de intercambio de información codificado en binario. Es de 8 bits por carácter y se utiliza a menudo en mainframes.

EFM

Código de modulación que añade bits adicionales para prevenir la presencia de 1's consecutivos y reducir las señales de baja frecuencia.

ELECTROFORMING

Proceso utilizado en la fabricación del disco maestro que servirá de molde para la reproducción masiva de CD's.

ESCALABILIDAD

Característica de los mainframes de Unisys que permite crecer desde el equipo más básico hasta un equipo más poderoso, cambiando solamente los módulos necesarios.

FDDI

Fiber Distribution Data Interfase. Estándar reciente de comunicaciones basado en fibra óptica que ha sido establecido por ANSI. FDDI especifica una velocidad de señalización de 100 Mbps y una longitud máxima de 200 Km.

FIRMWARE

Hardware programado permanentemente.

FQDN

Full Qualified Domain Name. El FQDN es el nombre completo de un sistema, no solamente el nombre del host en sí. Por ejemplo, el sistema tzetza1 que se encuentra en la DCAA tiene un FQDN de tzetza1.dcaa.unam.mx

FRAMES

Es un paquete transmitido por una línea serial. El término deriva de los protocolos orientados a carácter que añaden caracteres especiales que indican el principio y fin del frame cuando transmite paquetes de información.

FRONT END

Término que se refiere a la parte de una aplicación que interactúa directamente con el usuario.

FTP

File Transfer Protocol. Este es el protocolo de alto nivel estándar de la Internet para la transferencia de archivos de una computadora a otra.

FYI

For Your Information. Término dado a los documentos almacenados en varios **host** de la red **Internet** con un lenguaje no técnico que describen tópicos relacionados con la misma red.

GATEWAY

Una computadora dedicada a unir dos redes y rutear paquetes de una a otra. Los **gateways** rutean paquetes a otro **gateway** hasta que sean entregados a su destino final a través de la red física.

GUI

Graphical User Interfase. Interfaz de salida que permite la visualización de imágenes, gráficas, etc.

HARDWARE

Elementos físicos que componen a un sistema electrónico.

HIPERLIGA

Medio a través del cual se puede apuntar de forma única a un documento, imagen, video, sonido a través de la red.

HIPERMEDIOS

Conjunción de datos de diferentes formatos, como pueden ser textos, imágenes, audio, video, etc. los cuales pueden ser accedidos por una aplicación.

HOST

Cualquier computadora capaz de ejecutar programas de aplicación que se encuentra conectada a una red.

HTML

HyperText Markup Language. Este es un lenguaje de descripción muy simple, pero lo suficientemente complejo para permitir formateo y presentación de documentos de hipertexto.

HTTP

HyperText Transfer Protocol. Protocolo estándar para la transferencia de documentos de hipertexto.

INTERLEAVING

Proceso realizado durante la grabación de la información en disco compacto que evita los errores al leerla, producidos por suciedad en los discos. Este proceso reacomoda la información de tal manera que piezas consecutivas de esta no sean grabadas en forma continua en el disco.

ISO 9660

Estandar definido por el organismo llamado International Standards Organization que indica exactamente como se debe grabar la información en un CD-ROM de tal forma que esta pueda ser consultada por cualquier computadora no importando el hardware o sistema operativo de ésta, además describe el contenido y organización de la información a ser grabada en el disco.

JUKEBOX

Dispositivo electromecánico que permite la consulta de manera concurrente de varios CD-ROM's.

LAN

Local Area Network. Cualquier tecnología física de redes que opera a altas velocidades sobre distancias cortas (por ejemplo a pocos cientos de metros).

LISTAS DE CORREO (Mailing List).

Una lista de correo es un grupo de discusión, distribuido por correo electrónico desde una computadora central que mantiene una lista de gente relacionada con algún tema de discusión.

MAINFRAME

Término aplicado a las computadoras de alto rendimiento con características de un alto nivel transaccional, gran capacidad para atender gran cantidad de usuarios y con requerimientos especiales en cuanto a condiciones ambientales para su funcionamiento.

MBUS

Módulos que conectan la memoria principal y los procesadores SuperSparc de los equipos Sun. Contiene los siguientes circuitos centrales de procesamiento: Unidad entera, unidad de punto flotante y controlador de memoria caché.

MCP

Master Control Program. Sistema operativo diseñado desde 1963 para equipos Bourroughs y Unisys.

MICROFICHA

Unidad de material fotográfico en que las imágenes de información son grabadas en un arreglo matricial.

MICROGRAFICACIÓN

Proceso mediante el cual se reduce el tamaño del medio en donde reside la información a imágenes fotográficas o fílmicas, comúnmente llamadas microformas.

MIPS

Millones de instrucciones por segundo. Término utilizado para medir la capacidad de procesamiento de un equipo de cómputo.

MULTIMEDIA

Conjunción de video, texto, imágenes, sonidos, gráficas, etc. en un programa que permite la más fácil interacción entre el programa y los usuarios.

OSI

Open System Interconnection. Estándar de arquitectura de redes propuesto por ISO.

PÁGINA

Subdivisión natural de la memoria. El número de página se identifica por los bits más significativos de la dirección. Las posiciones dentro de la página las determinan los bits menos significativos.

PREMASTER

Proceso por el cual se obtiene una imagen de la información que posteriormente será grabada a un CD-ROM.

PROTOCOLO

Conjunto de convenciones y normas para establecer un diálogo a distintos niveles de implementación de una arquitectura de redes.

QUERY

Programa o consulta que permite al usuario tener acceso a información contenida en algún repositorio (base de datos o archivos).

REDUNAM

Proyecto de interconexión de cómputo en la comunidad universitaria.

RFC

Request For Comments. Documentos técnicos que se encuentran distribuidos en la red Internet que definen estándares y políticas de ésta.

RISC

Reduced Instruction Set Computer. Arquitectura de procesadores que implementan un conjunto reducido de instrucciones para obtener un mejor desempeño en el equipo de cómputo. Utilizado en estaciones de trabajo.

SBUS

Módulos utilizados en los equipos Sun que se encargan de la conexión del equipo con sus periféricos y dispositivos de entrada/salida.

SCRIPT

Conjunto de instrucciones y/o comandos (generalmente de un sistema operativo como UNIX) cuyo fin es el de automatizar o programar alguna acción.

SCSI

Small Computer System Interface. Definición lógica y física para la interconexión de **hosts**, unidades de disco, cinta, y otros periféricos.

SEARCH ENGINE

Parte de un programa o programa encargado de realizar búsquedas de información en archivos planos o bases de datos.

SERVIDOR

Programa de computadora o máquina que da recursos a otras computadoras o programas.

SISTEMA OPERATIVO

Software del sistema que controla el funcionamiento de la computadora, incluyendo varias tareas, como asignación de posiciones de memoria, procesamiento de interrupciones y manejo de tareas.

SITE

Centro de cómputo.

SOFTWARE

Conjunto de programas que hacen uso a su vez de los respectivos conjuntos de datos para su funcionamiento.

SQL

Structured Query Language. Lenguaje que permite al usuario manejar *queries* de una forma sencilla.

STACK

Grupo de registros o posiciones de memoria, usados para guardar las direcciones de retorno durante la ejecución de subrutinas.

TCP/IP

(Transmission Control Protocol / Internet Protocol). Este es un conjunto de protocolos, resultado de los esfuerzos de ARPA, usado por la Internet para mantener servicios como login's remotos (telnet), transferencias de archivos (FTP) y correo electrónico (SMTP).

TELEPROCESO

Forma de comunicación entre terminales y el mainframe, en donde las terminales solo realizan la función de presentación y captura de datos de parte del usuario mientras que el mainframe realiza el procesamiento en sí.

TELNET

Es el protocolo estándar de la Internet del servicio de conexión remota de terminales. Telnet permite a un usuario en una máquina interactuar con un sistema remoto de tiempo compartido en algún otro lugar como si la terminal del usuario estuviera conectada directamente al sistema remoto.

TPS

Ticks Per Second. Terminó utilizado para medir la rapidez del reloj de una computadora.

TRACK

Termino utilizado para describir el lugar físico donde la información es almacenada en un disco, se les conoce comúnmente como pistas, las cuales a su vez se dividen en sectores. Cada una de estas pistas es la ruta que siguen las cabezas lectoras cada vez que el disco da vuelta.

UNIX

Sistema Operativo interactivo de tiempo compartido desarrollado en 1969 por Ken Thompson después que los laboratorios Bell abandonaran el proyecto Multics. UNIX existe en varias formas e implementaciones, siendo las más comunes System V y BSD.

URL

Uniform Resource Locator. Se puede pensar el URL como una extensión de los nombres de los archivos, ya que no solamente podemos apuntar al archivo en sí, sino que también en qué máquina se encuentra y puede ser accedida por muchos métodos.

WFL

Work Flow Language. Lenguaje a través del cual se provee al usuario de los medios para construir un job (trabajo) que permita a múltiples tareas o programas ser ejecutados en sistemas de las Series A. WFL permite a cada trabajo ser descrito como un conjunto de tareas a ser ejecutadas teniendo la posibilidad de la ejecución de forma paralela o serial.

WORKSTATION

Estación de trabajo. Cualquier combinación de dispositivos de cómputo que provee al usuario de un gran poder de procesamiento y velocidad, gráficas sofisticadas y un medio de almacenamiento permanente y de rápido acceso. Generalmente conectado a red y con un sistema operativo poderoso.

WORM

Write Once Read Many. Término que se aplica a los discos compactos (CD's) que permiten una grabación (por parte del usuario) de forma permanente, pudiéndose acceder múltiples veces.

WYSIWYG

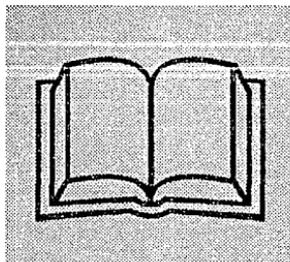
What You See Is What You Get. Acrónimo de la frase "lo que ves es lo que obtienes". Esta es una característica de programas de aplicación en el que la presentación en la pantalla será igual a la obtenida por una impresión en papel.

WWW

World Wide Web (W³). Proyecto realizado por el CERN (European Laboratory for Particle Physics, ubicado en Ginebra Suiza) el cual permite acceder información a través de la red sobre cualquier tópico utilizando para ello hipertexto y técnicas de multimedia.

X11

Conjunto estandarizado de rutinas para el manejo gráfico de pantallas, aplicaciones, etc. que son totalmente independientes del hardware utilizado para dicho proposito. Este fue desarrollado por el MIT y es de dominio público.



Bibliografía

- Getting Started with NCSA Mosaic
Marc Anderssen
National Center for Supercomputing Applications
Mayo 8, 1993
marca@ncsa.uiuc.edu
- NCSA Moiscac Technical Summary
Marc Anderssen
National Center for Supercomputing Applications
Mayo 8, 1993
marca@ncsa.uiuc.edu
- A Begginer's Guide to URLs
Marc Anderssen
National Center for Supercomputing Applications
Mayo 8, 1993
marca@ncsa.uiuc.edu
- A Beginner's Guide to HTML
Marc Anderssen
National Center for Supercomputing Applications
Mayo 8, 1993
marca@ncsa.uiuc.edu
- Surfing the Internet: an Introduction
Jean Armour Polly
Mayo 15, 1993
jpolly@nysernet.org

Bibliografía

- FYI on "What is the Internet ?
E. Krol, E. Hoffman
Mayo, 1993
Network Working Group
FYI #20
RFC #1462
- The Whole Internet User's Guide and Catalog.
Ed Krol
O'Reilly & Associates, Inc.
1993
- Zen and the Art of the Internet. A Beginner's Guide to the Internet
Brendan P. Kehoe
Febrero 2, 1992
brendan@cs.winder.edu
- WAIS Bibliography
WAIS Inc.
Agosto 11, 1993
info@wais.com
- Páginas de Manual (man) del compilador de "C" de GNU (gcc)
Free Software Foundation.
gnu@prep.ai.mit.edu
- Internet Gopher User's Guide
Paul Lindner, Universidad de Minnessota
1993
gopher@boombox.micro.unm.edu
- Manual de MakeDisk
Young Minds, Inc.
1993
support@ymi.com
- Memory Management for all of us
John M. Goodman
SAMS
1992
- Client/Server System Desing & Implementation
Larry T. Vaughn
Mc-Graw Hill
1994

- Listas de correo de Mosaic, Solaris Admin y grupos de Noticias.
- Revista: CD-ROM Professional.

Artículos:

- "Premastering and CD-WO software for the PC-Evaluations of top In-house publishing software"
Data Distribution Laboratory at the Jet Propulsion Laboratory
Mayo, 1993
- "So you want to Publish a CD-ROM--Practical tips and Hard-Won Advice"
David A. Schultz
Enero, 1993
- "CD-R: the next stage in CD-ROM Evolution"
Dataware Technologies, Inc.
Marzo, 1993
- "Compact Disc Recording: A Technical Overview."
John M. Hartigan
Septiembre, 1993
- "Tips for Designing the Next Generation of CD-ROM Software"
Candy Schwartz
Enero, 1994
- "Glosary of Basic CD-ROM Terms"
Dana J. Parker
Enero, 1994
- "Choosing a CD-ROM Replicator"
Robert J. Headrick
Mayo, 1994
- FAQ mensual del grupo de noticias alt.cd-rom
rab@cdrom.com
Aprobado por: news-answers-request@MIT.edu
ftp.cdrom.com : /pub/cdrom/faq

- Manuales de los equipos Unisys
 - A Series ALGOL
Programming Reference Manual
 - A Series System
Funtional Overview
 - Capabilities Overview
A 12 System
 - A Series System Software Utilities
Operations Reference Manual
- Manuales del equipo SparcStation 10 de SUN
 - Programming Utilities
 - User's Guide
 - Reference Manual