

25  
2ej

UNIVERSIDAD NACIONAL AUTONOMA DE MEXICO  
SECRETARIA DE EDUCACION PUBLICA  
DIRECCION GENERAL DE INSTITUTOS DE INVESTIGACION Y ENSEÑANZA  
CARRERAS DE INGENIERIA  
MEXICO, D.F. 1995

**UNIVERSIDAD NACIONAL AUTONOMA DE MEXICO**

**ESCUELA NACIONAL DE ESTUDIOS PROFESIONALES**

**" A R A G O N "**



**\*SISTEMATIZACION DE LA INSCRIPCION DE LAS DIFERENTES AREAS QUE SE IMPARTEN EN LA E.N.E.P.ARAGON (ARQUITECTURA, PEDAGOGIA, ECONOMIA, DISEÑO INDUSTRIAL, Y PLANIFICACION PARA EL DESARROLLO AGROPECUARIO)\***

**T E S I S**

**QUE PARA OBTENER EL TITULO DE INGENIERO EN COMPUTACION**

**P R E S E N T A**

**ERENDIRA DE LOS ANGELES HERNANDEZ CRASBORN**

**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.

**EN LA VIDA TRIUNFA AQUEL QUE SE ENFRENTA A SUS  
PROBLEMAS, NO QUIEN SE LAMENTA DE ELLOS.**

**Dedico esta tesis a:**

**A mi Madre:**

**POR SU EJEMPLO, POR SU CONSTANCIA  
Y DISCIPLINA. GRACIAS POR TODO.**

**A mi Padre:**

**POR SU GRAN APOYO EN ESTA PARTE DE MI  
VIDA. POR SU GRAN ENTENDIMIENTO, POR  
SU CONFIANZA Y POR QUE ALGUN DIA  
ME DIJO:**

**YO SE QUE LO VAS A LOGRAR A PESAR DE  
TODO.**

**MUCHAS GRACIAS.**

**A mi Hermano:**

**CON MUCHO CARIÑO Y RESPETO. PORQUE SE QUE TAMBIEN VA A LOGRAR  
TODO LO QUE SE PROPONGA. PORQUE YO CONFIO EN EL.**

**A mi Tia Yoli:**

**GRACIAS POR TU ANIMO Y POR TU EJEMPLO.**

**Al Ing. Marcela Castillo y al Ing. Antonio Ortiz:**

Por su colaboración conmigo y por darme las facilidades para lograr esto. Gracias

**A mis compañeros en :**

Moisés y José por su ayuda y por los momentos que nos estuvimos riendo juntos.

A Claudia y a Gabriel :  
Por su apoyo moral.

Al personal del Departamento de Servicios Escolares:  
Muchas gracias por su apoyo.

Y Agradezco a todas las personas que participaron y me motivaron a realizar esta tesis.

Si hay algunas personas que no mencionó es porque son muchas y no terminaría nunca, lo cual también les agradezco.

En especial al Ing. Manuel Martinez Ortiz:

Por las ganas de vivir y hacer las cosas diferentes.

Porque a través del poco tiempo que lo conozco, se que es una persona que vale más que muchas más.

Gracias por tu motivación, tu apoyo y la confianza que me has dado.

SISTEMATIZACION DE INSCRIPCION DE LAS  
DIFERENTES AREAS QUE SE IMPARTEN EN LA ENEP ARAGON  
(ARQUITECTURA, DISEÑO INDUSTRIAL, ECONOMIA, PEDAGOGIA, Y  
PLANIFICACION PARA EL DESARROLLO AGROPECUARIO).

OBJETIVO: DISEÑAR UN SISTEMA, PARA MEJORAR EL PROCEDIMIENTO  
DE REINSCRIPCION DE LAS CARRERAS DE ARQUITECTURA,  
DISEÑO INDUSTRIAL, ECONOMIA, PEDAGOGIA Y  
PLANIFICACION PARA EL DESARROLLO AGROPECUARIO, A  
TRAVES DE LA COMPUTADORA.

## INDICE

## INDICE

Pag.

INTRODUCCION .....	5
--------------------	---

### CAPITULO I.- SISTEMAS DE INFORMACION Y BASES DE DATOS.

I.1. SISTEMAS DE INFORMACION	
I.1.1. Definición de Sistema.....	8
I.1.2. Definición de Información.....	9
I.1.3. Definición de Sistemas de Información.....	10
I.2. BASES DE DATOS.	
I.2.1. Introducción y Evolución de las Base de Datos.....	12
I.2.2. Definición Bases de Datos.....	15
I.2.3. Niveles de Abstracción de una Base de Datos.....	16
I.2.4. Estructuras o Modelos de datos.....	17
I.2.5. Tipos de Bases de Datos.....	18
I.2.6. Estructura general de una Base de Datos.....	20
I.2.7. Enfoques para el Análisis de Datos.....	21

### CAPITULO II.- ESTRUCTURA Y ANALISIS DEL PROCESO DE REINSCRIPCION

II.1. FUNCIONES DEL DEPARTAMENTO.....	23
II.2. ANALISIS DEL PROCESO DE REINSCRIPCION DE LAS CARRERAS A AUTOMATIZAR.	
II.2.1. Proceso de Reinscripción (manual).....	24
II.2.2. Proceso de ALTAS, BAJAS y CAMBIOS.....	26
II.2.3. Procesamiento de la Información generada por la reinscripción.....	27
II.2.4. Desventajas del proceso de reinscripción manual.....	27
II.2.5. Planteamiento del problema y soluciones, Análisis del Sistema.....	28

### CAPITULO III.- IMPLEMENTACION DE LA RED

III.1. INTRODUCCION A LAS REDES.....	32
III.1.1. Estratos Fundamentales de Arquitecturas de las Redes.....	34
III.1.2. Niveles de Interconexión.....	35
III.1.3. Redes Locales (LAN NETWORK).....	36
III.1.4. Características de las Redes Locales.....	38
III.1.5. Características de Operación de tres Redes Locales importantes.....	41
III.2. DESCRIPCION DE LA RED NOVELL	
III.2.1. Introducción a la Red Novell.....	43
III.2.2. Componentes de una red Novell.....	44
III.2.3. Estructura de la red Novell.....	47
III.3. ARQUITECTURA DE LA RED NOVELL.....	48

### CAPITULO IV.- DISEÑO DEL SOFTWARE

IV.1. DISEÑO DEL SOFTWARE.....	53
IV.2. ALTERNATIVAS DE DISEÑO.....	53
IV.3. DIAGRAMA MODULAR DEL SISTEMA.....	56
IV.3.1. El Sistema Visión General.....	56
IV.3.2. Modulo de Reinscripción.....	57
IV.3.3. Modulo de Consultas.....	62
IV.4. FLUJO DE LA INFORMACION	
IV.4.1. Archivos de Trabajo o Bases de Datos.....	64

### CAPITULO V.- APLICACION

V.1. PRUEBAS Y VALIDACION.....	70
V.2. DEPURACION.....	72
V.3. MANTENIMIENTO DEL SISTEMA.....	73
V.3.1. Confiabilidad.....	74
V.3.2. Confidencialidad.....	74

**INDICE**

	<b>Pag.</b>
<b>CONCLUSIONES</b> .....	<b>76</b>
<b>APENDICE A</b> .....	<b>78</b>
<b>BIBLIOGRAFIA</b> .....	<b>104</b>

## INTRODUCCION

Actualmente el hombre ha llegado a un punto de avance tecnológico, que ha creado aparatos e instrumentos que le ayudan a facilitar su trabajo, inclusive a disfrutar más de la vida.

El almacenamiento y el análisis de la información ha sido uno de los grandes problemas al que se ha enfrentado desde que inventó la escritura. No es sino hasta la segunda mitad del siglo XX que el hombre ha podido resolver, parcialmente, ese problema gracias a la invención de la computadora.

Sin embargo, para resolver ese problema, se han ido estableciendo varias técnicas de análisis de la información. Este análisis ha servido para integrar grandes bancos de información. A su vez, toda esa información almacenada sirve para conjuntar un grupo de bases de datos, que posteriormente servirán para integrarse en programas y/o sistemas, que ayuden a simplificar el trabajo en distintos lugares requeridos.

Por lo tanto, esto nos lleva a integrar y organizar mejor los centros de trabajo (llamense Empresas, Negocios u Oficinas) dentro del concepto de AUTOMATIZACION.

El siguiente concepto explica un poco a cerca de la AUTOMATIZACION de las oficinas:

*"La automatización de una oficina es la convergencia de herramientas de cómputo que permiten a los empleados y ejecutivos la manipulación efectiva de la información de una forma rápida y eficiente." Ahora el usuario ya no tendrá que levantarse para acceder un archivero y consultar información o esperar días para que le entreguen correspondencia. Estas y otras tareas serán efectuadas electrónicamente."*

De acuerdo a lo anterior, el tema principal de esté proyecto es:

La Automatización de las Reinscripciones de las carreras de Arquitectura, Diseño Industrial, Economía, Planificación para el Desarrollo Agropecuario y Pedagogía en el departamento de Servicios Escolares de la ENEP-ARAGON. Con el objetivo de mejorar los servicios hacia los alumnos, así como, la calidad del trabajo aportado por el personal.

Para esto en breve, he trabajado en la realización de esta Tesis, presentándola en cinco capítulos.

Los primeros capítulos son conceptos generales de Sistemas de Información, Bases de datos e Ingeniería de Software y conceptos de redes (en este caso se menciona el manejo de la red novell).

Los siguientes capítulos muestran el seguimiento que se le da a un sistema. Es decir, los sistemas llevan los siguientes procedimientos o lo que se llama el ciclo de vida clásico de un sistema de información:

#### **ANALISIS, DISEÑO, CODIFICACION, PRUEBA Y MANTENIMIENTO.**

Particularmente, el capítulo uno hace mención del significado de un sistema de información, hasta integrar un conjunto de datos relacionado entre sí, definiendo así lo que se conoce como una base de datos.

El capítulo dos hace una descripción de las funciones del departamento. Así como, algunas comparaciones con el planteamiento del proyecto actual, determinando soluciones.

En el capítulo tres se describen algunos conceptos básicos de Redes y la constitución estándar o general de una Red Novell. Ya que el departamento de Servicios Escolares cuenta con la instalación de una red Novell, en la cual se desarrolla el proceso de reinscripciones para el mejor flujo de la información.

El capítulo cuatro plantea el desarrollo del sistema aplicado a la información obtenida a partir de la investigación realizada en las coordinaciones de las carreras que se integrarán en el sistema.

El capítulo cinco determina la funcionalidad del sistema desarrollado. Es decir, que tan factible, cuanta confiabilidad y confiabilidad podemos tener, así como que tipo de mantenimiento y mejoras le podemos hacer al sistema.

Finalmente hago referencia al código del programa dentro de un apéndice y un glosario de términos técnicos que es de gran importancia especificar su definición.

Y por último espero que este proyecto sea de gran ayuda y sirva para mejorar la forma de inscripción de los alumnos de esta escuela. Así como el mejoramiento de nosotros mismos en todos los aspectos y que de alguna u otra manera estemos a la altura y vanguardia de las tecnologías nuevas que invaden a nuestro país.

*"Un término abstracto es como una valija de doble fondo, es posible poner en ella las ideas que uno desee y sacarlas de nuevo, sin ser observado."*

Alexis de Tocqueville  
*La democracia en América (1835)*

## **CAPITULO I.- SISTEMAS DE INFORMACION Y BASES DE DATOS.**

### **I.1. SISTEMAS DE INFORMACION**

- I.1.1. Definición de Sistema.**
- I.1.2. Definición de Información.**
- I.1.3. Definición de Sistemas de Información.**

### **I.2. BASES DE DATOS.**

- I.2.1. Introducción y Evolución de las Base de Datos.**
- I.2.2. Definición de Base de Datos.**
- I.2.3. Niveles de Abstracción Datos.**
- I.2.4. Estructuras o Modelos de Datos**
- I.2.5. Tipos de Bases de Datos**
- I.2.6. Estructura General de un Sistema de Base de Datos.**
- I.2.7. Enfoques para el análisis de Datos.**

**OBJETIVO: DEFINIR LOS CONCEPTOS GENERALES DE SISTEMAS DE INFORMACION Y BASES DE DATOS PARA UN MEJOR ENTENDIMIENTO DE ESTE TRABAJO.**

En este capítulo se describirá los conceptos básicos para el desarrollo de un Sistema de información, desde la especificación de sistema e información hasta la integración de uno. Así como la integración de los datos en una Base de Datos. Ya que en la aplicación de este trabajo es muy importante.

## **I.1. SISTEMAS DE INFORMACION**

Los sistemas de bases de datos se diseñan para manejar grandes cantidades de información. El manejo de los datos incluye tanto la definición de las estructuras para el almacenamiento de la información como los mecanismos para el manejo de la misma.

El procesamiento de información ha constituido una de las tareas básicas de cualquier organización. Debido al crecimiento económico y demográfico, existen necesidades crecientes de administrar grandes cantidades de datos interrelacionados con fines comerciales administrativos. Los datos interrelacionados considerados en conjunto forman lo que se llama un sistema. Todo sistema que simplifica significativamente los asuntos humanos presumiblemente necesitará un modelo complejo del mundo real, además de otras formas convencionales de conocimiento.

### **I.1.1 Definición de Sistema.**

La palabra sistema se utiliza demasiado, y en el sentido más amplio, un sistema es simplemente un conjunto de componentes que interactúan para alcanzar un objetivo. Los sistemas son, de hecho, todo lo que rodea al ser humano. El hombre se comunica por medio del lenguaje, que es un sistema altamente desarrollado de palabras y símbolos que tienen significado, vive de acuerdo a un sistema económico en el cual los bienes y servicios se intercambian por otros de valor comparable y por medio de los cuales (al menos en teoría) los participantes de este intercambio se benefician.

Con frecuencia no se advierte pero un negocio también es un sistema. Sus partes tienen nombres como mercadotecnia, producción, ventas, etc. Estos componentes trabajan todos juntos para crear una utilidad que beneficie a los empleados y al personal en general en su negocio. Cada una de esas partes es un sistema en sí mismo.

Cuando se comienza a ver lo abundante que son los sistemas, no sorprende darse cuenta que cada sistema del negocio depende de una o más entidades abstractas llamadas **SISTEMAS DE INFORMACION**.

**FALLA DE ORIGEN**

### I.1.2. Definición de Información.

De acuerdo al concepto dado por John G. Burch:<sup>1</sup>

**"La Información es un recurso crítico de las organizaciones, tan fundamental como la energía o las máquinas. Es el eslabón indispensable que une a todos los componentes de la organización para una mejor operación y coordinación y su supervivencia en un ambiente competitivo y poco amigable. A decir verdad, las compañías actuales funcionan por la información."**

La información la componen datos que se han colocado en un contexto significativo y útil y se ha comunicado a un receptor, quien la utiliza para tomar decisiones. La información implica la comunicación y recepción de inteligencia o conocimiento. Evalúa y notifica, sorprende y estimula, reduce la incertidumbre, revela alternativas adicionales o ayuda a eliminar las irrelevantes o pobres, e influye sobre otros individuos y los estimula a la acción.

La información esta compuesta de datos, imágenes, texto, documentos y voz, a menudo en forma inexplicable, pero siempre organizados en un contexto significativo.

El siguiente esquema muestra el ciclo de la información. Los datos se procesan mediante modelos para crear información, el receptor recibe la información y luego toma una decisión y actúa; esto genera otras acciones o eventos, que a su vez crean diversos datos dispersos que se capturan y sirven como entrada, y el ciclo se vuelve a repetir.

---

<sup>1</sup> Autor del libro DISEÑO DE SISTEMAS DE INFORMACION TEORIA Y PRACTICA Ed. Noriega Megabyte.

### **I.1.3. Definición de Sistema de Información.**

De acuerdo a los datos mencionados anteriormente podemos definir que es un Sistema de información, así como determinar su funcionalidad.

Los sistemas de información son como cualquier otro sistema dentro de una empresa en cuanto que tienen propósitos e interactúan con otros componentes de la compañía. La tarea de los sistemas de información consiste en procesar la entrada, mantener archivos de datos en relación con la empresa y producir información, reportes y otras salidas.

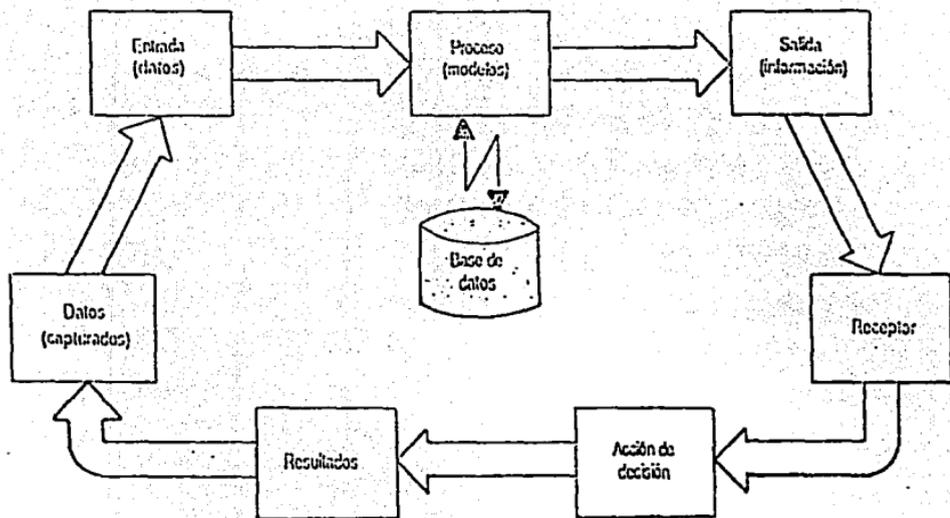
Los sistemas de información están integrados por subsistemas que incluyen el hardware, software y almacenamiento de datos para los archivos y bases de datos. Procedimientos específicos describen los sistemas utilizados. El conjunto particular de subsistema, es decir, el equipo específico, programas, archivos y procedimientos, comprenden una aplicación de sistema de información. Por lo tanto, los sistemas de información pueden tener aplicaciones de compras, ventas, etc.

Los componentes de los sistemas de información incluyen hardware, software y datos almacenados en archivos y bases de datos. Las aplicaciones de los sistemas de información son procedimientos específicos, programas, archivos y equipo, todos cuidadosamente integrados para lograr metas específicas.

Por medio de estos sistemas los datos pasan de una persona a otra, de un departamento a otro y pueden realizar cualquier cosa. Los sistemas de información sirven a todos los sistemas de un negocio, son el lazo que mantiene unidos a diferentes componentes en forma total, que pueden trabajar de manera efectiva hacia el mismo objetivo.

**FALLA DE ORIGEN**

FALLA DE ORIGEN



CICLO DE LA INFORMACION

## **I.2. BASES DE DATOS**

### **I.2.1. Introducción y Evolución de una Base de Datos**

Los sistemas de bases de datos comenzaron a aparecer en la década de los 60's. En los siguientes veinte años sufrieron grandes transformaciones en sus conceptos y tecnología. Actualmente, sin embargo, parece que gran parte de la teoría y práctica de los sistemas de bases de datos ha madurado y se ha estabilizado; sus conceptos fundamentales están bien definidos y se conocen a fondo. Aunque no hay duda que seguirán avanzando, es poco probable que cambie el enfoque básico hacia su manejo.

#### **Etapa 1.- Simple archivos de datos (principios de los 60's)**

- \* Organización de manera serial.
- \* La estructura física era la misma que la lógica.
- \* Procesamiento Batch.
- \* Sólo operaciones de entrada y salida.
- \* Existía exceso de redundancia.
- \* Los datos servían para una sola aplicación.

#### **Etapa 2.- Métodos de acceso (finales de los 60's)**

- \* Existen accesos seriales y aleatorios.
- \* Empieza a notarse diferencias entre organización física y lógica.
- \* Procesamiento en Batch, en línea y en tiempo real.
- \* No se usa recuperación por múltiples llaves.
- \* Empieza a usarse medidas de seguridad en los datos.
- \* No existe manipulación de datos, sólo métodos de acceso.
- \* Continúa existiendo la redundancia.

- \* Se permiten estructuras jerárquicas, pero el usuario las tiene que construir.
- \* Los datos siguen siendo diseñados para una sola aplicación.

### **Etapa 3.- Nacimiento de los Sistemas de Bases de Datos (inicio de los 70's)**

- \* Surge el DBMS. (DATA BASE MANAGER SYSTEM)<sup>2</sup>
- \* Se puede derivar varios archivos lógicos a partir de los mismos datos físicos.
- \* Los mismos datos pueden ser accedados de diferentes maneras.
- \* Disminuye la redundancia.
- \* Se comparten los datos entre distintas aplicaciones.
- \* Se facilita la integridad de la información.
- \* El almacenamiento físico es independiente de las aplicaciones.
- \* Se permite recuperación por múltiples llaves.
- \* Se permiten estructuras de redes.

### **Etapa 4.- Sistemas de Bases de Datos (finales de los 70's)**

- \* Se permite la independencia de datos físicos y lógicos.
- \* La base de datos puede crecer sin incurrir en costos de mantenimiento.
- \* Se dan facilidades al DBA (administrador de bases de datos) para la administración de y control de la base de datos y usuarios.
- \* Surgen métodos efectivos para la seguridad, privacidad e integridad de los datos.

---

<sup>2</sup> Manejador de Sistema de Bases de Datos.

- \* Se usan los archivos invertidos (puede encontrar entidades a partir de atributos).
- \* Las bases de datos son usadas para responder preguntas aun desconocidas en el momento de diseño.
- \* Se permite la migración de datos (transferencia de información).
- \* Existen lenguajes para la declaración y manipulación de datos.
- \* Tiene auge el modelo de redes y jerárquico, no así el modelo relacional.
- \* Los recursos computacionales (RAM, Disco Duro) son limitados.

#### **Etapa 5.- Bases de Datos Multimedia (finales de los 80's)**

- \* Al ser los recursos computacionales (RAM, Disco Duro) un aspecto secundario, los DBMS's y DBA's se hacen más eficientes y eficaces.
- \* Manejo de datos en RAM evitando ir a disco.
- \* Facilidad para el manejo de Bases de Datos no sólo de datos sino de imágenes, texto, gráficas y sonido.
- \* Aumenta la transportabilidad no sólo en los datos sino también en las aplicaciones.
- \* Inicia el auge del modelo relacional y viene en decadencia lo de redes y jerárquicos.
- \* Apoyo hacia los sistemas abiertos permitiendo todas estas características.

### **I.2.2. Definición de una Base de Datos.**

Después de tener toda la información y realizar un análisis de la misma, el paso siguiente es la definición de los archivos o bases de datos a utilizar dentro del sistema. A continuación se hace mención de lo que significa esto.

Como ya hemos dicho el procesamiento de la información ha constituido una de las tareas básicas de cualquier organización y la base de datos es el componente estructural clave en el diseño de sistemas de información.

Es la principal fuerza de integración del sistema de información de una organización. Debe lograrse un ajuste entre las necesidades de procesamiento y de toma de decisiones de la organización y la estructura y composición de la base de datos. Si no se logra esto, se desperdiciarán los esfuerzos del analista de sistemas.

*" Una base de datos es un conjunto de elementos de datos organizados de tal manera que cubran los requerimientos de información de los usuarios".*

Un sistema de manejo de bases de datos (DBMS) es un conjunto de archivos interrelacionados y una serie de programas que permiten a los usuarios tener acceso a estos archivos y modificarlos. Uno de los objetivos principales de un sistema de base de datos es proporcionar a los usuarios una visión abstracta de la información. Es decir, el sistema oculta ciertos detalles relativos a la forma como los datos se almacenan y mantienen. Sin embargo, para que el sistema sea útil, la información debe recuperarse en forma eficiente. La búsqueda de la eficiencia conduce a la estructura de la base de datos complejas para representar la información en la base de datos. Pero como estos muchas veces son utilizados por personal que no cuenta con conocimientos de computación, esta complejidad debe estar escondida para los usuarios. Para ocultarla, se definen varios niveles de abstracción en los que puede observarse la base de datos.

### 1.2.3. Niveles de Abstracción de la Base de Datos.

Los niveles de abstracción que se pueden observar en una base de datos son:

**Nivel Físico:** Es el nivel más bajo de abstracción, en el que se describe como se almacena realmente los datos. En este nivel se describen al detalle las estructuras de datos complejas del nivel más bajo.

**Nivel Conceptual:** Este es el siguiente nivel más alto de abstracción, en el que se describe cuales son los datos reales almacenados en la base de datos y que existen entre los datos. Este nivel contiene todas la base de datos en términos de unas cuantas estructuras relativamente sencillas. Aunque es posible que la implantación de las estructuras simples de nivel conceptual requiere estructuras complejas en el nivel físico, no es forzoso que el usuario de nivel conceptual se de cuenta de ello. El nivel conceptual de abstracción lo utilizan los administradores de las bases de datos; quienes deciden que información se guarda en la base de datos.

**Nivel de Visión:** Este es el nivel de visión más alto, en el cual se describe solamente una parte de la base de datos. Aunque en el nivel conceptual se utilizan estructuras más simples, todavía queda una forma de complejidad que resulta del gran tamaño de la base de datos. Muchos usuarios de la base de datos no tendrán que ocuparse de toda esta información. Más bien, necesitarán solamente una parte mejor de la base de datos. Para simplificar la interacción entre estos usuarios y el sistema, se define el nivel de abstracción de visión. El sistema puede proporcionar muchas vistas diferentes de la misma base de datos.

En el nivel físico, un registro de **cuentahabiente**, cuenta o empleado puede describirse como un bloque de localidades de almacenamiento consecutivas (por ejemplo, palabras o bytes). En el nivel conceptual, cada uno de estos registros se determina por registro **cuentahabiente**, y se describe la interrelación entre los tres tipos de registro. Finalmente en el nivel de visión, se definen varias vistas de la base de datos, que en este caso son cada uno de los registros **cuentahabiente**, **cuenta** y **empleado**.

#### 1.2.4. Estructuras o Modelos de Datos.

Para describir la estructura de una base de datos es necesario describir el concepto de modelos de datos. Este es un grupo de herramientas conceptuales de para describir los datos, sus relaciones, su semántica y sus limitantes. Los modelos de datos diferentes se pueden dividir en tres grupos: Los modelo lógicos basados en objetos y en registros, y los modelos y los modelos físicos de datos.

**Modelos Lógicos basados en objetos:** Estos se utilizan para describir los conceptos en los niveles conceptual y de visión. Se caracterizan por el hecho de que permiten una estructuración bastante flexible y hacen posible especificar claramente las limitantes de los datos. Algunos modelos de los más conocidos son:

- El modelo entidad-relación
- El modelo binario
- El modelo semántico de datos
- El modelo infológico

Se ha escogido el modelo entidad-relación como representativo de la clase de modelos basados en objetos ya que es apropiado para el diseño de bases de datos y porque se utiliza ampliamente en la practica.

El modelo entidad-relación (E-R) se basa en una percepción de un mundo real que consiste en un conjunto de objetos básicos llamados entidades, y de las relaciones entre estos objetos. Una entidad es un objeto que existe y puede distinguirse de otros. La distinción se logra asociando a cada entidad un conjunto de atributos que describen al objeto. Por ejemplo, Cliente asocia a un cliente con cada una de las cuentas que tiene. El conjunto de todas las entidades y relaciones del mismo tipo se denominan conjunto de entidades y conjunto de relaciones, respectivamente.

Además de entidades y relaciones el modelo E-R representa ciertas limitantes que debe cumplir el contenido de una base de datos. Una de estas limitantes es la cardinalidad de mapeo, que quiere decir, el número de entidades con las que puede asociarse otra entidad por medio de un conjunto de relaciones.

### **Modelos Lógicos basados en registros:**

Los modelos lógicos basados en registros se utilizan para describir los datos en los niveles conceptual y de visión, estos modelos sirven para especificar tanto la estructura general de la base de datos como una descripción en un nivel más alto de la implementación.

### **I.2.5. Tipos de Bases de Datos.**

#### **Base de Datos del tipo Relacional:**

Los datos y las relaciones entre los datos se representan por medio de una serie de tablas, cada una de las cuales tiene varias columnas con nombres únicos.

Las tablas presentan las siguientes propiedades:

- 1.- Cada entrada de la tabla representa un campo de datos
- 2.- Cada columna contiene valores acerca del mismo atributo y cada columna está asignada a un nombre único.
- 3.- Cada fila es distinta, es decir, una fila no puede duplicar a otra fila en columnas seleccionadas de atributos.
- 4.- Tanto las filas como las columnas pueden verse en secuencia, en cualquier momento, sin afectar el contenido de información de la relación.

Se puede utilizar una notación abreviada para representar abstractamente las tablas de relacionales. Por ejemplo:

**CLASE PROGRAMADA** (Número de clases, Sección, Número de Docente, Inscripción, Número de Salón, Día de Clase, Hora de Clase)

**MIEMBRO DOCENTE** (Número de Docente, Nombre del Docente, Posición, Nivel Académico)

**ESTUDIANTE** (Número del Estudiante, Nombre del Estudiante, Posición en la clase)

**ESTUDIANTE-CLASE** (Número del estudiante, Número de la clase, Sección)

**SALON** (Número del Salón, Descripción del Salón, Capacidad total, Tomas Eléctricas, Arreglo de Asientos).

El nombre de la entidad esta a la izquierda del primer paréntesis y se separa con comas. La llave primaria para cada relación se subraya. Las llaves primarias deben ser atributos únicos empleados para diferenciar los valores de datos dentro de una entidad.

Una forma de determinar cuales atributos pueden servir como llaves primarias consiste simplemente en buscar entre los atributos aquel o aquellos que hagan única a cada fila.

El empleo frecuente de un modelo relacional requiere agregar entidades y atributos adicionales a la base de datos.

El modelo relacional es capaz de mostrar relaciones complicadas de datos similar a la capacidad de datos de una red compleja. Sin embargo, en comparación con el modelo de red, es más fácil de entender y diagramar.

#### **Base de Datos del tipo de Red**

Los datos en el modelo de red se representan por medio de conjuntos de registros y las relaciones entre los datos se representan con ligas que puedan considerarse como apuntadores. Los registros de la base de datos se organizan en forma de conjuntos de gráficas arbitrarias.

#### **Base de Datos del Tipo Jerárquico**

El modelo jerárquico es similar al modelo de red en cuanto a que los datos y las relaciones entre los datos se representan por medio de registros y ligas, respectivamente. El modelo jerárquico difiere de red en que los registros están organizados como un conjunto de árboles en vez de gráficas arbitrarias.

### 1.2.6. ESTRUCTURA GENERAL DE UN SISTEMA DE BASE DE DATOS.

Un sistema de base de datos se divide en módulos que se encarga de cada una de las tareas de sistema general. Algunas de las funciones del sistema de base de datos pueden ser realizadas por el sistema operativo. En la mayor parte de los casos, el sistema operativo proporciona únicamente los servicios más elementales y la base de datos debe de partir de ese fundamento. Así el diseño de la base de datos debe considerar una interfaz del sistema operativo.

#### Lenguaje de Definición de Datos

Un esquema de bases de datos se especifica por medio de una serie de definiciones que se expresen en un lenguaje especial llamado lenguaje de definición de datos (DDL, Data Definition Language). El resultado de la compilación de las proposiciones en el DDL es un conjunto de tablas que se almacenan en un archivo especial llamado diccionario (o directorio de datos).

Un directorio de datos es un archivo que contiene datos de los mismos datos. Este archivo se consulta antes de leer o modificar los datos reales en el sistema de base de datos.

La estructura de almacenamiento y los métodos de acceso empleados por el sistema de bases de datos se especifican por medio de un conjunto de definiciones de tipo especial del DDL, llamado lenguaje de almacenamiento y definición de datos.

#### Lenguaje de Manejo de Datos.

Un lenguaje de manejo de datos (DML, Data Manipulation Language) permite a los usuarios manejar o tener acceso a los datos que estén organizados por medio del modelo adecuado. Existen básicamente dos tipos de DML:

De procedimientos: Estos necesitan que el usuario especifique cuales datos quiere y como deben obtenerse.

Sin procedimientos, requiere que el usuario especifique cuales datos quiere sin especificar como obtenerlos.

Los DML sin procedimientos son más fáciles de aprender y usar que los de procedimientos.

### **Administrador de la Base de Datos.**

Una de las razones principales para contar con sistemas de base de datos es tener un control centralizado tanto de los datos como de los programas que tiene acceso a ellos, la persona que tiene este control centralizado es el administrador de base de datos (DBA) Las funciones de este son:

Definición de esquema.

Definición de la estructura de almacenamiento y del método de acceso.

Modificaciones del esquema y de la organización física.

Concesión de autorización a los datos.

Estos son algunos de los conceptos de base de datos que por lo general se manejan a un nivel interno, es decir, desde el punto de vista de datos.

#### **I.2.7. ENFOQUES PARA EL ANALISIS DE DATOS.**

Se pueden adoptar dos enfoques para el análisis de los datos: un enfoque orientado a los procesos o un enfoque orientado a la información.

El enfoque orientado a los procesos se concentra en una salida definida previamente y en los requerimientos básicos de procesamiento de datos de la organización. Un enfoque orientado a la información se concentra en los elementos de datos que están almacenados en la base de datos y en sus relaciones y enlaces para proporcionar una diversidad de requerimientos de información específicos y aún no conocidos. Con frecuencia se emplean ambos enfoques para satisfacer los requerimientos de todos los usuarios debido que a menudo un proyecto de sistemas contiene aspectos de ambos enfoques.

FALLA DE ORIGEN

*" Los más cuidadosos esquemas de ratones y de Hombres, se rigen por el azar."  
Robert Bruns, a un ratón. (1787)*

## **CAPITULO II.- ESTRUCTURA Y ANALISIS DEL PROCESO DE INSCRIPCION**

### **II.1. FUNCIONES DEL DEPARTAMENTO**

### **II.2. ANALISIS DEL PROCESO ACTUAL DE REINSCRIPCION**

- II.2.1. Proceso de Reinscripción (manual)**
- II.2.2. Proceso de ALTAS, BAJAS y CAMBIOS.**
- II.2.3. Procesamiento de la información generada por reinscripción**
- II.2.4. Desventajas del proceso de Reinscripción manual.**
- II.2.5. Planteamiento del problema y soluciones.**

**OBJETIVO: ANALIZAR EL PROCESO DE REINSCRIPCION EN EL DEPARTAMENTO DE SERVICIOS ESCOLARES, ASI COMO ESTRUCTURAR LA INFORMACION ADQUIRIDA PARA EL DESARROLLO DE LAS REINSCRIPCIONES AUTOMATIZADAS.**

## II.1. FUNCIONES DEL DEPARTAMENTO DE SERVICIOS ESCOLARES

El departamento de Servicios Escolares cumple con varias funciones. Estas funciones son de gran importancia, ya que apoyan al alumno en la realización de sus trámites escolares.

El departamento de Servicio Escolares esta dividido en cinco secciones:

- Sección de Actas
- Sección de Cómputo
- Sección de Egresados
- Sección de Revisión de Estudios
- Sección de Ventanillas

La sección de actas coordina todo lo relacionado con las actas de calificaciones como distribución, elaboración, envío, etc.

La sección de cómputo es responsable de automatizar los diferentes actividades del departamento de servicios escolares. Por ejemplo: el proceso de ajuste (ALTAS, BAJAS Y CAMBIOS), el registro de exámenes extraordinarios computarizados, etc.

La sección de Egresados se encarga de realizar los trámites de los alumnos que han cumplido con el cien por ciento que marca su plan de estudios como: constancias, certificados y todo lo relacionado con la realización de su examen profesional.

A la sección de revisión de estudios se le pide que verifique todas y cada una de las calificaciones obtenidas por los alumnos con el propósito de certificar que cumplan con los planes de estudio.

En la sección de ventanillas se atienden a los alumnos inscritos en las diferentes carreras, en la cual los alumnos realizan trámites de inscripciones y/o reinscripción; constancias de inscripción, de créditos y promedio, resello de credenciales, registro de exámenes extraordinarios, entrega de Historias Académicas, etc.

Uno de los trámites más importante dentro de este departamento es el de las reinscripciones de los alumnos a los semestres siguientes. Así como el proceso de ajuste (ALTAS, BAJAS Y CAMBIOS) y registro de exámenes extraordinarios.

Por lo tanto se pretende que esta actividad este totalmente automatizada.

## **II.2. ANALISIS DEL PROCESO ACTUAL DE REINSCRIPCION.**

Actualmente el proceso de reinscripciones de algunas carreras todavía se realiza de manera manual. Es decir, los alumnos tienen que llenar sus hojas ópticas y esperar turno para reinscribirse. Este proceso cambia un poco dependiendo de la carrera que se trate.

Las carreras de Arquitectura, Pedagogía, Economía, Diseño Industrial y Planificación para el Desarrollo Agropecuario todavía realizan este proceso en forma manual.

### **II.2.1. Proceso de Reinscripción manual.**

Las carreras de Arquitectura, Pedagogía, Economía, Diseño Industrial y Planificación para el Desarrollo Agropecuario realizan los trámites de reinscripción en coordinación con su jefatura, y el departamento de Servicios Escolares.

El proceso que siguen para realizar este trámite por lo general es muy similar y es en términos generales de la siguiente manera:

Se determina una fecha de reinscripción, así como una fecha para entrega de papelería. Posteriormente los alumnos pasan al área correspondiente para realizar su reinscripción.

Esta se realiza de tal manera que los alumnos pueden verificar los grupos saturados y el turno correspondiente, así como clave de materia y grupos. Para hacer esto, hay personas que evalúan la información del alumno antes de entregarle su hoja óptica; ya que tienen esto, pasan a llenar su hoja óptica y después la entregan a la persona correspondiente, verificando otra vez que sus códigos de materias y grupos estén correctos.

Este proceso es muy tardado y requiere mucha paciencia por parte del alumno.

En estas carreras como la población de alumnos no es tan elevada, como en las carreras de Derecho e Ingeniería; el proceso es un poco más rápido. Pero esto no implica que sea ineficiente y complicado.

Un ejemplo de un proceso de reinscripción es el de la carrera de Pedagogía que a continuación se describe:

- 1.- Los alumnos se presentan en la hora y fecha señalados por un número de sorteo, en el área que le corresponde, se forman y van entrando en un salón de 10 en 10. En la puerta hay una persona que verifica su número de sorteo. (En este proceso se emplean aproximadamente de 1 a 2 minutos).
- 2.- Después pasan a las mesas donde hay de cinco a seis personas. Las tres primeras verifican el semestre y los grupos de las materias a cursar por el alumno. También verifican el cupo por grupo; en este caso tienen un límite de 60 alumnos máximo. Los alumnos deben de traer su tira de materias provisional, ya con sus grupos y materias a cursar. Esta información es la que se verifica.

Las siguientes dos personas tienen las hojas ópticas de los alumnos de la carrera ordenadas por semestre y alfabéticamente. Los alumnos pasan a que se les entregue está y realicen el llenado de la misma. Aproximadamente, en este paso se tardan de tres a cinco minutos.

- 3.- Ya que tienen su hoja óptica, lo siguiente es que la llenen y la entreguen ese mismo día. Este proceso se de 20 a 25 minutos dependiendo del alumno y las materias que vaya a cursar.
- 4.- Por último entregan su hoja óptica a otra persona. Esta revisa que las claves de materias y grupos estén correctos, así como sus datos anotados en la hoja óptica. En realizar este proceso, el alumno pierde de 2 a 3 minutos.

En total, el promedio de tiempo que emplea el alumno en realizar este trámite es de 30 a 45 minutos.

Las demás carreras que aquí se analizan llevan casi el mismo procedimiento y el tiempo que se pierde es muy similar.

#### II.2.2. Proceso de Ajuste (ALTAS, BAJAS Y CAMBIOS).

El proceso de Ajuste se realiza dos semanas después de las reinscripciones. Para realizar este proceso, los alumnos tienen que pasar a su ventanilla correspondiente pedir la hoja para este trámite, llenarla, entregarla y regresar por la respuesta al siguiente día.

Servicios Escolares hace la evaluación de los ajustes solicitados por los alumnos, para lo cual los hace esperar un día. Ya que debe verificar que los grupos tengan cupo e irlos actualizando con los alumnos que se dan de baja en esos mismos grupos o en otros.

Los movimientos que puede realizar el alumno son : ALTA, BAJA y CAMBIO de grupo.

Cuando es baja de grupo, el movimiento es aceptado y se actualiza el cupo en ese grupo. Si es una Alta de grupo, se verifica que el grupo tenga lugar, si es así, se acepta el movimiento; actualizando el cupo del mismo. En dado caso de que no haya cupo, que sea grupo saturado, el movimiento no es aceptado. Y cuando es un Cambio de grupo, se verifica también que el grupo tenga lugar disponible, si hay lugar, se da de baja y se da de alta en el grupo que pida el alumno. Actualizandose ambos grupos.

Después de ser evaluadas las solicitudes de los alumnos, se procede a dar la hoja óptica otra vez a los alumnos y continuar con el trámite, en caso contrario, se le explica al alumno las causas por las que fue rechazada su petición.

### **II.2.3. Procesamiento de la Información generada por la Reinscripción.**

Al termino de las reinscripciones, la información obtenida (hojas ópticas) es recopilada y acomodada para ser enviada a CU.

Esta información es leída por la lectora óptica y es regresada en un diagnóstico, el cual contiene la información en forma condensada del trámite realizado por los alumnos.

Las hojas ópticas son almacenadas para su verificación cuando es necesario. Esto ocurre con las carreras que no se encuentran automatizadas en este caso: Arquitectura, Diseño industrial, Economía, Pedagogía y Planificación para el Desarrollo Agropecuario.

### **II.2.4. Desventajas del proceso manual de Reinscripción.**

El procedimiento de reinscripción siempre ha sido manual pero con la introducción de las computadoras, se ha requerido que este proceso sea automatizado.

A través de la automatización, se pretende que las reinscripciones en las carreras mencionadas anteriormente tengan un mejor control y una mayor eficiencia en cuanto a la información obtenida.

Las desventajas más comunes que se han observado en la realización de este trámite son:

- 1.- El tiempo que los alumnos emplean en la realización de este trámite.
- 2.- El incorrecto llenado de las hojas ópticas. La gran cantidad de errores por parte de los alumnos.

- 3.- La saturación de los grupos debido al mal manejo de la información.

Una vez que se ha presentado la saturación en algunos grupos, los alumnos acuden con el jefe de la carrera para solicitar que les sea autorizada su inscripción, este tipo de situación provoca una sobresaturación en algunos grupos y pocos alumnos en otras asignaturas.

- 4.- Como la información de las hojas ópticas hay que enviarla a C.U., provoca una perdida tiempo.
- 5.- La falta de instalaciones adecuados hace que se le brinde un mal servicio a los alumnos.

Estas son algunas de las desventajas que se presentan en forma general. Pero cada carrera tiene en especial algunas desventajas que necesitan sean superadas.

#### II.2.5. Planteamiento del Problema y Soluciones.

Para poder desarrollar el sistema de reinscripciones, antes que nada hay que tomar en cuenta la problemática del flujo de la información.

Anteriormente se mencionaron algunos de los problemas o desventajas que ocasiona la reinscripción en forma manual.

La solución más adecuada para cubrir las necesidades del usuario tanto en atención al alumno como en el manejo de la información es un sistema que cumpla con las siguientes características:

Que permita acceder la información correspondiente de cada uno de los alumnos mediante el uso de su número de cuenta como llave. De tal manera que se pueda consultar la situación escolar, solamente accediendo dicho número.

Será capaz de detectar algunos factores que aún cuando no se manejen en el área de servicios escolares, se les pueda apoyar para mejorar el control de todos los procesos, algunos de estos son:

- El alumno no debe tener adeudo de libros en la biblioteca.
- Que no haya incurrido en situaciones que lo haga acreedor a alguna sanción.
- Será necesario establecer las reinscripciones por medio de sorteo.

Por lo tanto cuando se despliegue en pantalla la situación del alumno se deberá considerar la fecha y la hora que le fue asignada para su reinscripción, de acuerdo con la información generada por los números de sorteo.

El sistema hará también una identificación de los alumnos que realizaron ya el trámite de reinscripción, para poder detectar posibles casos de reinscripciones duplicadas.

Además de realizar la verificación de las claves de asignaturas que puede cursar el alumno, de tal forma que solamente si la asignatura corresponde a su plan de estudios, el nombre se desplegará en la pantalla, de lo contrario se debe rechazar, esto para evitar la inscripción a asignaturas que: pueden no ser parte del plan de estudios y que además el alumno no desea cursar.

Se llevará un control adecuado de la capacidad de los grupos, con el fin de lograr un balance adecuado en el número de alumnos inscritos que existe en cada grupo.

Finalmente se emitirá un comprobante de reinscripción computarizado, que el alumno no pueda falsificar o alterar.

Este es un procedimiento general para todas las carreras automatizadas. Sin embargo fue necesario acudir a las coordinaciones de las carreras faltantes; en este caso, Arquitectura, Diseño Industrial, Pedagogía, Economía y Planificación para el Desarrollo Agropecuario, para analizar las necesidades específicas de cada una y llevar un mejor control y organización del sistema a implementar.

#### **NOTA CONCEPTUAL:**

Está es la fase de definición. Esta parte se centra sobre el qué. Esto es, durante la definición, el que desarrolla el software intenta identificar qué información ha de ser procesada, qué función y rendimiento se desea, que interfases han de establecerse, que restricciones de diseño existen y qué criterios de validación se necesitan para definir un sistema correcto. Por tanto, han de identificarse los requisitos clave del sistema y del software. Aunque los métodos aplicados durante la fase de definición variarán, por lo regular, se producirán tres pasos específicos:

**Análisis del Sistema.** El análisis define el papel de cada elemento de un sistema informático, asignando finalmente al software el papel que va a desempeñar.

**Planificación del proyecto de software.** Una vez establecido el ámbito del software, se analizan riesgos, se asignan los recursos, se estiman los costes, se definen las tareas y se planifica el trabajo.

**Análisis de requisitos.** El ámbito establecido para el software proporciona la dirección a seguir, pero antes de comenzar a trabajar es necesario disponer de una información más detallada del ámbito de información y de función del software.

*El orden y conexión de las ideas es lo mismo que el orden y conexión de las cosas.*

*Baruch (Benito de) Espinosa.*

*Prob. VII de la Etica, parte dos.*

### **CAPITULO III.- IMPLEMENTACION DE LA RED**

#### **III.1. INTRODUCCION A LAS REDES**

**III.1.1. Estratos Fundamentales de Arquitecturas.**

**III.1.2. Niveles de Interconexión.**

**III.1.3. Redes Locales (LAN NETWORK)**

**III.1.4. Características de Operación de los tres tipos de Redes LAN**

#### **III.2. DESCRIPCION DE LA RED NOVELL (LAN) 3.11**

**III.2.1. Introducción a las Redes NOVELL**

**III.2.2. Componentes de una Red NOVELL**

**III.2.3. Estructura de la Red NOVELL**

#### **III.3. ARQUITECTURA DE LA RED NOVELL**

**III.4. IMPLEMENTACION DE LA BASE DE DATOS**

**OBJETIVO: DESCRIBIR LOS CONCEPTOS BASICOS DE REDES, ASI COMO LA RED DE AREA LOCAL NETWARE Y LA INTEGRACION DE LAS BASES DE DATOS EN RED DE LAS CARRERAS QUE FALTAN AUTOMATIZAR.**

Este capítulo describirá brevemente algunas características de las redes, en especial la Red LAN NOVELL, ya que es el sistema que se tiene integrado en el departamento de Servicios Escolares.

### III.1. INTRODUCCION A LAS REDES

#### Necesidad de una Arquitectura.

Existen cuatro tendencias fundamentales presentes en el mundo del procesamiento y la comunicación de datos:

- La necesidad de protocolos de comunicación de datos que trasciendan los límites de los anteriores protocolos.
- La capacidad de trasladar inteligencia a dispositivos cada vez más pequeños.
- El desarrollo de las comunicaciones entre clases más nuevas de dispositivos.
- La necesidad de interconectarse con medios más nuevos de transportadoras comunes.

De este modo, cualquier producto a desarrollarse debe ser lo suficientemente versátil como para adaptarse no solamente al nuevo ambiente sino también ajustarse a nuevos factores que surjan en el futuro. Esto significa que cada producto debe cumplir con una serie de estándares que posibiliten que los dispositivos se entremezclen de diversas maneras. En otras palabras, se necesita de un PLAN MAESTRO que sea independiente de los dispositivos. Este plan deberá constar de todos los protocolos de comunicación permitidos entre varios tipos de equipos. No se puede diseñar un dispositivo, tan exclusivo, que obligue a crear componentes físicos y software especiales para comunicarse con él. Un plan maestro o arquitectura haría entonces posible utilizar diversos productos en diferentes redes de clientes, en diferentes partes de la misma red, y/o en diferentes momentos en las mismas partes de la red. Los productos y los sistemas, de los usuarios, deben contar con la posibilidad de ser trasladados hacia y desde las instalaciones específicas, con un mínimo de modificaciones al producto de la transmisión.

### Propósito de las arquitecturas

Una arquitectura de comunicaciones debe ser el vínculo de unión para todos los productos de comunicación de datos, tanto existentes, como futuros de un vendedor. La arquitectura logra esto especificando rigurosamente:

- Los conceptos y estructuras lógicas involucrados.
- Una serie de reglas y pautas referentes a la interconexión
- Una serie de configuraciones de red posibles.

Estos tres aspectos de la arquitectura, al combinarse de diferentes maneras, especifican un producto en particular. De este modo, la arquitectura asegura que todos los productos de comunicación relacionados, trabajen juntos en una forma consistente y compatible.

Con la complejidad en aumento de la filosofía de comunicación, se hace cada vez más necesario ocultar al usuario y a los programadores de aplicación asociados, ciertas complejidades. Estas aplicaciones deben tener la capacidad de adquirir datos de diversas fuentes y de proporcionar datos a muchos usuarios de sistemas de comunicación sin conocimiento sobre la ubicación física y sus características especiales. Esto es particularmente cierto en el caso de redes a las que conectan múltiples tipos de dispositivos. La arquitectura oculta características tales como control de dispositivos, formatos de código y restricciones de almacenamiento intermedio, proporcionando interconexiones de alto nivel y orientadas a mensajes.

Una arquitectura debe brindar una filosofía de control de red que integre completamente al control dentro de la red, de manera que una porción de red del sistema de comunicación no dependa de otros componentes para mantenerse en operación. También deben ser inherentes a la arquitectura las disposiciones para la alta eficiencia, redes de bajos tiempos adicionales, redes ultra resistentes y redes en las cuales los datos estén seguros.

**Una arquitectura de comunicaciones es:**

*Una estrategia comercial para solucionar problemas de compatibilidad en línea de productos mediante un conjunto de especificaciones técnicas que definen las interrelaciones entre las partes de sistema y la distribución de funciones entre los componentes de la red.*

Además tiene como propósito fundamental estandarizar y coordinar el desarrollo entre muchos grupos de trabajo, para producir un resultado armonioso, que vincule a una familia de productos (implementación), en donde estos se complementen en la forma buscada.

### III.1.1. ESTRATOS FUNDAMENTALES DE LAS ARQUITECTURAS

Una característica común a las arquitecturas, es la de un estructura en capas en cada nodo de la red. Las capas básicas de protocolo consisten de transmisión, administración de servicios y funciones de aplicación.

#### **Funciones de Transmisión**

La capa de transmisión se ocupa del ruteo y movimiento de datos entre el origen y el destino. Esta capa maneja el camino y las conexiones a nivel de origen de enlace. El que los datos sean transportados a través de cables, microondas, o satélites es intrascendente, mientras sean entregados sin alteración en el destino apropiado. Nuevas técnicas de transmisión, tales como conmutación de paquetes, requieren algún formato específico de los datos. El protocolo de transmisión es responsable de esta función. Esta capa debe incluir también un control de error para asegurar que los datos se entreguen correctamente, y si no es así, para corregirlos o retransmitirlos.

#### **Servicios**

La segunda capa o capa de "administración de servicios" proporciona transformaciones específicas según el dispositivo: de las características del emisor a las necesidades del receptor.

Esta capa es clave para la confección de redes, ya que proporciona las funciones de traducción entre las características de terminales físicas y programas de aplicación. La capa de administración de servicios tiene también otros dos papeles fundamentales: control de la red y control de sesión. El control de sesión vigila el estado de la sesión, coordinando la activación/desactivación de una sección y brindando apoyo para comenzar, borrar y resincronizar los flujos de datos relacionados con la sesión.

### Aplicaciones

La capa de aplicaciones se ocupa de las funciones del usuario final y puede ser un programa de aplicación o un dispositivo de Entrada/Salida, tal como una terminal. Los servicios de la capa de administración de servicios se invocan por pedidos, desde la capa de aplicación. En la computadora, la capa de aplicación consiste de programas de aplicación, de los cuales el usuario de la terminal solicita información a través del procesamiento de los datos. En la terminal, la capa de aplicación está representada por el operador de la terminal o un programa de aplicación en una unidad de control programable. A estos orígenes y destinos de información se les llama "usuarios finales" y también se incluyen en este grupo diversos medios físicos de almacenamiento, tales como cintas y discos magnéticos.

### III.1.2. NIVELES DE INTERCONEXION

Otra característica común a las arquitecturas de redes de computadores es la de tener tres niveles de conexión dentro de la red.

El primer nivel más básico, es el *enlace entre dos nodos*. Este enlace podría ser una única línea de comunicación que interconecta dos equipos, o una línea que une un dispositivo con varios, administrada por un procesador de comunicaciones, un concentrador remoto o un controlador inteligente.

El segundo nivel de conexión dentro de la red es el que se encuentra entre el nodo origen y el nodo de destino, y se le llama *vía extremo a extremo*. Esta es una conexión lógica o grupo de reglas de ruteo para trasladar datos desde el nodo de entrada hasta el nodo destino.

El tercer nivel de conexión es el que existe entre usuarios finales apareados. Estos no siempre son personas. Por otro lado, son siempre origen y destino de datos. Uno de los usuarios finales apareados será generalmente una persona en un teclado, pero puede ser una impresora o pantalla. El otro miembro del par será a menudo un programa de aplicación que recibe la entrada y/o envía la salida al primero. Sin embargo, ambos podrían ser programas de aplicación. Generalmente se considera que el usuario final está fuera de la arquitectura de la red.

### III.1.3. REDES LOCALES (LAN: LOCAL AREA DE NETWORK)

#### La red de comunicaciones del área local

La red de área local puede estar integrada de varios mecanismos de entrada/salida y transferencia de información.

**Procesamiento de imágenes.** La entrada en forma discretizada del resultado del examen electrónico de documentos; el archivo de documentos en imágenes en una base de datos y el procesamiento de estos documentos con intervención del hombre, a través de la red, de emisiones intensificadas en imágenes.

**Distribución de Documentos.** La transferencia de alto volumen de facsímiles y/o imágenes discretizadas entre centros de distribución de documentos y la subsiguiente distribución, parcialmente electrónica y parcialmente manual, desde estos centros de distribución a sus receptores. Un subgrupo de esta clase de aplicación es el correo electrónico, que trata de usar técnicas similares a los sistemas normales de correo.

**La oficina automatizada.** La captación electrónica de golpes a teclas de máquinas de escribir electrónicas ( u otros medios de entrada) y el uso de comunicaciones y procesamiento de datos para apoyar las funciones de registro, archivo, recuperación, copia y distribución del complejo de la oficina.

**Información Doméstica.** La fusión de pequeños procesadores, máquina de escribir y televisores permitirá a las personas acceder a servicios antes no disponibles en el hogar. Los datos disponibles serán tanto personales como comerciales. Los datos personales consistirán en hechos entrados por el usuario y almacenados localmente, tales como recibos, datos sobre impuestos, información financiera y proyectos personales.

Estos nuevos dispositivos serán parte de un sistema que combinará la función de transmisión de datos, la función de transmisión por video y la función de comunicación por la voz en un sistema de comunicación común en caso, conocido como la RED LOCAL. Estas redes locales proporcionan los enlaces de transmisión dentro de un edificio grande o un grupo de edificios cercanos, manejando aproximadamente dos tercios del volumen total de las necesidades de comunicación de grandes organizaciones.

Estas redes locales habitualmente usan una arquitectura basada en cables seriales para unir computadores, periféricos, terminales de datos y equipos de oficina. Ofrecen una red de componentes barata, capaz de operar a velocidades entre 100 Kilobits/seg y 500 Megabits/seg, con márgenes de error de  $10^{-9}$  a  $10^{-19}$  (en bits).

*Una red local se define como un sistema de comunicación intra-oficina, intra-edificio, intra-servicios, que apoya algún tipo de procesamiento de comunicaciones y transferencia de información transparente entre usuarios y/o dispositivos electrónicos.*

Hay dos cosas que una red local no es: no es una red local geográfica dispersa (esta red de largo alcance), tampoco es un computador multiprocesor.

La mayoría de las redes locales emplean un medio coherente para las comunicaciones. Los medios incluyen pares de cables trenzados, cable coaxial y fibra óptica. Otros usan técnicas de transmisión inalámbrica tales como subcanales de frecuencia de radio e infra-rojos.

También requieren de algún tipo de interconexión lógica entre los sistemas conectados y la red.

Por definición, la red local avanzada debe incluir alguna clase de salida al exterior, a otra red local, a una red intra-oficinas, a un computador principal o una red transportadora común de largo alcance.

#### III.1.4. CARACTERISTICAS DE LAS REDES LOCALES

Las redes locales se caracterizan por:

- 1.- *Protocolo de comunicación (método de acceso)*
- 2.- *Topología*
- 3.- *Medio de transmisión*

##### 1. PROTOCOLOS

Los nodos en una red deben comunicarse. La manera en que se controla y administra la comunicación se conoce como protocolo de acceso.

*El protocolo de comunicación es la forma como se envía la información por medio diversos estandares establecidos.*

Existen tres Protocolos de acceso básico para redes locales

##### **CSMA/CD(Carrier Sense Multiple Access/Collision Detection)**

En este protocolo de acceso un mensaje se transmite por cualquier estación o nodo de la red en cualquier momento, mientras la línea de comunicación se encuentra sin tráfico.

CSMA/CD es un protocolo basado en un esquema de detección de colisiones en donde, el primer mensaje enviado es el primer en ser atendido.

Cuando dos o más nodos transmiten simultáneamente ocurren colisiones y entonces, el proceso se repite hasta que la transmisión es exitosa. Debido a que entre más transmisiones se intenten más colisiones pueden ocurrir, los tiempos de respuesta son inconsistentes e impredecibles.

### **Token Passing**

Este protocolo está basado en un esquema libre de colisiones. El TOKEN (señal) se pasa de un nodo o estación de la red al siguiente nodo, independientemente de si ese nodo necesite transmitir o no. Cada estación cuenta con un tiempo para transmitir idéntico al de las demás estaciones y sólo puede transmitir su mensaje cuando tiene el TOKEN.

En este método de acceso la línea de comunicación siempre está libre para transmitir mensajes por lo que se pueden tener tiempos de respuesta predecibles aún con gran actividad en la red.

### **Protocolo de Poleo**

Este método de acceso se caracteriza por contar con un dispositivo controlador central, que es una computadora inteligente, como un servidor. Pasa lista a cada nodo en una secuencia predefinida solicitando acceso de red. Si tal solicitud se realiza, el mensaje es transmitido, si no el dispositivo central se mueve a pasar lista al siguiente nodo.

## **2. TOPOLOGIA**

*La topología de red es la forma física o lógica como se conectan las computadoras. Existen varias topologías y se reducen a unos cuantos grupos.*

Existen tres topologías básicas para redes locales:

### **Bus**

En esta topología todos los nodos o estaciones de la red están interconectados a un único cable de comunicaciones llamado BUS o TRONCAL DE COMUNICACION, formando una trayectoria abierta y limitada en sus extremos por terminadores.

### **Anillo**

Esta topología se caracteriza por tener todos los nodos de la red interconectados entre sí, a través de cables de interfase, formando una trayectoria cerrada o de anillo. Si en esta topología, partimos de cualquier nodo de la red en cualquier dirección, siempre regresaremos al nodo original.

### **Estrella**

Esta topología se caracteriza por contar con un dispositivo central (File Server, Repetidor o Centro de Alambrado) al cual se conectan directamente las estaciones o nodos de la red a través de cables de interfase únicos.

### **3. MEDIO TRANSMISION**

*El medio de transmision es el cable o guía por la cual pasa la información. Existen varios tipos de transmisión:*

- Cable telefónico
- Coaxial
- Fibra Optica

### III.1.5. CARACTERISTICAS DE OPERACION DE TRES REDES LOCALES IMPORTANTES

#### ETHERNET

La red ETHERNET utiliza el protocolo de acceso CSMA/CA y su topología es de tipo Bus.

En esta red cada estación se encuentra monitoreando constantemente la línea de comunicaciones con el objeto de transmitir sus mensajes. Si la línea tiene tráfico, la estación espera un período muy corto, pero continúa monitoreando la red. Si la línea esta libre la estación transmisora envía su mensaje en ambas direcciones por toda la red. Cada mensaje incluye una identificación del nodo receptor puede leer el mensaje completo.

Cuando dos estaciones transmiten sus mensajes simultáneamente una colisión ocurre y es necesaria una retransmisión. El protocolo incluye las reglas que determinan cuánto tiempo tendrán que esperar los nodos para realizar sus envíos nuevamente.

Si una estación se encuentra procesando un mensaje, en el momento en que otra transmisión le llega, el mensaje se pierde y la estación transmisora lo retransmite nuevamente hasta que dicho mensaje es atendido.

Debido a esta forma de operación se tienen tiempos de respuesta inconsistente e impredecibles pero por la gran velocidad de transferencia de información con que cuenta ETHERNET (10 Megabits por segundo), su rendimiento es muy superior al de otras redes locales.

#### ARCNET

La red ARCNET utiliza comúnmente el protocolo de acceso Token Passing y la topología de Anillo con cableado en forma de estrella.

En ARCNET el Token pasa de un nodo de la red a otro en un orden ascendente.

Cuando una estación transmisora quiere transmitir su mensaje envía un "aviso" (Free Buffer) a la estación receptora preguntándole si puede aceptar dicho mensaje. Cuando la estación receptora le indica a la transmisora su incapacidad para aceptar su mensaje, la transmisora pasa el Token a la siguiente estación y transmite su mensaje la próxima vez que reciba el Token. Este

"aviso" evita el envío de datos antes de que la estación receptora tenga espacio para recibirlos.

Cada mensaje incluye una identificación del nodo fuente y del nodo destino y solamente el nodo destino puede leer el mensaje completo. En esta red no es necesario que cada estación regenere el mensaje antes de transmitirlo al siguiente nodo.

Todas las estaciones tienen la capacidad de indicar inmediatamente si pueden o no aceptar el mensaje y además, reconocen cuando ya fue recibido.

En ARCNET lo anterior elimina la necesidad de ocupar tiempos extras para retransmisiones pero su velocidad de transferencia de información es más baja (2.5 Megabits por segundo) comparada con ETHERNET.

#### TOKEN RING

TOKEN RING es una red con protocolo Token Passing y topología de Anillo con cableado en forma de estrella.

En este sistema el Token pasa de un nodo a otro de la red en una red en una sola dirección hasta completar el circuito. Cada estación le habla sólo a la estación que esta físicamente junto a ella en el anillo.

Cuando una estación tiene el Token transmite su mensaje, si es que tiene alguno, o simplemente pasa el Token a la siguiente estación.

En el momento en que una estación transmisora manda su mensaje, si es el Token pasa de un estado de "vacío" a un estado de "ocupado" y no puede ser usado para enviar mensajes por otro nodo. Cuando el nodo receptor lee su mensaje indica en el Token tal situación y lo transmite al siguiente nodo. Sólo cuando el Token regresa al nodo transmisor pasa a un estado de "vacío" pudiendo entonces ser utilizado para otra transmisión.

En TOKEN RING cada vez que el Token llega a un nodo el mensaje es regresado por dicho nodo, antes de pasarlo al siguiente. Es por esto que se reduce el rendimiento de la red pero se asegura una transmisión exitosa desde la primera vez que se envía el mensaje. TOKEN RING opera a una velocidad de transferencia de información de 4 Megabits por segundo.

Los conceptos antes mencionados sirven para darnos una idea de como funcionan las redes en general.

FALLA DE ORIGEN

### III.2. DESCRIPCION DE LA RED NOVELL.

#### III.2.1. INTRODUCCION A LAS REDES NOVELL.-

Las primeras redes locales estaban basadas en "Disk Servers<sup>3</sup>". Estos equipos permitían a cada usuario el mismo acceso a todas las partes del disco. Esto causaba obvios problemas de seguridad y de integridad en los datos.

La compañía Novell, Inc. fue la primera en introducir un "File Server<sup>4</sup>", en el que todos los usuarios pueden tener acceso a la misma información, compartiendo archivos y contando con niveles de seguridad, lo que permite que la integridad de la información no sea violada.

Novell basó su investigación y desarrollo en la idea de que es el Software de la red, no del Hardware, el que hace la diferencia en la operación de una red, esto se ha podido constatar.

Las tendencias actuales indican una definitiva orientación hacia la conectividad de datos. No es solo el envío de información de una computadora a otra sino, sobre todo, en la distribución del procesamiento a lo largo de grandes redes en toda la empresa.

La versión 3.11 de NetWare (las versiones nos indican actualización y mejoras en los sistemas) considera el manejo de la red a través de "MENUS", pequeñas pantallas que contienen opciones y cada una de estas opciones nos ofrecen posibilidades diferentes.

---

<sup>3</sup> Disk Server = Servidor de Disco

<sup>4</sup> File Server = Servidor de Archivo

### III.2.2. COMPONENTES DE UNA RED NOVELL

#### Red

El concepto de red se basa en el *principio de compartir recursos entre los usuarios*. La principal característica de este tipo de red es el contar con un lugar común de almacenamiento de los datos para compartir la información, entre los distintos usuarios que están dados de alta en la red. Esto es lo que se llama "Multiusuario".

Cada usuario tiene una microcomputadora que funciona como una Estación de Trabajo. Puede tener acceso a la información de un lugar común de almacenamiento, este lugar es el File Server (Servidor) donde se administra la operación de las estaciones de trabajo. El lugar común de almacenamiento es el disco duro "Hard Disk".

#### Componentes de una red

En una red controlada por Netware podemos señalar seis componentes indispensables:

- 1.- Servidor (F/S) (File/Server)
- 2.- Estaciones de trabajo (W/S) (Work/Station)<sup>5</sup>
- 3.- Tarjetas de Red (NIC)
- 4.- Cables
- 5.- Sistema Operativo Local
- 6.- Sistema Operativo Netware.

#### Servidor de Archivos

Es la computadora central encargada de compartir los recursos de la red. En ella reside el Sistema Operativo Netware. Adicionalmente existen otros tipos de Servidores como pueden ser los Servidores de Impresión, Bases de Datos, etc.

---

<sup>5</sup> Work Station = Estación de Trabajo

### **Estaciones de Trabajo**

Este es el nombre con el que se identifica a cada una de las computadoras que están conectadas a la red. Estas estaciones de trabajo tienen memoria y procesador propios (terminales inteligentes), es decir que en ellas se procesa cada programa a aplicación que es traído desde el Servidor de Archivos. Algunos tipos de Estación de trabajo son AT, XT, PS/2, Macintosh, etc.

### **Tarjetas de red**

Es el dispositivo encargado de la comunicación entre el Servidor de Archivos y la Estación de trabajo.

### **Cables**

Es el medio a través del cual viaja la información entre los componentes físicos de la red.

### **Sistema Operativo Local**

Es el software encargado de controlar la comunicación con todos los elementos internos de la Estación de Trabajo; por ejemplo: D.O.S. , OS/2, MAC, etc.

### **Sistema Operativo Netware**

Es el software encargado de controlar la comunicación con todos los recursos de la red.

### **Personajes de la red**

La red puede ser operada como:

#### **Usuario**

Es toda aquella persona que ha sido dada de alta como autorizada para operar la red. Cada usuario tiene una cuenta con un nombre específico el cual sirve como llave de entrada, además tiene asignados privilegios que están en función del papel que desempeña en la red.

### **Supervisor**

Es aquella persona que coordina a los usuarios y tiene absolutamente toda la responsabilidad en la organización y buen funcionamiento de la red. Debido a sus funciones, el supervisor requiere de todos los privilegios de operación y no puede ser borrado de la red.

### **Superusuario**

Dado el volumen de trabajo que implica administrar la red y como medida de seguridad es recomendable que exista en la red al menos un usuario con "equivalencias de supervisor", es decir un usuario con los mismos privilegios que el supervisor.

### **Manager**

Es cualquier usuario al que se le han asignado ciertos derechos de supervisor para que ayude a éste a manejar usuarios. Dependiendo del grado de responsabilidad que se le asigne los Managers se clasifican en :

**Manager de grupo de trabajo (WGM)**  
**Manager de administración de usuarios (UAM)**

### **Operador**

Es cualquier usuario al que se le han asignado ciertos derechos de supervisor para que ayude a éste con ciertas tareas de la red. Dependiendo del tipo de tarea que se le asigne, un operador puede clasificarse en :

**Operador de servidor de archivos (CO)**  
**Operador de colas de impresión (PQO)**  
**Operador del servidor de impresión (PSO)**

### **Grupos**

Los grupos son una forma de organizar a los usuarios que pertenecen a una misma área o perfil de operación, cada grupo al igual que los usuarios tendrá sus privilegios asignados.

### III.2.3. ESTRUCTURA DE LA RED

Dentro del disco duro del Servidor existe una estructura de directorios semejante a la que podemos crear en un disco controlado por sistema operativo DOS.

Netware crea cuatro directorios en el volumen, que no se deben modificar: Login, Mail, Public y System.

#### Login

Es el directorio de acceso a la red porque en el se encuentran las órdenes de entrada para los usuarios que deseen hacer uso de los recursos.

#### Mail

Es un directorio en el que esta definido un buzón (box) para cada usuario de la red. Dentro de este se guarda su Login Script (secuencia de órdenes específicas para dicho usuario, que se ejecuta para entrar a la red).

#### Public

Es el lugar donde se encuentran todas las órdenes de NetWare que puede utilizar el usuario, es decir aquellas que son de dominio público.

#### System

En el se encuentran las órdenes exclusivas para el supervisor de la red. A ellas no tiene acceso el usuario normal de la red.

Además de los directorios anteriores, es recomendable crear los siguientes:

#### Programa

Es el directorio donde se definen las áreas de trabajo de los usuarios, es decir el espacio de disco que estos utilizan para guardar sus archivos de datos.

Una estructura de este tipo, que se puede modificar de acuerdo a las necesidades y características de cada empresa, permite una mejor organización y administración de los recursos de la red.

### III.3. ARQUITECTURA DE LA RED NOVELL

La arquitectura universal de la red consta de las siguientes partes:

- 1.- Tecnología de File Server
- 2.- Tolerancia a fallas
- 3.- Tecnología de Protocolo Abierto
- 4.- Administración de la Red.

En 1987, la participación de Novell en el mercado creció desde el 50% hasta el 62% sin embargo, de las corporaciones que harán próximamente adquisiciones reales, el 76% planean comprar Netware.

La estrategia se basa en la UNA (Universal NetWare Architecture) proyecto utilizado para determinar los tipos de tecnología que Novell construye.

La UNA acompaña a todas las tecnologías de servidores, incluyendo al servidor de archivos, de comunicación, de impresoras y de base de datos.

#### PRIMER LOGRO

##### Tecnología de File Server

La tecnología de File Server comprende cuatro componentes básicos:

- Servicios de archivo de alta velocidad utilizados por los protocolos del núcleo de Netware. Estos servicios incluyen la capacidad de compartir archivos remotos, impresoras y herramientas de manejo de la red.

- Capacidad de base de datos distribuida (razón por lo cual se creó el Netware SQL) que permite a los usuarios usar los paquetes estándar que usa SQL para dar acceso a la información por vía de Netware.

FALLA DE ORIGEN

- Servicio de Manejo de Mensaje (MHS). Tecnología que almacena y dirige, mediante el procesamiento de datos distribuidos. Ayuda a los desarrolladores líderes a llegar a la siguiente generación de aplicaciones.

- Subsistema de comunicación. Permite que las LANs geográficamente dispersas se comuniquen utilizando protocolos asíncronos y tecnología X.25

## **SEGUNDO LOGRO**

### **Tolerancia de Fallas**

NetWare proporciona integridad de la información por medio de tres niveles de Tolerancia de Falla del Sistema (STF).

- Todo los sistemas de NetWare utilizan el Nivel I, el cual incluye el Hot Fix e información de archivo duplicado.

- NetWare SFT proporciona un reflejo de disco y rastreo de transacción.

## **TERCER LOGRO**

### **Tecnología de Protocolo Abierto.**

- Ya que los usuarios requieren de un acceso transparente desde medios ambientes heterogéneos, la Tecnología de Protocolo Abierto (OPT) es un paso natural en el desarrollo de las redes de proceso distribuido.

- Cinco medios ambientes claves que requieren de acceso transparentes incluyen DOS, OS/2, el Sistema Operativo Macintosh y el NetWare, UNIX y VMS.

- Dos productos OPT son el NetWare para Macintosh y el NetWare para VMS.

**FALLA DE ORIGEN**

## CUARTO LOGRO

### Administración de la Red

El manejo de la red llega a ser crítico, conforme las redes se incrementan en tamaño y complejidad.

La estrategia de Novell para mejorar el manejo de la red incluye cuatro fases:

- La primera fase de Novell proporciona herramientas y capacidad dentro del NetWare para manejar las instalaciones tales como, la configuración del sistema operativo, la seguridad, el análisis del funcionamiento y el aislamiento de falla.

- La segunda fase asegura que estas capacidades puedan tener acceso remotamente a través de las Netware Application Program Interfaces (APIs) (Interfases para programas de aplicación de NetWare).

- La tercera fase permite que los usuarios del NetWare se conecten a otros sistemas huéspedes.

- La cuarta fase permite que los usuarios del NetWare "mapéen" a otro sistema centralizado del manejo de la red de los vendedores, tales como la Netview de IBM.

NetWare tiene :

- Interacción con MS-DOS
- Comparte disco duro e impresoras
- Soporta hasta cuatro tarjetas de red permitiendo usar diferentes tipos de Hardware.
- Cuenta con el Software necesario para que un microcomputador se conecte vía Modem.
- Maneja un administrador de cola impresión.

- Administrador de recursos a través de la consola del servidor.
- Cuenta con gran cantidad de utilerías y ordenes del sistema.
- Consola Virtual.
- Contabilidad de recursos
- Permite limitar la cantidad de almacenamiento en disco.
- Absoluto control sobre el acceso de los usuarios a la red a través de cuatro niveles:
  - \* Login y Password
  - \* Derechos de Usuario
  - \* Derechos de Directorio
  - \* Atributos de los Archivos

Estos son algunos de los conceptos y/o elementos integrados en una red LAN NOVELL.

Por lo cual el departamento de servicios escolares cuenta con el siguiente equipo integrado en red :

Cuenta con equipo integrado en una *Red Lan Novell 3.1, con protocolo Ethernet CSMA/CA, topología en Bus, con un medio de transmisión a través de Cable Coaxial.*

Existen cinco estaciones de trabajo, cada una constituida por una microcomputadora GAMA con procesador 80286, monitor VGA monocromático, disco duro de 52 MBytes, 1 MByte de memoria RAM y tarjeta de red Novell Ethernet 3com de 8 bits.

El servidor está constituido por una microcomputadora GAMA con procesador 80486, monitor VGA monocromático, dos discos duros de 320 MBytes, 16 MBytes de memoria RAM y una tarjeta de red Novell Ethernet 3com de 16 bits.

Además de una impresora Epson de matriz de punto. Modelo DFX-8000.

La red es administrada mediante un sistema operativo Novell Ethernet versión 3.11 para 20 usuarios.

Además de la red, el departamento cuenta con dos computadoras adicionales para su servicio:

Una microcomputadora GAMA con procesador 80286, monitor VGA monocromático, 1 MByte de memoria RAM y disco duro de 52 MBytes, destinada al departamento de revisión de estudios.

FALLA DE ORIGEN

**CAPITULO IV.- DESARROLLO DEL SOFTWARE****IV.1. DISEÑO DEL SOFTWARE****IV.2. ALTERNATIVAS DEL DISEÑO****IV.3. DIAGRAMA MODULAR DEL SISTEMA****IV.3.1. El Sistema Visión General****IV.3.2. Modulo de Reinscripción****IV.3.3. Modulo de Consultas****IV.4. FLUJO DE LA INFORMACION****IV.4.1. Archivos de Trabajo o Bases de Datos.**

**OBJETIVO: DISEÑAR UN SISTEMA DE REINSCRIPCIONES,  
A TRAVES DE LA INFORMACION OBTENIDA Y  
LAS OBSERVACIONES REALIZADAS.**

#### IV.1. DISEÑO DEL SOFTWARE.

La fase de desarrollo se centra en el cómo. Esto es, durante esta fase, el que desarrolla el software intenta cubrir cómo han de diseñarse las estructuras de datos y la arquitectura del software, cómo ha de realizarse la prueba. Los métodos aplicados durante la fase de desarrollo variarán, pero de alguna forma se producirán tres pasos concretos:

**Diseño del software.** El diseño traduce los requisitos del software a un conjunto de representaciones (algunas gráficas y otras tabulares o basadas en lenguajes) que describan la estructura de los datos, la arquitectura, el procedimiento algorítmico y las características de la interfaz.

**Codificación.** La representación del diseño debe ser traducido a un lenguaje artificial (un lenguaje de programación convencional o un lenguaje no procedimental), dando como resultado unas instrucciones ejecutables por la computadora. El paso de la codificación es el que lleva a cabo esa traducción.

**Prueba del Software.** Una vez que el software ha sido implementado en una forma ejecutable por la máquina, debe ser probado para descubrir los defectos que puedan existir en la función, en la lógica y en la implementación.

#### IV.2. ALTERNATIVAS DE DISEÑO

Por lo general para el diseño de el sistema es necesario trabajar en un manejador de Base de Datos.

En la actualidad existen varios manejadores de bases de datos, con un ambiente fácil de manejar. Sin embargo, no podemos utilizar cualquiera, ya que se deben de adaptar a las condiciones del lugar. Debemos verificar las condiciones de equipo con que cuenta las instalaciones a realizar este proceso, además de la compatibilidad del software que se ocupe y la capacidad de datos que se puedan almacenar, entre otras características.

En nuestro caso, el departamento de servicios escolares cuenta con un equipo regular y no se adapta a algunos manejadores de datos.

Por lo tanto, se ha optado por usar uno de uso más simple y sencillo.

Entre los manejadores de bases de datos que más se usan tenemos al D'BASE IV, CLIPPER 5.01, FOXPRO, ORACLE, INFORMIX, ACCESS entre otros.

En este caso, se ha utilizado el CLIPPER 5.01, por su flexibilidad, sencillez de manejo y adaptabilidad con el equipo empleado. Además que los sistemas anteriores están programados en este lenguaje orientado a las bases de datos.

Las ventajas de ocupar CLIPPER es que no ocupa mucho espacio de memoria en nuestra computadora. Otra ventaja es que el programa puede ejecutarse directamente en el Sistema Operativo. Además se tiene la opción de utilizar arreglos multidimensionales. El código generado por los programadores en Clipper es pequeño en comparación con el generado por los otros manejadores considerados. También es posible adicionar funciones en lenguaje C, lo que hace que su potencia aumente considerablemente.

Por lo que podemos mencionar que el proceso de compilación de CLIPPER funciona de la siguiente manera: el programa fuente (con extensión .PRG) generado por el editor de textos es convertido inicialmente en un módulo (con extensión .OBJ) que, a su vez, es sometido al proceso de encadenamiento por el LINKAGE EDITOR cuya finalidad es la de generar un programa ejecutable (con extensión .EXE)

El Clipper permite compilar programas, PROCEDURE, formatos de pantalla y funciones definidas por el usuario, siempre y cuando contengan la extensión .PRG.

Básicamente, el proceso de encadenamiento agregará al programa objeto rutinas en lenguaje de máquina y otros programas de utilidad para permitir que el .EXE pueda ejecutarse directamente en el Sistema operativo.

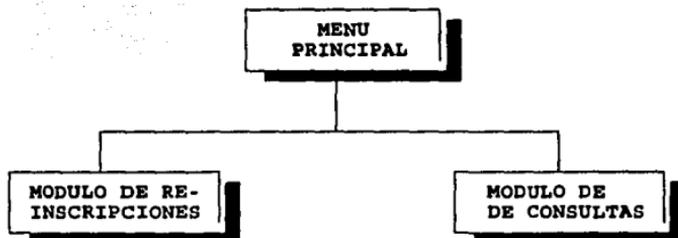
El Clipper es un compilador de dos pasadas. En la primera, genera una serie de símbolos que representan los comandos y elementos del programa. En la segunda pasada, utiliza esos símbolos para generar el código objeto.

### IV.3. DIAGRAMA MODULAR DEL SISTEMA

#### IV.3.1. El Sistema Visión General.

Después de haber obtenido la información necesaria, el próximo paso es la definición del sistema en forma sencilla y la manera más fácil es a través de un diagrama a bloques.

El siguiente diagrama muestra en la forma en que esta dividido. Básicamente, consta de un menú principal. Veamos la estructura del sistema:



El sistema general de reinscripciones cuenta con un menú principal, en el cual se dan las opciones de Inscripción, Consultas y Salida.

Ante algunas observaciones, se ha agregado al sistema un módulo nuevo. Este módulo pretende realizar consultas, en cuanto al registro de los alumnos. Posteriormente este módulo se explicará detalladamente.

**IV.3.2. Modulo de Reinscripción.****MODULO DE RE-  
INSCRIPCION****ARQUITECTURA****DISEÑO IND.****ECONOMIA****PLANIFICACION****PEDAGOGIA**

El modulo de Reinscripciones está integrado por las carreras que van a realizar este trámite. En este caso: Arquitectura, Diseño Industrial, Economía, Pedagogía y Planificación para el Desarrollo Agropecuario.

Cada uno de los bloques trabaja con su respectivo directorio, es decir, con sus respectivas bases de datos. Las bases de datos son las mismas que anteriormente se explicaron.

#### ALGORITMO

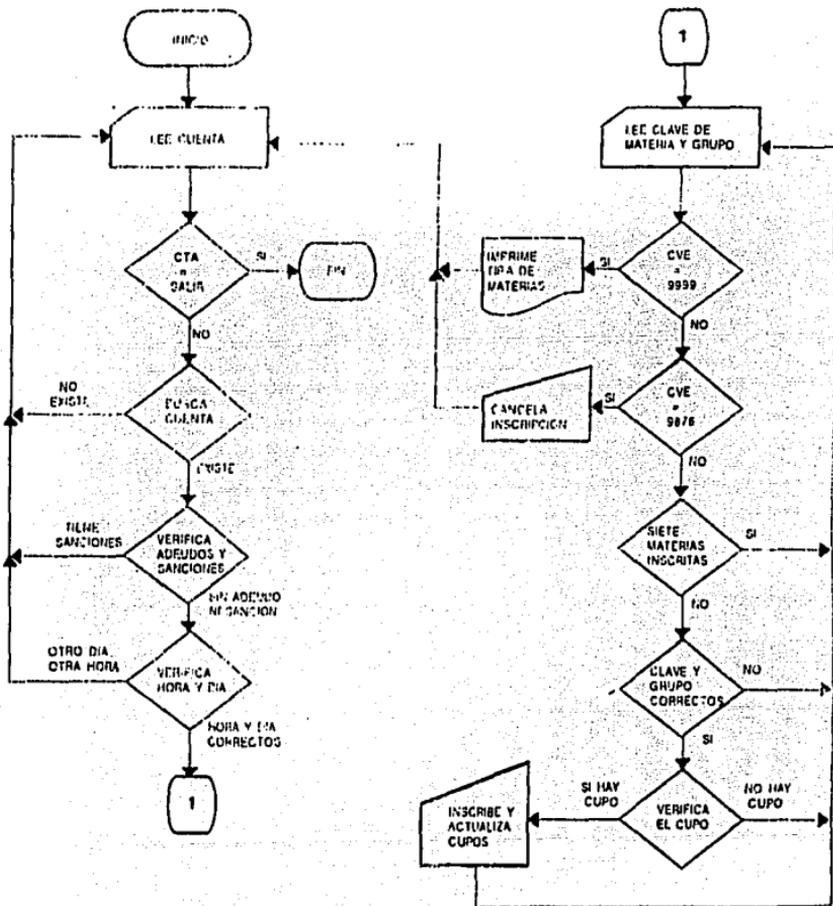
El proceso de reinscripción se realiza, de acuerdo a las condiciones de cada carrera.

El proceso general de Inscripción sigue los siguientes pasos:

1. INICIA el proceso
2. Lee cuenta. Si cuenta = salir termina  
Si cuenta = cuenta continua
3. Busca cuenta. Si no existe se regresa y pide cuenta otra vez. Si existe continua
4. Verifica adeudos y sanciones. Si tiene sanciones regresa al inicio, sino continua y
5. Verifica la hora y el día, sino cumple con esto se regresa otra vez al inicio, sino vuelve a continuar
6. Lee claves de materias y grupos
7. Si clave = 9999, inscripción aceptada manda imprimir tira de materias y regresa al inicio.
8. Si clave = 9876, inscripción cancelada y regresa al inicio otra vez. Sino continua
9. Verifica el número de materias inscritas hasta alcanzar el número correcto correspondiente para cada carrera, por lo tanto mientras sea no, regresa a leer la clave y el grupo de la materia. Si es si llena el total de materias a cursar y entonces continua.
10. Verifica que la clave y grupo estén correctos, sino están correctos regresa a leerlos otra vez, si están correctos continua.
11. Verifica el cupo, sino hay cupo regresa a leer otra clave y grupo, pero si existe cupo, inscribe y actualiza cupos. Y regresa a leer otra clave de materia y grupo, hasta alcanzar el total de materias aceptadas por cada carrera.

Para todos los módulos de inscripciones, es decir, para cada carrera el proceso de inscripción es el mismo. Solo varían en algunas condiciones explicadas a continuación.

# DIAGRAMA DE FLUJO DEL PROGRAMA DE INSCRIPCIONES



FALLA DE ORIGEN

### **MODULO DE ARQUITECTURA**

La carrera de Arquitectura realiza sus inscripciones por medio de bloques. Cada uno ellos consta de 7 materias por semestre. En caso de que los alumnos reprobarán alguna materia, solo pueden cursar seis materias y una más en recursamiento. Tiene dos horarios para asistir a clases, matutino y vespertino, pero a partir del séptimo semestre el horario es mixto. El sistema se adecua a esta condición también, para que los alumnos no tuvieran ningún problema al realizar su reinscripción.

### **MODULO DE DISEÑO INDUSTRIAL**

El modulo de Diseño Industrial tiene un solo turno y el máximo de materias que cursan son de nueve materias los tres primeros semestres, los demás son de ocho materias. La carrera consta de ocho semestres.

### **MODULO DE ECONOMIA**

La carrera de Economía lleva un máximo de ocho materias por semestre con materias recursadas por la tarde. Por lo cual, lleva un horario mixto.

### **MODULO DE PEDAGOGIA**

En Pedagogía, los alumnos cursan un máximo de 10 materias por semestre con recursamiento. No tiene seriación de materias y tampoco tienen turno mixto. Es decir, dos turnos, Matutino y Vespertino. Consta de ocho semestres.

### **MODULO DE PLANIFICACION**

En Planificación Agropecuaria no existe la seriación obligatoria. Se cursan un máximo de siete materias por semestre con tres materias recursadas. La carrera se imparte en dos turnos, Matutino y Vespertino y consta de ocho semestres.

#### IV.3.3. MODULO DE CONSULTAS

El modulo de consultas esta dividido de la siguiente manera:

**MODULO DE  
CONSULTAS**

**DIRECTORIO**

**TIRAS MATER.**

**HISTORIAL  
ACADEMICO**

**PLAN DE  
ESTUDIOS**

Este modulo de consultas pretende que de alguna manera, cuando lo requiera el usuario pueda verificar sus datos personales en el directorio, agregándole su dirección y su teléfono, pueda ver la tira de materias de su anterior inscripción, así como su historial académico y que también pueda mostrarse el plan de estudios de su carrera. En este caso este modulo sólo se encuentra integrado en este sistema de inscripciones a las carreras de Arquitectura, Diseño Industrial, Economía, Pedagogía y Planificación para el Desarrollo Agropecuario.

Las carreras estarán integradas en un submenú y aparecerán en el momento de escoger alguna opción del menú de consultas.

#### **MODULO DE DIRECTORIO**

Este módulo desplegará los datos de los alumnos registrados en su carrera respectiva. Es decir, tomará los datos integrados en la base de datos DIRALUM.DBF que se localiza en el subdirectorio de cada carrera, integrada en este sistema. Al momento de escoger un alumno aparecerá un cuadro que desplegará el nombre, la dirección y el teléfono del alumno. Regresando al menú de consultas con un ESC(Tecla).

#### **MODULO DE TIRAS DE MATERIAS**

Esta parte despliega la tira de materias de los alumnos ya inscritos en las carreras (Arquitectura, Diseño, Economía, Planificación y Pedagogía). Se puede realizar de forma general o de manera personal, escogiendo al alumno que se quisiera verificar. Ocupa las base de datos que se ocupan para la inscripción (INSC961.DBF)

#### **MODULO DE PLAN DE ESTUDIOS**

Este modulo desplegará el plan de estudios de cada una de las carreras integradas en este sistema. Sus bases de datos correspondientes son: PLARQUI.DBF, PLADISE.DBF, PLAECON.DBF, PLAPLANIF.DBF, PLAPEDAG.DBF. Integradas casi igual que la base de datos ocupada para inscripciones TABLMAT.DBF, adicionandole un registro.

#### **MODULO DE HISTORIAS ACADEMICAS**

Se pretende que este modulo tome las Historias Académicas de cada uno de los alumnos de las distintas carreras y las muestre en la pantalla al pedir la cuenta del alumno o nombre del alumno.

#### V.4. FLUJO DE LA INFORMACION

##### Archivos de Trabajo o Bases de Datos.

Este sistema actúa básicamente con varios archivos o bases de datos. Sin estos archivos, el programa no funcionaría. Uno de ellos es el archivo de alumnos, que es utilizado con mayor frecuencia. Así como este, utilizamos otras bases de datos, empezaré por mostrar su estructura de cada una ellas.

##### DIRALUM

##### ESTRUCTURA DE DIRALUM.DBF

REGISTRO	TIPO	ANCHO
CUENTA	CARACTER	7
D	CARACTER	1
NOMBRE	CARACTER	32
SEXO	CARACTER	1
NAC	CARACTER	1
FECHANAC	CARACTER	6
GEN	CARACTER	2
TING	CARACTER	2
SISTEMA	CARACTER	1
MDINSC	CARACTER	4
HORAINSC	CARACTER	1
CLASINSC	CARACTER	1
ULTINSC	CARACTER	3
CARRERA	CARACTER	2
PLANTEL	CARACTER	3
INSCRIBIO	CARACTER	1
DIRECCION	CARACTER	30
TELEFONO	CARACTER	10

La base de datos DIRALUM.DBF almacenará la información general de los alumnos de las carreras a automatizar, para esto se ha creado un directorio para cada carrera. Es decir, hay un directorio para Arquitectura con sus respectivas bases de datos, y así sucesivamente para las otras carreras.

Esta base de datos se estructuró de forma que sólo tuviera información del alumno pero, al mismo tiempo, tuviera la facilidad de relacionar sus campos con cualquier otra base de las empleadas.

Se le ha agregado dos registros más: Dirección y Teléfono como referencia para el módulo de consultas.

El campo INSCRIBIO se utiliza para tener un control sobre los alumnos que ya se reinscribieron. El campo ULTINSC contiene la información del último semestre en el que el alumno realizó su reinscripción.

Los campos MDINSC y HORAINSC contiene la información de el día y la hora de inscripción de cada alumno.

El campo CLASINSC puede contener cualquiera de las siguientes claves:

01	Alumno con carrera terminada
02	Afectado por el Art. 19
03	No inscrito en semestre anterior
	Sin información de clasificación
05	Procede la reinscripción

Si la información contenida en el campo CLASINSC es diferente de 5, la reinscripción no procede.

#### TABLMAT

#### ESTRUCTURA DE TABLMAT.DBF

REGISTRO	TIPO	ANCHO
CVE	CARACTER	4
NOMAT	CARACTER	36
CREDITOS	CARACTER	2
SEMESTRE	CARACTER	2

Esta base de datos contiene la información referente a las materias. Nombre, clave, créditos y al semestre que pertenecen. Es empleada en reinscripciones y en ajuste de las mismas.

## MG961

## ESTRUCTURA DE MG961.DBF

REGISTR	TIPO	ANCHO
CVEMAT	CARACTER	4
GRUPO	CARACTER	4
CUPO	CARACTER	3
NALUM	CARACTER	3
NOMPROF	CARACTER	30
HORARIO	CARACTER	25

Esta base de datos tiene registradas las materias por clave y grupos, así como el cupo limite de alumnos inscritos a ese grupo, el nombre del profesor que imparte esa materia y sus horario.

## CTRL961

## ESTRUCTURA DE CTRL961.DBF

REGISTRO	TIPO	ANCHO
IDREG	CARACTER	1
CUENTA	CARACTER	7
TIPO	CARACTER	2
TEXTO	CARACTER	60

Esta base de datos sirve para llevar el control de los alumnos que tengan adeudos en biblioteca o en sección escolar. De acuerdo a esta opción, los alumnos no tienen derecho a realizar el trámite de reinscripción a menos que salden los mismos. La base de datos CTRL961 tiene la finalidad de controlar dichos alumnos no permitiendo que su inscripción proceda hasta haber resuelto su situación. Teniendo su comprobante de no adeudo, el sistema tiene un modulo mediante una clave para poder realizar la reinscripción.

El campo texto es empleado para dar información del adeudo que tiene el alumno.

MINF961

## ESTRUCTURA DE MINF961.DBF

REGISTRO	TIPO	ANCHO
CUENTA	CARACTER	7
CVEMAT	CARACTER	4
INDCOD	CARACTER	1

Esta base es de utilidad para detectar si un alumno ya se ha inscrito dos veces a la misma materia. Al entrar al sistema la materia se buscará en la base de datos. Si la clave de la materia ya existe y el campo INCOD es igual a uno, la materia ha sido tomada una sola vez, si la materia aparece pero INDCON es diferente de uno entonces la materia ya se habrá cursado dos veces. En caso de que la clave no exista dentro de la base de datos, será indicativo de que la materia no ha sido tomada con anterioridad.

## INSC961

## ESTRUCTURA DE INSC961.DBF

REGISTRO	TIPO	ANCHO
CUENTA	CARACTER	7
D	CARACTER	1
GEN	CARACTER	2
NOMBRE	CARACTER	32
MAT1	CARACTER	4
GPO1	CARACTER	4
MAT2	CARACTER	4
GPO2	CARACTER	4
.		
.		
MAT10	CARACTER	4
GPO10	CARACTER	4
FOLIO	CARACTER	6
TIMEINI	CARACTER	6
TIMEFIN	CARACTER	6
FECHA	CARACTER	6
ATENDIO	CARACTER	20

Las bases de datos anteriores son las de mayor importancia y las que ocupa el sistema.

**CAPITULO V. - APLICACION****V.1. PRUEBAS Y VALIDACION****V.2. DEPURACION****V.3. MANTENIMIENTO****V.3.1. Confiabilidad****V.3.2. Confidencialidad**

**OBJETIVO: DETERMINAR LA APLICACION TOTAL DEL SISTEMA  
MEDIANTE LAS PRUEBAS, VALIDACION Y  
POSTERIORMENTE EL MANTENIMIENTO DEL MISMO.**

## V.1 PRUEBAS Y VALIDACION

La importancia básica de realizar este sistema es de que funcione y que las condiciones dadas por cada carrera se lleven a cabo. Por lo tanto, en esta parte se describirán las pruebas realizadas y la validación de las mismas.

Antes de comenzar con las pruebas de mayor peso, se hicieron ajustes al sistema, mediante la introducción de información ficticia.

Se le hicieron varios ajustes al inicio, en cuanto a que estuviera tomando las carreras con dos turnos y las de turno mixto. Así como el ajuste de materias para las mismas. Además que tomará el día y las horas de inscripción para los alumnos.

Se comprobó que desde el punto de vista del alumno, los trámites de realizarán dentro de los parámetros establecidos. Esto incluye verificar que un alumno no tenga adeudos en biblioteca o sección escolar, verificar que le corresponda su reinscripción de acuerdo al día y a la hora en función de su número de sorteo, verificar que el alumno no se inscriba dos veces, que no intente realizar inscripciones el mismo alumno en terminales simultaneas entre otros casos particulares.

Operativamente y desde el punto de vista del software se verificaron los siguientes puntos:

- Que los registros de los alumnos que efectúen su reinscripción queden adecuadamente marcados.
- Que las materias y los grupos a los que un alumno se inscribe queden adecuadamente marcados.
- Que las actualizaciones de los cupos de los grupos se efectúen de manera inmediata y correcta.
- Que los cambios efectuados en la base de datos que tiene el control de los adeudos se realicen en forma automática.

Desde el punto de vista de Hardware, los posibles errores pueden definirse solamente como errores impredecibles ya que no se puede determinar si van a suceder, ni cuando van a suceder. Estos errores se refieren al suministro de energía eléctrica y la operación de la impresora.

Las simulaciones realizadas para tomar acciones en caso de errores en el equipo son las siguientes:

- Falla en el suministro de energía eléctrica. Cuando sucede esto, los datos son procesados inmediatamente y antes son respaldados sin ningún problema. Los procesos que se estén ejecutando durante la falla del sistema deberán ser repetidos al restablecerse la energía. El servidor de la red cuenta con una fuente de poder ininterrumpible la cual da un tiempo de 15 min. aproximadamente, suficientes para restablecer el sistema hasta que comience a funcionar la planta de la energía del plantel.

- Fallas durante el proceso de impresión. Al imprimirse las tiras de materias, puede darse el caso de que la impresora tenga problemas. Esto no tiene ningún efecto sobre la información generada. El sistema cuenta con un programa de utilería que sirve para que dado un número de cuenta de algún alumno ya inscrito, se imprima otra tira de materias. El efecto de que la impresora salga de línea durante el funcionamiento del sistema hace que las terminales reciban un mensaje de aviso, solicitando dar (ctrl-enter) para continuar su trabajo. Los problemas que pueda tener la impresora son de en general, termino de las formas de impresión o atascamiento del papel. Para corregir estos errores se pedirá a una persona que vigile, además de que separe y organice las tiras de materias.

Después de realizar este tipo de pruebas, se procedió a realizar pruebas dentro de la red.

Para este proceso se llenaron las bases de datos con datos más reales y se procedió a crear subdirectorios dentro de la red. Es decir, las rutas de acceso para cada una de las carreras y proceder a realizar las reinscripciones.

De acuerdo a todo lo anterior, se acepta la funcionalidad del sistema y haciendo validos todos los cambios. Agregando así, el modulo de consultas descrita en el capitulo anterior.

## V.2. DEPURACION

A diferencia de la prueba (una actividad sistemática y planificada), la depuración se puede considerar un arte. A partir de una indicación sintomática de un problema, la actividad de depuración debe rastrear la causa del error. De entre los recursos disponibles durante la depuración, el más valioso puede ser el apoyo de otros miembros de la plantilla de desarrollo del software.

La depuración no es una tarea de prueba, aunque siempre se da como una consecuencia de la prueba.

Para aplicar el concepto de depuración a nuestro sistema, después de realizarse algunas pruebas se hizo lo siguiente:

El sistema se presentó a cada una de las coordinaciones para verificar que el sistema hacía las reinscripciones de acuerdo a las condiciones establecidas por cada coordinación.

Al realizar esto, la carrera de Arquitectura pidió un ajuste para la reinscripción de los alumnos a los dos últimos semestres (séptimo y octavo sems.), que se pudieran reinscribir a grupos en la mañana y en la tarde, es decir, que pudieran tomar horario mixto durante estos semestres.

La carrera de Diseño Industrial hizo una observación, que si era posible que ellos pudieran tener un mayor control en cuanto esto, es decir, que si aparte de los datos de referencia de los alumnos se pudieran incluir la dirección y teléfono de los mismos. Lo cual al final se incluye el módulo de consultas tomando en cuenta también el despliegue de la tira de materias anterior, plan de estudios de su carrera y los historiales académicos de cada alumno.

### V.3. MANTENIMIENTO

La fase de mantenimiento se centra en el cambio que va asociado a la corrección de errores, a las adaptaciones requeridas por la evolución del entorno del software y a las modificaciones debidas a los cambios de los requisitos del cliente/usuario dirigidos a reforzar o ampliar el sistema. La fase de mantenimiento vuelve a aplicar los pasos de las fases de definición y de desarrollo, pero en el contexto del software ya existente. Durante la fase de mantenimiento se encuentran tres tipos de cambios:

**Corrección.** Incluso llevando a cabo las mejores actividades de garantía de calidad, es muy probable que el cliente o usuario descubra defectos en el software. El mantenimiento correctivo cambia para corregir los defectos.

**Adaptación.** Con el paso del tiempo es probable que cambie el entorno original (por ejemplo, la UCP, el sistema operativo, los periféricos) para el que se desarrolló el software. El mantenimiento adaptativo consiste en modificar el software para acomodarlo a los cambios de su entorno externo.

**Mejora.** Conforme utilice el software, el cliente/usuario puede descubrir funciones adicionales que podría interesar que estuvieran incorporados en el software. El mantenimiento perfecto amplía el software más allá de sus requisitos funcionales originales.

Si se entiende claramente el sistema que se va automatizar, la estructura debe de estar bien definida. Por lo tanto, debe tener una facilidad de mantenimiento. Teniéndose en cuenta que un sistema no es una cosa estática, su arquitectura debe poseer medios para que se puedan efectuar modificaciones con poco trabajo. Cuanto más se detalle un sistema, más fácil será su mantenimiento.

A este sistema actualmente se le ha dado un enfoque nuevo, es decir la inclusión del módulo de consultas; con esto se pretende que los alumnos tengan mayor información de sus trámites de reinscripción y que también las coordinaciones tengan mayores referencias de ellos.

Como anteriormente se explica el módulo de consultas nos va a mostrar el directorio de cada carrera, el plan de estudios, las tiras de materias anteriores de los alumnos y también las historias académicas.

### **V.3.1. Confiabilidad.**

La confiabilidad se determina con el buen funcionamiento del sistema. En este caso, puedo decir, que el sistema es confiable, a menos que se tenga pérdidas de grandes cantidades de información. Lo cual no sucede en este caso. Ya que lo más probable es existan fallas con respecto a la energía eléctrica y como anteriormente se mencionaba se hace un respaldo automático de la información en el sistema. Es confiable en medida de que cualquier falla que suceda, ya sea, en el momento de impresión, caída de la red, entre otras, el sistema tiene unas utilerías de paro de las máquinas que estén en servicio, mandando mensajes de salida del sistema y respaldo de la información a los usuarios que estén dentro del sistema.

### **V.3.2. Confidenciabilidad.**

En este caso, la confidenciabilidad del sistema pretende ser total. Es decir, el acceso al sistema o a la información es de sumo cuidado, ya que solamente las personas que manejan el sistema tienen acceso a verificarla. Así como claves o password de entrada para el acceso de manejo al sistema.

## **CONCLUSIONES**

### CONCLUSIONES

Como hemos visto en esta tesis la computadora en la actualidad no es tan sólo una herramienta de trabajo que facilita el desempeño de diferentes actividades en casi todos las áreas del conocimiento humano, sino que es un fenómeno que está generando grandes cambios tecnológicos, administrativos, políticos, etc. en todo el mundo. Por lo que es fundamental no perder de vista la verdadera dimensión de la importancia que tiene dentro del enfoque de procesos cualesquiera que sean estos y utilizarla conscientemente en beneficio de una área de trabajo con un criterio crítico, analítico y cien por ciento empresarial. Y no automatizar procesos por el simple hecho de hacerlo.

Por otra parte también pudimos darnos cuenta de la importancia del mantenimiento a los programas de cómputo, ya que debemos de entender que no existen programas perfectos, lo que implica que los ingenieros de sistemas deben tener un criterio abierto a cualquier signo de cambio para que sus programas sigan siendo eficientes y no tan solo a sus programas sino a todo el ámbito que los rodea. Ya que la vida útil de un programa de cómputo no esta dada por el tiempo, sino por su eficacia.

## **APENDICE**

**APENDICE**

**PROGRAMA PRINCIPAL DE INSCRIPCIONES (MODULO DE INSCRIPCIONES Y CONSULTAS)**

```

static OP:=1
PERIODO = "961"
MESE='ENEFEEMARABRMAYJUNJULAGOSEPOCTNOVDIC'
DECLARE ASIGS[12], AMAT[12],AGPOS[12] //DECLARACION DE ARREGLOS

*DECLARACION DE VARIABLES GLOBALES*
*****
CONSEC:=HORAINIC:=INDMAT:=HORAFIN:=SEGURO:=AUX1:=CLRROR:=CAPGPO:=
COLOR1:=;
CTASIG:=CTA:=ERRORENC:=GENE:=HRINSC:=CHNUM:=IAUX:=INDAUX:=IN:=ASIG:=ASIGENC:=;
NACION:=CAP:=ALNUM:=SEX:=MATFIN:=TIPS:=ULTINSC:=HOY:=NUMPRN:=0
CTAS:= '.....'
ASIGREN := 05
MRGIZQ := 07
CLEAR
SET DELE ON
SET EXCL OFF
SET BELL OFF
SET SCOR OFF
SET COLOR TO
* USUARIO := '
*DO WHILE LEN(USUARIO) = 0
* @ 11,10 SAY '
* ACCEPT ' TECLEE SU NOMBRE POR FAVOR : ' TO USUARIO
* USUARIO := LTRIM(UPPER(USUARIO))
*ENDDO
IF .NOT. ISCOLOR()
COLOR1 := 'GR,N,GR+/G'
ERRORENC := 'GR+,R+*'
CLRROR := 'GR+,GR+'
ELSE
COLOR1 := 'BG,GR+'
ERRORENC := 'GR+,R+*'
CLRROR := 'R+,GR+'
ENDIF

empresa1 ="UNIVERSIDAD NACIONAL AUTONOMA DE MEXICO"
empresa2 ="ESCUELA NACIONAL DE ESTUDIOS PROFESIONALES - ARAGON"
sistema = "SISTEMA DE INSCRIPCION"
SET SCOREBOARD OFF
SET DATE BRITISH
SET WRAP ON
SET DELETED ON

```

```

DO WHILE .T.
  CLEAR
  SET WRAP ON
  @ 10,20 TO 18,52 DOUBLE
  SET MESSAGE TO 23 CENTER
  SAVE SCREEN
  DO WHILE .T.
    telapadiao(PROCNAME())
    menú := {" SISTEMA DE INSCRIPCIONES", " INSCRIPCIONES A VARIAS
CARRERAS "},;
    { " SISTEMA DE CONSULTAS ", " CONSULTAS DE ALUMNOS " },;
    { " SALIR " , " SALIDA DEL SISTEMA " }

montamenu(10,25,menú)
menú TO op
DO CASE
CASE OP = 1
  @ 10,20 CLEAR TO 18,52
  RESTORE SCREEN
  DO INSCRIB2
CASE OP = 2
  DO CONSUL
CASE OP = 3
  clear
  quit
  loop
ENDCASE
ENDDO

*SET DEFAULT TO &RUT
*sele 1
*AUX1 := "DIRALUM"
*USE &AUX1 INDEX &AUX1 ALIAS DIRALUM SHARED
*AUX1 := "MG"+&PERIODO
*USE &AUX1 INDEX &AUX1 ALIAS MATGRUP SHARED NEW
*AUX1 := "MINF"+&PERIODO
*USE &AUX1 INDEX &AUX1 ALIAS MATINFO SHARED NEW
*AUX1 := "CTRL"+&PERIODO
*USE &AUX1 INDEX &AUX1 ALIAS CONTROL SHARED NEW
*AUX1 := "INSC"+&PERIODO
*USE &AUX1 INDEX &AUX1 ALIAS INSCALUM SHARED NEW
*AUX1 := "TABLMAT"
*USE &AUX1 INDEX &AUX1 ALIAS TABLA SHARED NEW
*SET COLOR TO &COLOR1
ENDDO

FUNCTION montamenu(ls,cs,menú)
LOCAL retangulo := CHR(213)+CHR(205)+CHR(184)+CHR(179)+;
CHR(190)+CHR(205)+CHR(212)+CHR(179)+CHR(32)
@ls,cs,ls+LEN(menú)+1,cs+LEN(menú[1,1])+1 BOX retangulo
FOR i = 1 TO LEN(menú)
  @ls+i,cs+1 PROMPT menú[i,1]MESSAGE menú[i,2]

```

ESTA TESIS NO DEBE  
SALIR DE LA BIBLIOTECA

NEXT

RETURN NIL

PROCEDURE CONSUL

DO WHILE .T.

  telapadrao(PROCNAME())

  menú := {" DIRECTORIO

          ", "Directorio de los alumnos

  inscritos {"", ;

  inscritos {"HIST.ACADEMICAS", "Historias Academ. de los alumnos

          "}, ;

          {"TIRAS.MATERIAS", "Tiras de Materias anteriores

          "}, ;

          {"PLAN DE ESTUDIOS", "Plan de estudios de la carrera

          "}, ;

          {"F I N", "Regresa al sistema operacional

"} }

  montamenu(6,5,menú)

  menú TO op

  DO CASE

    CASE op = 1

      dic()

  \*     CASE op = 2

  \*     histo()

    CASE op = 3

      tiras()

    CASE op = 4

      plan()

    CASE op = 5

      IF confirma("Abandonar el sistema ") = 1

        EXIT

        CLEAR

      ENDIF

  ENDCASE

ENDDO

FUNCTION menumanut(ls,cs,nome)

  tipo = nome

  menú = {"ARQUITECTURA", "Verificación del &tipo.

          "}, ;

          {"DIS.INDUSTRIAL", "Verificación del &tipo.

          "}, ;

          {"PEDAGOGIA", "Verificación del &tipo.

          "}, ;

          {"PLANIF.", "Verificación del &tipo.

          "}, ;

          {"ECONOMIA", "Verificación del &tipo.

          "}, ;

          {"FIN", "Retorna al menú anterior

"} }

  montamenu(ls,cs,menú)

  RETURN NIL

```

FUNCTION confirma(texto)
LOCAL les, ces, largjan, telavelha
largjan = LEN(texto) + 2
IF largjan < 16
    largjan = 16
ENDIF
ces = (80 - largjan)/2
les = 19
telavelha = SAVESCREEN(les,0,les+4,79)
corantig = SETCOLOR()
cornova = "W/B,N/W"
SET COLCR to &cornova
@les,ces CLEAR TO les+3,ces+largjan
@les,ces TO les+3,ces+largjan DOUBLE
@les+1,ces+2 SAY texto
coluna = (largjan - 9)/2
@les+2,ces+coluna PROMPT "Si"
@les+2,ces+coluna+6 PROMPT "No"
menú TO confirm
SET COLOR TO &corantig
RESTSCREEN(les,0,les+4,79,telavelha)
RETURN(confirm)

```

```

FUNCTION telapadiao(programa)
CLEAR
@00,00 TO 04,79 DOUBLE
* @00,40 TO 03,79 DOUBLE
@05,00,21,79 BOX REPLICATE(CHR(176),9)
@05,00 TO 21,79
@22,00 TO 24,79 DOUBLE
@01,18 SAY empresa1
@02,10 SAY empresa2
@03,20 SAY sistema
@01,73 SAY LEFT(time(),5)
* @02,42 SAY programa
@02,70 SAY DATE()
@23,02 SAY "MENSAJE:"
RETURN NIL

```

```

FUNCTION dic
SAVE SCREEN
DO WHILE .T.
    @02,42 SAY procname()
    menunaut(09,13,"directorio")
    menú TO op
    telavelha = SAVESCREEN(04,01,21,79)
    DO CASE
        CASE op = 1
            * set default to c:\eren\ARQUI
            DIRE()
        CASE op = 2
            * set default to c:\eren\dise
            DIRE()
    
```

```

CASE op = 3
  *set default to c:\eren\ECONO
  DIRE()
CASE op = 4
  * RUT:="PEDAG"
  DIRE()
CASE op = 5
  * RUT:="PLANIF"
  DIRE()
CASE op = 6
  RETURN
  RESTORE SCREEN
ENDCASE

```

```

* SET DEFAULT TO &RUT
* sele 1
* AUX1 := "DIRALUM2"
* USE &AUX1 INDEX &AUX1 ALIAS DIRALUM2 SHARED
ENDDO

```

```

* DESPLIEGA UN DIRECTORIO*
FUNCTION DIRE()
telavelha = SAVESCREEN(00,00,27,79)
USE DIRALUM2
  DECLARE m3{20}, m4{20}
  FOR i = 1 TO 20
    m3[i] = NOMBRE
    m4[i] = RECNO()
  SKIP
  NEXT
  c1 = 01
  l1 = 01
  c2 = 30
  l2 = 22
  CLEAR

  DO WHILE .T.
    @ 11-1, c1-1 TO l2+1, c2+1
    resp = ACHOICE(l1, c1, l2, c2, m3)
    IF resp = 0
      EXIT
    ENDIF

    GOTO m4[resp]
  ASEGREG()
  @ 01, 35 TO 10, 79 DOUBLE
  @ 03, 36 SAY " Nombre....." GET NOMBRE
  @ 05, 36 SAY " Dirección..." GET DIRECCION
  @ 07, 36 SAY " Teléfono..." GET TELEFONO
  READ
  FIELD->NOMBRE:=NOMBRE
  FIELD->DIRECCION:=DIRECCION
  FIELD->TELEFONO:=TELEFONO
  UNLOCK
  @ 01, 35 CLEAR TO 12, 79

```

```

ENDDO
IF LASTKEY()=27
  CLEAR
  RESTSCREEN(00,00,28,79,telavelha)
  RETURN
ENDIF
CLEAR

```

FUNCTION plan

SAVE SCREEN

DO WHILE .T.

```

@02,42 SAY procname()
menumanut(12,13,"Plan de Estudios")
menú TO op
telavelha = SAVESCREEN(00,00,27,79)
DO CASE

```

CASE op = 1

USE PLARQUI

l1=5

c1=5

l2=20

c2=60

@l1-1,c1-1 TO l2+1,c2+1 DOUBLE

DBEDIT(l1,c1,l2,c2)

CLEAR

RESTSCREEN(00,00,28,79,telavelha)

CASE op = 2

USE PLADISE

l1=5

c1=5

l2=20

c2=60

@l1-1,c1-1 TO l2+1,c2+1 DOUBLE

DBEDIT(l1,c1,l2,c2)

CLEAR

RESTSCREEN(00,00,28,79,telavelha)

CASE op = 3

USE PLAPEDAG

l1=5

c1=5

l2=20

c2=60

@l1-1,c1-1 TO l2+1,c2+1 DOUBLE

DBEDIT(l1,c1,l2,c2)

CLEAR

RESTSCREEN(00,00,28,79,telavelha)

CASE op = 4

USE PAPLANIF

l1=5

c1=5

```

l2=20
c2=60
@l1-1,c1-1 TO l2+1,c2+1 DOUBLE
DBEDIT(l1,c1,l2,c2)
CLEAR
RESTSCREEN(00,00,28,79,telavelha)

```

```

CASE op = 5
USE PLAECON
l1=5
c1=5
l2=20
c2=60
@l1-1,c1-1 TO l2+1,c2+1 DOUBLE
DBEDIT(l1,c1,l2,c2)
CLEAR
RESTSCREEN(00,00,28,79,telavelha)
CASE op = 6
RETURN
RESTORE SCREEN

```

```
ENDCASE
```

```
ENDDO
```

```

FUNCTION tiras
SAVE SCREEN
telavelha = SAVESCREEN(00,00,27,79)
DO WHILE .T.

```

```

@02,42 SAY procname()
menumanut(11,13,"Tiras de Materias")
menu TO op

```

```

DO CASE
CASE OP=1
CLEAR SCREEN
USE INSC961
DBEDIT()
CLEAR
RESTSCREEN(00,00,28,79,telavelha)
CASE OP=6
RETURN
RESTORE SCREEN

```

```
ENDCASE
```

```
ENDDO
```

## PROGRAMA DE INSCRIPCIONES (MODULO DE INSCRIPCIONES)

```

static OPC:=1
PERIODO = "'961'"
MESE='ENEFEBMARABRMAYJUNJULAGOSEPOCTNOVDIC'
DECLARE ASIGS [12], AMAT [12], AGPOS [12] //DECLARACION DE ARREGLOS
*DECLARACION DE VARIABLES GLOBALES*
*****
CONSEC:=HORAINIC:=INDMAT:=HORAFIN:=SEGURO:=AUX1:=CLRROR:=CAPGPO:=
COLOR1:=;
CTASIG:=CTA:=ERRORENC:=GENE:=HRINSC:=CHNUM:=IAUX:=INDAUX:=IN:=ASIG
G:=ASIGENC:=;
NACION:=CAP:=ALNUM:=SEX:=MATFIN:=TIPS:=ULTINSC:=HOY:=NUMPRN:=0
CTAS:='.....'
ASIGREN := 05
MRGIZQ := 07
CLEAR
SET DELE ON
SET EXCL OFF
SET BELL OFF
SET SCOR OFF
SET COLOR TO
USUARIO := ''
DO WHILE LEN(USUARIO) = 0
  @ 11,10 SAY ' '
  ACCEPT ' TECLÉE SU NOMBRE POR FAVOR : ' TO USUARIO
  USUARIO := LTRIM(UPPER(USUARIO))
ENDDO
IF .NOT.ISCOLOR()
  COLOR1 := 'GR,N,GR+/G'
  ERRORENC := 'GR+,R+*'
  CLRROR := 'GR+,GR+'
ELSE
  COLOR1 := 'BG,GR+'
  ERRORENC := 'GR+,R+*'
  CLRROR := 'R+,GR+'
ENDIF
DO WHILE .T.
  CLEAR
  @ 7,0,23,78 BOX(REPLICATE(":",9))
  @ 0,0 TO 5,79 DOUBLE
  @ 6,0 TO 24,79 DOUBLE
  @ 1,1,4,78 BOX(REPLICATE(":",9))
  @ 2,17 SAY " UNIVERSIDAD NACIONAL AUTONOMA DE MEXICO "
  @ 3,16 SAY " ESCUELA NACIONAL DE ESTUDIOS PROFESIONALES "
  @ 4,25 SAY " PLANTEL * A R A G O N * "

```

```

@ 8,24 SAY " SISTEMA DE INSCRIPCIONES "
SET WRAP ON
@ 9,26 TO 21,46 DOUBLE
SET MESSAGE TO 23 CENTER
@ 10,30 PROMPT " ARQUITECTURA " MESSAGE " REALIZA INSCRIPCION
DE ARQUITECTURA"
@ 12,30 PROMPT " DISEÑO IND. " MESSAGE " REALIZA INSCRIPCION DE
DISEÑO IND."
@ 14,30 PROMPT " ECONOMIA " MESSAGE " REALIZA INSCRIPCION DE
ECONOMIA "
@ 16,30 PROMPT " PEDAGOGIA " MESSAGE " REALIZA INSCRIPCION DE
PEDAGOGIA "
@ 18,30 PROMPT " PLANIFICACION " MESSAGE " REALIZA INSCRIPCION
DE PLANIFICACION "
@ 20,30 PROMPT " SALIR " MESSAGE " SALIDA DEL SISTEMA "
MENU TO OPC
DO CASE // SELECCIONA EL DIRECTORIO ADECUADO SEGUN LA CARRERA
CASE OPC = 1
RUTT:= "ARQUI"
MATER:=8
ESPACIO:=1
CASE OPC = 2
RUTT:="DISE"
MATER:= 10
ESPACIO:=1
CASE OPC = 3
RUTT:="ECONO"
MATER:=9
ESPACIO:=1
CASE OPC = 4
RUTT:="PEDAG"
MATER:=11
ESPACIO:=1
CASE OPC = 5
RUTT:="PLANIF"
MATER:=9
ESPACIO:=1
CASE OPC = 6
RETURN
RESTORE SCREEN

```

ENDCASE

@ 8,16,14,60 BOX " |J-L|::"

@ 10,26 SAY 'ABRIENDO BASES,'

@ 12,20 SAY 'ESPERE UN MOMENTO POR FAVOR ..... '

INKEY (1)

SET DEFAULT TO &RUTT

sele 1

AUX1 := "DIRALUM"

USE &AUX1 INDEX &AUX1 ALIAS DIRALUM SHARED

AUX1 := "MG"+&PERIODO

USE &AUX1 INDEX &AUX1 ALIAS MATGRUP SHARED NEW

```

AUX1 := "MINF"+&PERIODO
USE &AUX1 INDEX &AUX1 ALIAS MATINFO SHARED NEW
AUX1 := "CTRL"+&PERIODO
USE &AUX1 INDEX &AUX1 ALIAS CONTROL SHARED NEW
AUX1 := "INSC"+&PERIODO
USE &AUX1 INDEX &AUX1 ALIAS INSCALUM SHARED NEW
AUX1 := "TABLMAT"
USE &AUX1 INDEX &AUX1 ALIAS TABLA SHARED NEW
SET COLOR TO &COLOR1

```

```

*CICLO PRINCIPAL DEL PROGRAMA*
*****

```

```

DO WHILE .T.          // DO PRINCIPAL
SELE 1                // DIRALUM
FLAG := 0
ASIGS[1] := '0000' //INDISPENSABLE
CLEAR
@ 0,0 TO 4,79 DOUBLE
INFALUM()           // ESQUELETO INF. DE ALUMNO
LOCCTA()            // LOCALIZA CTA. LEIDA
IF UPPER(CTAS) = 'SALIR'
EXIT
ENDIF
LLENAINF()          // LLENA ESQUELETO INF. ALUMNO
ADEUDOS()           // ADEUDOS DE LIB. O PAPELES
SANCIONES()         // MATS SANCIONADAS U OTRO PROB.
FHORAINSC()         // DIA Y HORA DE INSC.

IF FLAG = 1 .AND. A->INSCRIBIO='A' // HUBO ERROR PARA SU INSC.
TERMINAL()
ENDIF
IF FLAG = 0          // CERO ERRORES, PROCEDE INSC.
@ 7, 0 CLEAR TO 23,79
@ ASIGREN, 0 TO ASIGREN+14, 79
ESQUELETO()         // ESQUELETO DE MATERIAS
INDMAT := 1
INDANT := 0
MATFIN := 0
DO WHILE INDMAT<=MATER .AND. MATFIN = 0 // HACE INSCRIPCION
BEGIN SEQUENCE
INDANT := INDMAT
ASIGENC := 0
SELE 2 // MATERIA-GRUPO
LOCMAT()           // LOCALIZA CVE. DE MAT.
IF ASIGS[INDMAT] = '9999'.OR. ASIGS[INDMAT] = '9876'
IF ASIGS[INDMAT] = '9876' // CANCELA LA INSC
TERMINAL()
ENDIF
MATFIN := 1
BREAK // TERMINA INSC. A MATERIA
ELSE

```

```

CLEAR GETS
@ INDMAT*ESPACIO+ASIGREN,MRGI2Q+52 GET AGPOS[INDMAT] PICTURE
'9999'//LEE GPO
READ
IF AGPOS[INDMAT] = '0000'
  GPOMUEST() // INF. SOBRE GPOS.
ENDIF
OKMATGRUP() // QUE EXISTA MATERIA EN ESE GRUPO
VGT:=SUBSTR((AGPOS[1]),3,1) // PRIMER GRUPO
GT:=SUBSTR((AGPOS[INDMAT]),3,1) // GRUPO ACTUAL
VGT1:=SUBSTR((AGPOS[1]),2,1)
GT1:=SUBSTR((AGPOS[INDMAT]),2,1)
IF RUTT = "ARQUI"
  IF VGT1 >= "7" .AND. GT1 >= "7"
    MENSAJES('SE PUEDE INSCRIBIR TURNO MIXTO')
    GUARDGRUP()
    BREAK
  ENDIF
ENDIF
IF RUTT<>"ECON".AND. RUTT<>"DISE"
  IF (VGT = "0" .OR. VGT = "1") .AND. (GT != "0" .AND. GT !=
"1")
    MENSAJES('NO SE PUEDE INSCRIBIR A GPOS VESPERTINOS')
    ESPERART()
    BREAK
  ENDIF
  IF (VGT = "5" .OR. VGT = "6") .AND. (GT != "5" .AND. GT !=
"6")
    MENSAJES('NO SE PUEDE INSCRIBIR A GPOS MATUTINOS')
    ESPERART()
    BREAK
  ENDIF
ENDIF
GUARDGRUP() // SI HAY CUPO EN GPO; GUARDA LUGAR
ENDIF // FIN DE MAT=9999 O MAT=9876
END // FIN DE BEGIN
ENDDO // FIN DE HACE INSCRIPCION
ENDIF // FIN DE CERO ERRORES
IF INDMAT = 1 .AND. VAL(ASIGS[1]) = 9999
  MENSAJES('EL ALUMNO NO TIENE ASIGNATURAS. INSCRIPCION
CANCELADA')
  TERMINAL()
  FLAG := 1
ENDIF
IF (INDMAT >= 1) .AND. (INDMAT < MATER+1) .AND. (FLAG = 0)
  IF VAL(ASIGS[INDMAT]) = 9876 // CLAVE PARA CANCELAR INSC.
    TERMINAL()
    IF INDMAT > 1 // HAY ALGUNA INSCRIPCION
      SELE 2 // MATERIA-GPO
      INDAUX := 1
      DO WHILE (INDAUX < INDMAT) .AND. (INDAUX < 6)
        MATGPO := ASIGS[INDAUX] + AGPOS[INDAUX]
        FLAG2 := 0
        DO WHILE FLAG2 = 0 // HASTA ENCONTRAR MAT-GPO
          FIND &MATGPO

```

```

IF FOUND()
  FLAG2 := 1
ENDIF
ENDDO
ASEGREG() // BLOQUEA REG.
IF VAL(NALUM) > 0 // AUN HAY CUPO
  CHNUM := LTRIM(STR(VAL(NALUM)-1)) // DECREENTA NALUM, PUES
SE CANCELO
  CHNUM := SUBSTR("000",1,3-LEN(CHNUM))+CHNUM
  FIELD->NALUM:=CHNUM // ACTUALIZA No. DE ALUMNOS
INSCRITOS, YA --
  ENDIF // FIN DE AUN HAY CUPO
UNLOCK
INDAUX ++
ENDDO
ENDIF // FIN DE HAY INSCRIPCION
MENSAJES('INSCRIPCION CANCELADA. ASIGNATURAS ELIMINADAS.')
FLAG := 1
ENDIF // FIN DE CLAVE PARA CANCELAR
ENDIF
SELE 1 // DIRALUM
IF FLAG = 0 // SIN ERROR
  IF INDMAT > 0
    SELE 4 // CONTROL
    CTRL := "CFOLIO"
    FIND &CTRL
    ASEGREG() // BLOQUEA REG.
    CONSEC := LTRIM(STR(VAL(SUBSTR(TEXTO,1,4))+1))
    CONSEC := SUBSTR("0000",1,4-LEN(CONSEC))+CONSEC
    FIELD->TEXTO:=CONSEC // ACTUALIZA ULTIMO FOLIO
    UNLOCK
    SELE 1 // DIRALUM
    ASEGREG() // BLOQUEA REG.
    FIELD->INSCRIBIO := 'I' // MARCA DE YA INSCRITO
    FIELD->ULTINSC := &PERIODO // ACTUALIZA SU ULTIMA
INSCRIPCION
    UNLOCK
    SELE 5 // BASE CON INSCRIPCIONES POR ALUMNO.
    INDMAT--
    ORDTIRA() // ORDENA TIRA POR ASIGNATURA
    FSEG := 0
    DO WHILE FSEG = 0 // DO PARA BLOQUEAR EL ARCHIVO
    IF FLOCK() // DE INSCRIPCIONES HECHAS
      FSEG := 1
    ENDIF
    ENDDO // FIN DEL DO DE BLOQUEO DE ARCH.
    APPEND BLANK
    INDAUX := 1
    DO WHILE INDAUX <= INDMAT .AND. INDAUX < 11
    ICHAR := ALLTRIM(STR(INDAUX))
    AUXMAT := "INSCALUM->MAT" + ICHAR
    AUXGPO := "INSCALUM->GPO" + ICHAR
    &AUXMAT := ASIGS[INDAUX] // ACTUALIZA ASIG.INSCRITA EN INSC

```

```

&AUXGPO := AGPOS[INDAUX] // ACTUALIZA GPO. INSCRITO EN INSC
INDAUX ++
ENDDO
E->CUENTA := A->CUENTA // ACTUALIZA CTA. EN INSC
E->NOMBRE := A->NOMBRE // ACTUALIZA NOMBRE EN INSC
E->D := A->D // ACTUALIZA DIG. EN INSC
E->GEN := A->GEN // ACTUALIZA GEN EN INSC
E->FOLIO := CONSEC // ACTUALIZA No DE FOLIO EN INSC
E->TIMEINI := SUBSTR(HORAINIC,1,2)+SUBSTR(HORAINIC,4,2) ;
           + SUBSTR(HORAINIC,7,2)
HORAFIN := TIME()
E->TIMEFIN := SUBSTR(HORAFIN,1,2)+SUBSTR(HORAFIN,4,2) ;
           + SUBSTR(HORAFIN,7,2)
FECH := DTC(DATE())
E->FECHA := SUBSTR(FECH,4,2)+SUBSTR(FECH,1,2) ;
           + SUBSTR(FECH,7,2)
E->ATENDIO := USUARIO // ACTUALIZA ATENDIO EN INSC
UNLOCK // LIBERA EL ARCHIVO DE INSCRIPCION
MENSAJES('NUMERO DE FOLIO DE INSCRIPCION : ' ;
         +LTRIM(STR(VAL(CONSEC)))+ ' ')
@ 23,0
inkey(0)

TIRAPRINT() // IMPRIME TIRA DE MATS. PROVISIONAL
ENDIF
ELSE // SI HAY ERROR
@ 23,0
ACCEPT ' OPRIMA RETURN (ENTER) PARA
CONTINUAR ... ' TO XX
CLEAR GETS
ENDIF // FIN DE NO HAY ERROR
DBCMMITALL() // IGUAL QUE COMMIT, PERO EN TODAS LAS AREAS
UNLOCK ALL // LIBERA TODO LO BLOQUEADO
ENDDO // FIN DE DO PRINCIPAL
ENDDO // FIN DEL DO DEL MENU

SET TALK ON
SET STATUS ON
SET BELL ON

* CODIGO DE FUNCIONES *
*****

* MENSAJES DE ERROR O AVISO *
*****
FUNCION MENSAJES()
PARAMETERS MSGS
SET COLOR TO &CLRROR
??CHR(07)
@ ASIGREN+15, 0 TO ASIGREN+17, 79
MENS:=LEN(MSGS)
CENTRO:=INT((78-MENS)/2)
@ ASIGREN+16, CENTRO SAY '&MSGs.'
SET COLOR TO &COLOR1
RETURN(NIL)

```

```

* ESPERA HASTA OPRIMIR BARRA ESPACIADORA *
*****
FUNCTION ESPERART
  ESCAPE := '.'
  DO WHILE ESCAPE <> CHR(32)
    @ 23,20 SAY 'OPRIMA LA BARRA ESPACIADORA POR FAVOR ... ' GET
  ESCAPE
  READ
  IF ESCAPE <> CHR(32)
    ??CHR(7)
  ENDIF
ENDDO
@ ASIGREN+15, 0 CLEAR TO ASIGREN+18, 79 // BORRA ULTIMO MENSAJE
RETURN(NIL)

* AUTORIZA INSCRIPCION EN GRUPOS SATURADOS*
*****
FUNCTION AUTORIZA
  CVE1:='XXXX'
  BANDERA:= 1
  TECLA:=SPACE(1)
  P1:=SAVESCREEN(10,24,12,54)
  DO WHILE BANDERA=1
    @ 23,20 SAY 'OPRIMA LA BARRA ESPACIADORA POR FAVOR ... ' GET
  TECLA
  READ
  IF LASTKEY() <> 32
    ??CHR(7)
  ELSEIF LASTKEY()=32
    BANDERA:=2
  ENDIF
  IF READKEY() = 12
    @10,24 CLEAR TO 12,54
    @10,24 TO 12,54 DOUBLE
  SET COLOR TO W+*/N,W/W
  @11,25 SAY 'CLAVE DE AUTORIZACION : ' GET CVE1 PICT 'XXXX'
  READ
  IF CVE1 = 'AMOR'
    CUP:=LTRIM(STR(VAL(B->CUPO)+1))
    CUP:=SUBSTR("000",1,3-LEN(CUP))+CUP
    REPLACE B->CUPO WITH CUP
    KEYBOARD CHR(13) + CHR(13)
  ELSE
    @11,25 SAY ' AUTORIZACION INVALIDA '
    INKEY(3)
    KEYBOARD CHR(13) + CHR(13)
  ENDIF
  BANDERA:=2
  ENDIF
ENDDO
SETCOLOR(COLOR1)

```

```

RESTSCREEN(10,24,12,54,P1)
@ 23,20 CLEAR TO 23,78
RETURN(NIL)

```

```

* ESQUELETO DE INF. DE ALUMNO *
*****

```

```

FUNCTION INFALUM

```

```

a:=100
dot:=chr(250)
DO WHILE a < 1821
  row:=a/80+5
  col:=mod(a,80)+5
  @ row,col say "&dot."
  a:=a+30
ENDDO
@ 20,18 TO 22,63
@ 21,19 SAY 'TECLEE < SALIR > PARA ABANDONAR EL SISTEMA'
@ 1, 4 SAY "CUENTA"
@ 1, 18 SAY "-"
@ 1, 58 SAY "NAC"
@ 1, 67 SAY "SEXO"
@ 3, 7 SAY "GEN"
@ 3, 15 SAY "ULTINSC"
@ 3, 29 SAY "HOY ES:"
@ 3, 44 SAY "FECINSC"
@ 3, 59 SAY "HORA_INSC"
CLEAR GETS
RETURN(NIL)

```

```

* LOCALIZA CUENTA LEIDA *
*****

```

```

FUNCTION LOCCTA

```

```

SELE 1 // DIRALUM
FLAG3 := 'NO'
DO WHILE FLAG3 <> 'SI'
  CTAS := '.....'
  @ 1, 11 GET CTAS // PIDE CUENTA
  READ
  @ ASIGREN+15, 0 CLEAR TO ASIGREN+17, 79
  IF UPPER(CTAS) = 'SALIR'
    CLOSE ALL
    RETURN
  ENDIF
  FIND &CTAS // BUSCA LA CUENTA
  IF FOUND() // SI LA ENCONTRO
    DIGI := ' '
    @ 1, 19 GET DIGI
    READ
    IF A->D <> DIGI // EL DIGITO ES DIFERENTE
      MENSAJES('ERROR EN DIGITO VERIFICADOR')
    ELSE // EL DIGITO VERIFICADOR ES CORRECTO
      FLAG3 := 'SI' // CUENTA CORRECTA
    ENDIF // FIN DE VERIFICACION DE DIGITO
  ELSE // NO ENCONTRO EL NUMERO DE CUENTA

```

```

MENSAJES('ALUMNO NO REGISTRADO EN DIRECTORIO')
ENDIF // FIN DE SI LA ENCONTRO
ENDDO // FIN DE BUSQUEDA DEL ALUMNO
RETURN(NIL)

```

```
* LLENA EL ESQUELETO DE INF.DEL ALUMNO *
```

```

*****
FUNCTION LLENAINF
@ ASIGREN+15, 0 CLEAR TO ASIGREN+18, 79 // BORRA ULTIMO MENSAJE
@ 1, 23 GET A->NOMBRE
HORAINC := TIME()
NACION:=IF((A->NAC='1'),'MEX',;
            IF((A->NAC='2'),'EXT','---'))
@ 1,62 GET NACION
SEX := IF((A->SEXO='F'.OR. A->SEXO='2'),'FEM',;
          IF((A->SEXO='M'.OR. A->SEXO='1'),'MAS','---'))
@ 1, 72 GET SEX
GENE := IF(A->GEN=' ','--',A->GEN)
@ 3, 11 GET GENE
ULTINSC := IF(A->ULTINSC=' ','---',A->ULTINSC)
@ 3, 23 GET ULTINSC
MES := SUBSTR(MESE,VAL(SUBSTR(DTOC( DATE() ),1,2))*3-2,3)
DIA := SUBSTR(DTOC( DATE() ),4,2)
HOY := DIA + '-' + MES
@ 3, 36 GET HOY
IF A->MDINSC = ' '
FECHINS = '-----'
ELSE
DIA := SUBSTR(A->MDINSC,3,2)
MES := SUBSTR(MESE,VAL(SUBSTR(A->MDINSC,1,2))*3-2,3)
FECHINS = DIA + '-' + MES
ENDIF
@3, 52 GET FECHINS
DO CASE
CASE A->HORAINSC = '1'
HRINSC= '09:00'
CASE A->HORAINSC = '2'
HRINSC= '10:00'
CASE A->HORAINSC = '3'
HRINSC= '11:00'
CASE A->HORAINSC = '4'
HRINSC= '12:00'
CASE A->HORAINSC = '5'
HRINSC= '16:00'
CASE A->HORAINSC = '6'
HRINSC= '17:00'
CASE A->HORAINSC = '7'
HRINSC= '18:00'
CASE A->HORAINSC = '8'
HRINSC= '19:00'
OTHERWISE
HRINSC='-----'

```

```

ENDCASE
@ 3, 69 GET HRINSC
RETURN(NIL)

```

```

* LISTA DE CUENTAS CON ALGUN ADEUDO *
*****

```

```

FUNCTION ADEUDOS
SELE 4 // CONTROL
CTA := "D"+A->CUENTA
FIND &CTA
VAR1:=8
VAR4:=14
IF FOUND() // TIENE PROBLEMA DE INSC.
  FLAG := 1
  @ 6, 0 TO 21, 79
  AUX1:=1
  SET COLOR TO &CLRORR
  ??CHR(07)
  CTA1:=.T.
  DO WHILE CTA1
    VAR2:=VAR3:=.F.
    TIPS := VAL(TIPO)

    DO CASE // PROBLEMA SEGUN TIPO
      CASE TIPS = 0
        @ 8, 7 SAY 'NO ESTA DEFINIDO EL PROBLEMA'
      CASE TIPS = 3
        SET KEY -9 TO BIB
        @ 8, 7 SAY 'ADEUDO EN SECCION ESCOLAR'
        VAR2:=.T.
      CASE TIPS = 4
        @ 14, 7 SAY 'ADEUDO EN BIBLIOTECA'
        VAR3:=.T.
        SET KEY -9 TO BIB
      OTHERWISE
        @ 8,7 SAY 'ERROR DESCONOCIDO EN LISTA NEGRA'
    ENDCASE
    IF AUX1 = 1
      IF VAR2 =.T.
        @ ++VAR1,10 SAY TEXTO // ESPECIFICA MAS EL PROBLEMA
      ELSEIF VAR3 = .T.
        @ ++VAR4,10 SAY TEXTO
      ENDIF
    ENDIF
    SKIP
    CTA2 := D->CUENTA
    IF !(CTA2= A->CUENTA)
      CTA1:=.F.
    ENDIF
  ENDDO
  SET COLOR TO &COLOR1
ENDIF // FIN DE TIENE PROBLEMA DE INSC.
RETURN(NIL)

```

\* CUENTAS CON MATERIAS SANCIONADAS U CON OTRO PROBLEMA \*  
 \*\*\*\*\*

FUNCTION SANCIONES

```

SELE 1 // DIRALUM
IF INSCRIBIO = 'I'
  MENSAJES('EL ALUMNO YA SE INSCRIBIO')
  FLAG := 1
ENDIF
IF INSCRIBIO = 'A'
  MENSAJES('EL ALUMNO SE ESTA INSCRIBIENDO EN OTRA TERMINAL')
  FLAG := 1
ENDIF
IF FLAG = 0 // CERO ERRORES
  IF CLASINSC<>'5' // SIN DERECHO A INSCRIPCION
    FLAG := 1
    @ 6, 0 TO 10, 79
    SET COLOR TO &CLRROR
    ??CHR(07)
    DO CASE // POR QUE NO TIENE DERECHO A INSCRIPCION
      CASE CLASINSC = ' '
        @ 8, 10 SAY 'SIN INFORMACION DE CLASIFICACION'
      CASE CLASINSC = '1'
        @ 8, 10 SAY 'ALUMNO CON CARRERA TERMINADA'
      CASE CLASINSC = '2'
        @ 8, 10 SAY 'ALUMNO AFECTADO POR ARTICULO 19'
      CASE CLASINSC = '4'
        @ 8, 10 SAY 'SIN TIRA DE MATERIAS EN EL SEMESTRE PASADO
        (IRREGULAR)'
      OTHERWISE
        @ 8, 10 SAY 'ERROR EN CODIGO DE CLASIFICACION'
    ENDCASE // FIN DE POR QUE SIN INSCRIPCION
    SET COLOR TO &COLOR1
  ENDIF // FIN DE SIN DERECHO A INSCRIPCION
ENDIF // FIN DE CERO ERRORES
RETURN(NIL)

```

\* REvisa su dia y hora de INSC. \*  
 \*\*\*\*\*

FUNCTION FHORAINSC

```

SELE 4 // CONTROL
FECHHOY := DATE()
MDH := MONTH(FECHHOY) * 100 + DAY(FECHHOY)
IF (VAL(A->MDINSC)) > (MDH)
  MENSAJES('NO LE CORRESPONDE INSCRIBIRSE EL DIA DE HOY')
  FLAG := 1
ELSE
  KEY := 'CTOLHORA'
  FIND &KEY
  HRTOLER := VAL(SUBSTR(TEXTO,1,4)) // LEE DE BASE TOLERANCIA DE
INSC
  HOR := TIME()
  HOR := (VAL(SUBSTR(HOR,1,2))*60)+(VAL(SUBSTR(HOR,4,2)))

```

```

IF (HOR < ((VAL(SUBSTR(HRINSC,1,2))*60)-HRTOLER))
  MENSAJES('TODAVIA NO ES HORA DE SU INSCRIPCION (REGRESAR MAS
TARDE)')
  FLAG := 1
ENDIF
IF (A->INSCRIBIO = ' ') // SI NO SE HABIA INSCRITO . . .
  AREA := SELECT()
  SELE 1
  ASEGREG()
  FIELD->INSCRIBIO:='A' // MARCA DE QUE LO ESTA HACIENDO
AHORA
  UNLOCK
  SELE &AREA
ENDIF
RETURN(NIL)

```

```

* ESQUELETO DE INSC. EN MATERIAS *
*****

```

```

FUNCTION ESQUELETO

```

```

  INDMAT:=0

```

```

  @ 6,1,18,78 BOX(REPLICATE("X",9))

```

```

  DO WHILE INDMAT<MATER // LLENA ESQUELETO DE INSCRIPCION

```

```

    INDMAT ++

```

```

    ICHAR := ALLTRIM(STR(INDMAT))

```

```

    ICHAR := PADL(ICHAR,2,"0")

```

```

    ASIGS[INDMAT] := 'AS' + ICHAR

```

```

    AGPOS[INDMAT] := 'GP'+ICHAR

```

```

    @ INDMAT*ESPACIO+ASIGREN, MRGIZQ+5 SAY ASIGS[INDMAT]

```

```

    @ INDMAT*ESPACIO+ASIGREN, MRGIZQ+52 SAY AGPOS[INDMAT]

```

```

  ENDDO // FIN DE LLENA ESQUELETO DE INSC

```

```

indmat*ESPACIO+asigren, mrgizq+52, indmat*ESPACIO+asigren, mrgizq+56
box(replicate("X",9))

```

```

  RETURN(NIL)

```

```

* LOCALIZA CVE. DE MATERIAS Y VE SI PUEDE INSCRIBIRSE A ELLA *
*****

```

```

FUNCTION LOCMAT

```

```

  DO WHILE ASIGENC = 0 // PIDE LA MATERIA A INSCRIBIR

```

```

    CLEAR GETS

```

```

    @ INDMAT*ESPACIO+ASIGREN,MRGIZQ+5 GET ASIGS[INDMAT] PICTURE
'9999'// LEE MATERIA

```

```

    READ

```

```

    @ ASIGREN+15, 0 CLEAR TO ASIGREN+17, 79 // BORRA ULTIMO
MENSAJE

```

```

    ASIG := UPPER(ASIGS[INDMAT])

```

```

    if indmat = MATER .and. (asig != "9999" .and. asig != "9876")

```

```

      mensajes("SOLO CLAVE DE CANCELAR o ACEPTAR INSCRIPCION")

```

```

      esperart()

```

```

      loop

```

```

    endif

```

```

  IF ASIG = '9999' .OR. ASIG = '9876'

```

```

EXIT // TERMINA INSC CON 9999. CON 9876 LA CANCELA
ENDIF
FIND &ASIG // BUSCA MATERIA (EN MATGRUP)
IF FOUND() // ENCONTRO LA MATERIA
  AREA:=SELECT()
  SELE 6
  FIND &ASIG // PUNTERO EN CVE DE ASIG. EN TABLA
  @ INDMAT*ESPACIO+ASIGREN, MRGIZQ+13 GET NOMMAT //
DESPLIEGA NOMBRE DE MAT
  AMAT[INDMAT] := NOMMAT
  CLEAR GETS
  SELE &AREA

  IF INDMAT>1 // HAY OTRAS MATS. INSCRITAS
  INDAUX := 1
  ASIGREP := .F.
  DO WHILE INDAUX < INDMAT // COMPARA MATERIA CON ANTERIORES
    IF ASIGS[INDAUX] = ASIGS[INDMAT] // MATERIA YA INSCRITA
      MENSAJES('YA ESTA INSCRITO EN ESTA ASIGNATURA')
      ESPERART()
      ASIGREP := .T.
      EXIT // YA NO COMPARA CON ANTERIORES
    ENDIF
    INDAUX ++
  ENDDO // FIN DE COMPARA CON MATERIAS ANTERIORES
  IF ASIGREP // SI MATERIA ESTA REPETIDA
    @ INDMAT*ESPACIO+ASIGREN, MRGIZQ+13 SAY REPLICATE("✖",36)
  // BORRA MAT ANTERIOR
  LOOP // VA A PEDIR OTRA
  ENDIF
  ENDIF // FIN DE HAY OTRAS MATS. INSCRITAS
  SELE 3 // INF. DE MATERIAS SANCIONADAS
  CTASIG := A->CUENTA + ASIGS[INDMAT]
  FIND &CTASIG // BUSCA MATERIA APROBADA O SANCIONADA
  IF FOUND() // SI ESTA APROBADA O SANCIONADA
  IF INDCOD = '1' // MATERIA APROBADA
    MENSAJES('LA ASIGNATURA YA ESTA APROBADA')
    ESPERART()
  ELSE // MATERIA SANCIONADA
    MENSAJES('YA TIENE DOS INSCRIPCIONES A ESTA ASIGNATURA')
    ESPERART()
  ENDIF // FIN DE MATERIA APROBADA O SANCIONADA
  SELE 2 // MATERIA-GRUPO
  LOOP // VA A PEDIR OTRA
  ENDIF // FIN DE SI APROBADA O SANCIONADA
  SELE 2 // MATERIA-GRUPO
  EXIT // VA A VERIFICAR EL GRUPO
  ELSE // MATERIA NO ENCONTRADA
  @ INDMAT*ESPACIO+ASIGREN,MRGIZQ+13 SAY REPLICATE("✖",36) //
BORRA MAT ANTERIOR
  MENSAJES('ERROR EN CVE. DE ASIGNATURA, FAVOR DE CHECAR')
  ESPERART()
  LOOP // VA A PEDIR OTRA

```

```

    ENDIF          // FIN DE ENCONTRO LA MATERIA
  ENDDO          // FIN DE PIDE LA MATERIA A INSCRIBIR
RETURN(NIL)

```

```
* BLOQUEA UN REG. PARA ACTUALIZARLO *
```

```
*****
```

```

FUNCTION ASEGREG
  SEGURO := 0
  DO WHILE SEGURO = 0
    IF RLOCK()
      SEGURO := 1
    ENDIF
  ENDDO
RETURN(NIL)

```

```
* MUESTRA INF. DE GRUPOS Y CUPOS *
```

```
*****
```

```

FUNCTION GPOMUEST
  A:=SAVESCREEN(5,0,19,79)
  BOXDESCD(5,0,19,79)
  @ 5,0 TO 19,79 DOUBLE
  @ 7,1 TO 7,78
  DECLARE INF[30] // MAXIMO 30 GPOS POR MATERIA
  I:=0
  SET COLOR TO &CLROR
  @ 6,1 SAY ' GPO. CUPO          NOMBRE DE PROFESOR
HORARIO'
  SET COLOR TO W+*/N
  @ 6,13 SAY 'DISP.'
  SET COLOR TO &COLOR1
  DO WHILE ASIG = CVEMAT .AND. (.NOT.EOF())
    I++
    CAP:=LTRIM(STR(VAL(CUPO)-VAL(NALUM)))
    CAP:=SUBSTR("00",1,2-LEN(CAP))+CAP
    ACHO:=' '+B->GRUPO+' '+CUPO+' '+CAP+' '+B->NOMPROF+'
'+B->HORARIO
    INF[I]:='&ACHO.' // UNA SOLA CADENOTA EN ACHO PARA USAR
ACHOICE
  SKIP
  ENDDO
  MSG:='INF. DE LA ASIGNATURA : ' + RTRIM(F->NOMMAT)
  MENSAJES('&MSG.')
  @ 23,7 SAY 'Grupos = Flechas Arriba - Abajo          Salir = ENTER
(RETURN)'
  ASIZE(INF,++I) // MATRIZ INF SOLO CON ELEMENTOS CON DATOS
  ACHOICE(8,1,18,78,INF,"","FUNACHO")
  CLOSEDESCD(5,0,19,79)
  RESTSCREEN(5,0,19,79,A)
  @ ASIGREN+15,0 CLEAR TO ASIGREN+18,79
  BREAK
RETURN(NIL)

```

```
* CONTROL DE ACHOICE *
```

```

*****
FUNCTION FUNACHO
PARAMETERS MODO
ULT:=LASTKEY() // ULTIMA TECLA PRESIONADA
IF (MODO=0.OR.MODO=1.OR.MODO=2)
  RETURN(2)
ENDIF
IF ULT=19 .OR. ULT=4 // FLECHA IZQ O DER
  RETURN(2)
ENDIF
IF MODO=3
  DO CASE
    CASE ULT=1 // HOME
      RETURN(2)
    CASE ULT=6 // END
      RETURN(2)
    CASE ULT=13 // ENTER
      RETURN(0)
    CASE ULT=27 // ESCAPE
      RETURN(2)
    OTHERWISE
      RETURN(2)
  ENDCASE
ENDIF
RETURN(NIL)

* VE QUE EXISTA LA MATERIA EN EL GRUPO *
*****
FUNCTION OKMATGRUP
MATGPO := ASIG+UPPER(AGPOS(INDMAT))
FIND &MATGPO // BUSCA RELACION ASIG CON GPO EN MAT-GPO
IF .NOT.FOUND() // NO HAY RELACION
  MENSAJES('NO SE IMPARTE LA ASIGNATURA EN ESTE GRUPO')
  ESPERART()
  BREAK // VA A PEDIR MAT OTRA VEZ
ENDIF // FIN DE NO HAY RELACION
CLEAR GETS
RETURN(NIL)

* SI HAY CUPO, GUARDA LUGAR EN EL *
*****
FUNCTION GUARDGRUP
ASEGREG()
CHNUM := LTRIM(STR(VAL(CUPO)-VAL(NALUM)))
CHNUM := SUBSTR("000",1,3-LEN(CHNUM))+CHNUM
CAPGPO := VAL(CUPO)-VAL(NALUM)
IF (CAPGPO>0) // SI HAY CUPO
  CHNUM := LTRIM(STR(VAL(NALUM)+1))
  CHNUM := SUBSTR("000",1,3-LEN(CHNUM))+CHNUM
  FIELD->NALUM := CHNUM // ACTUALIZA NO DE INSCRITOS EN EL GRUPO
  UNLOCK
ELSE // NO HAY CUPO

```

```

MENSAJES('YA NO HAY CUPO EN ESTE GRUPO')
AUTORIZA()
UNLOCK
BREAK // VA A PEDIR OTRA VEZ MATERIA
ENDIF // FIN DE SI HAY CUPO
INDMAT ++
UNLOCK
RETURN(NIL)

```

```
*ORDENA LA TIRA DE MATERIAS*
```

```
*****
```

```
FUNCTION ORDTIRA
```

```

IF INDMAT > 1
  FOR II = 1 TO INDMAT-1
    FOR JJ = II+1 TO INDMAT
      IF ASIGS[II] > ASIGS[JJ]
        A1 := ASIGS[II]
        A2 := AMAT[II]
        A3 := AGPOS[II]
        ASIGS[II] := ASIGS[JJ]
        AMAT[II] := AMAT[JJ]
        AGPOS[II] := AGPOS[JJ]
        ASIGS[JJ] := A1
        AMAT[JJ] := A2
        AGPOS[JJ] := A3
      ENDIF
    NEXT JJ
  NEXT II
ENDIF
RETURN(NIL)

```

```
* IMPRIME TIRA DE MATERIAS PROVISIONAL *
```

```
*****
```

```
FUNCTION TIRAPRINT
```

```

* IF:CAPTURE NB NFF Q=Q0 TI=1
** SET DEVI TO PRINT
** SETPRC(0,0)
clear
@ 0,0 SAY SUBSTR(E->FECHA,1,2) + '/' + ;
      SUBSTR(MESE,VAL(SUBSTR(E->FECHA,3,2))*3-2,3) + '/' + ;
      SUBSTR(E->FECHA,5,2)
@ 0,63 SAY E->FOLIO
@ 4,1 SAY A->PLANTEL
@ 4,6 SAY A->CARR
@ 4,10 SAY E->CUENTA + '-' + E->D
@ 4,22 SAY E->NOMBRE
@ 4,57 SAY E->GEN
@ 4,61 SAY SUBSTR(PERIODO,2,2) + '-' + SUBSTR(PERIODO,4,1)
IAUX := 1
SELE B // MATERIA-GPO
DO WHILE IAUX <= INDMAT .AND. IAUX < 11
  ICHAR := ALLTRIM(STR(IAUX))
  AUX1 := 'E->MAT' + ICHAR
  AUX2 := 'E->GPO' + ICHAR

```



```

@ 11,25 CLEAR TO 11,55
SET COLOR TO W+*/N,W/W
@ 11,25 SAY ' CLAVE DE AUTORIZACION : ' GET CVE2 PICT 'XXXX'
READ
IF CVE2 = 'PUMA'
  CTA := "D"+A->CUENTA
  SELE 4
  FIND &CTA
  IF FOUND() // TIENE PROBLEMA DE INSC.
    CTA1:=.T.
    DO WHILE CTA1
      IF VAL(TIPO)=BS
        ASEGREG()
        DELETE
        UNLOCK
      ENDIF
      SKIP
      CTA2 := D->CUENTA
      IF !(CTA2= A->CUENTA)
        CTA1:=.F.
      ENDIF
    ENDDO
  ENDIF
ELSE
  @11,25 CLEAR TO 11,55
  @11,25 SAY ' AUTORIZACION INVALIDA '
  INKEY(3)
ENDIF
KEYBOARD CHR(13)
SETCOLOR(COLOR1)
RESTSCREEN(10,24,12,56,P1)
@ 23,20 CLEAR TO 23,78
SET KEY -9 TO
RETURN(NIL)

```

\* ABRE CAJA HACIA ABAJO \*

\*\*\*\*\*

```

FUNCTION BOXDESCD(Lin1,col1,lin2,col2)
PUBLIC TELABOX ; SAVE SCREEN TO TELABOX
FOR CONTA=LIN1 TO LIN2
  @ LIN1,col1 TO CONTA,col2 double
  @ LIN1+1,col1+1 CLEAR TO CONTA-1, col2-1
  INKEY(.01)
NEXT
RETURN (NIL)

```

\* CIERRA CAJA HACIA ARRIBA \*

\*\*\*\*\*

```

FUNCTION CLOSEDESCD(LIN1,COL1,LIN2,COL2)
FOR CONTA=LIN2 TO LIN1 STEP-1
  @ conta,col1-2 CLEAR TO conta, col2+2
  INKEY(.01)
NEXT
RETURN (NIL)

```

## **BIBLIOGRAFIA**

**BIBLIOGRAFIA****DISEÑO DE SISTEMAS DE INFORMACION**

Teoría y Práctica  
John G. Burch Gary Grudnitski  
Ed. Megabyte.

**INGENIERIA DEL SOFTWARE**

Un enfoque Práctico.  
Roger S. Pressman  
Tercera Edición 1993  
Ed. Mc Graw Hill

**CLIPPER 5.01 Básico**

Ramalho  
Ed. Mc Graw Hill

**COMUNICACION Y REDES DE PROCESAMIENTO DE DATOS**

Néstor Gonzalez Sainz  
Ed. Mc. Graw Hill

**SEMINARIO DE CONECTIVIDAD**

Novellco de México

**MANUAL DEL SISTEMA OPERATIVO NETWARE 386**

Netware ver. 3.11.