



UNIVERSIDAD NACIONAL AUTONOMA DE MEXICO

FACULTAD DE ESTUDIOS SUPERIORES CUAUTITLAN

**“DISEÑO DE UNA APLICACION CON ARQUITECTURA
CLIENTE/SERVIDOR PARA UN CENTRO
DE CAPACITACION”**

**TRABAJO DE SEMINARIO :
REDES DE COMPUTADORAS
QUE PARA OBTENER EL TITULO DE:
LICENCIADA EN INFORMATICA
P R E S E N T A:
SONIA INFANTE GALINDO**

ASESOR: ING. JESUS MOISES HERNANDEZ DUARTE

CUAUTITLAN IZCALLI, EDO. DE MEX.

1997

**TESIS CON
FALLA DE ORIGEN**



Universidad Nacional
Autónoma de México



UNAM – Dirección General de Bibliotecas
Tesis Digitales
Restricciones de uso

DERECHOS RESERVADOS ©
PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL

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

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



FACULTAD DE ESTUDIOS SUPERIORES CUAUTITLAN
 UNIDAD DE LA ADMINISTRACION ESCOLAR
 DEPARTAMENTO DE EXAMENES PROFESIONALES

U. N. A. M.
 FACULTAD DE ESTUDIOS
 SUPERIORES-CUAUTITLAN



DEPARTAMENTO DE
 EXAMENES PROFESIONALES

DR. JAIME KELLER TORRES
 DIRECTOR DE LA FES-CUAUTITLAN
 PRESENTE.

AT'N: ING. RAFAEL RODRIGUEZ CEBALLOS

Jefe del Departamento de Exámenes
 Profesionales de la FES-C.

Con base en el art. 51 del Reglamento de Exámenes Profesionales de la FES-Cuautitlan, nos permitimos comunicar a usted que revisamos el Trabajo de Seminario:

Redes de Computadoras. Diseño de una aplicación con
Arquitectura Cliente/Servidor para un centro de
capacitación.

que presenta la pasante: Sonia Infante Galindo
 con número de cuenta: 9003989-2 para obtener el Título de:
licenciada en Informática

Considerando que dicho trabajo reúne los requisitos necesarios para ser discutido en el EXAMEN PROFESIONAL correspondiente, otorgamos nuestro VISTO BUENO.

ATENTAMENTE.

"POR MI RAZA HABLARA EL ESPIRITU"

Comité Interz, Edo. de México, a 15 de Octubre de 19 97

MODULO:	PROFESOR:	FIRMA:
<u>III</u>	<u>Ing. J. Moisés Hernández Duarte</u>	<u>[Firma]</u>
<u>III</u>	<u>M. I. Gloria Ponce Venegas</u>	<u>[Firma]</u>
<u>IV</u>	<u>Ing. Francisco Chavez Castañeda</u>	<u>[Firma]</u>

Gracias a Dios:
Por darme la oportunidad

A mi Madre:
Por su amor, sacrificio y
apoyo.

A mi Padre:
Por su cariño.

A mis Hermanos:
Nancy, Héctor, Raúl,
José Juan y Minerva,
por su cariño, y
apoyo.

A mi Sobrina:
Jessica, por que
la quiero.

A mis Amigos:
Por su gran apoyo y
amistad.

A mi Asesor:
Ing. J. Moisés Hernández
D. por su gran ayuda,
paciencia, y por ser una
gran persona.

A los Ingenieros:
M.I. Gloria Ponce V.
y al Ing. Francisco
Chávez C. por su
gran ayuda.

A los profesores:
Lic. Rosa Guadalupe Valadez
Ing. Sergio Acosta, Lic. Héctor
Sánchez, Lic. Araceli Nivón
Zagui y Lic. Carlos Pineda por
sus enseñanzas, paciencia,
cariño y formación, y por la
gran admiración que siento
por ellos

AGRADECIMIENTO:

A la Universidad Nacional Autónoma de México
por la oportunidad que me dió de estudiar
y por la formación que me brindó, así como
a todos los profesores que contribuyeron a este fin.

OBJETIVOS

Objetivo general:

1.- Diseñar una aplicación para un centro de capacitación en una arquitectura Cliente/Servidor

Objetivos específicos:

- 1.- Subrayar la eficiencia del sistema operativo Windows NT en la arquitectura Cliente/Servidor, así como de las herramientas de diseño y desarrollo de sistemas : ERWIN, Visual Basic y SQL Server.
- 2.- Exponer el diseño que se utilizará para la implantación del sistema SACE.

ÍNDICE

INTRODUCCIÓN	1
I.- GENERALIDADES	4
I.1 Conceptos y características	4
I.1.1 Redes de Computadoras	4
I.1.2 ERWIN	5
I.1.3 Windows NT	5
I.1.4 SQL SERVER.	14
I.1.4 Visual Basic.	16
I.1.5 SACE	17
II.- LA ARQUITECTURA CLIENTE/SERVIDOR	20
II.1 Definición	22
II.2 ¿Cómo funciona un sistema Cliente/Servidor?	25
II.3 Funciones de los componentes	29
II.3.1 Función del Cliente	29
II.3.2 Función del Servidor	29
II.3.4 Función de la Red	29
II.4 Modelos de la Arquitectura	30
II.4.1 Presentación Distribuida	30
II.4.2 Presentación Remota	30
II.4.3 Distribución Lógica	30
II.4.4 Datos Remotos	31
II.4.5 Datos Distribuidos	31

III.- LA ARQUITECTURA CLIENTE/SERVIDOR JUNTO CON WINDOWS NT	33
III.1 Características de la combinación Cliente/Servidor - Windows NT	33
III.2 ¿ Por qué Usar Visual Basic y SQL Server para un sistema Cliente/Servidor?	37
IV.- DISEÑO DE UNA APLICACIÓN CON ARQUITECTURA CLIENTE/SERVIDOR BAJO WINDOWS NT, UTILIZANDO VISUAL BASIC Y SQL SERVER	39
IV.1 Flujo Organizacional	40
IV.2 El Modelo de Procesos	41
IV.3 Diseño Lógico	43
IV.4 Diseño de la Red	45
IV.5 Diseño Físico	47
CONCLUSIONES	52
BIBLIOGRAFÍA	54

INTRODUCCIÓN

Las computadoras y redes informáticas han producido en nuestra sociedad un gran impacto. Estas herramientas han multiplicado la productividad y eficacia del trabajo, tanto para las empresas como para los usuarios individuales. Cada día, millones de personas acuden a las redes de computadoras para llevar a cabo sus actividades personales o de trabajo, y cada día va creciendo más el uso de estas redes.

Las redes de computadoras tienen una gran cantidad de usos, como por ejemplo: operaciones bancarias, operaciones comerciales, reservación de hoteles, etc. Sin embargo, hay que mencionar que las computadoras como las redes con que se comunican tienen el mismo grado de importancia. Las redes tienen como finalidad transferir e intercambiar datos entre ordenadores y terminales.

El trabajo que presento es el diseño de una aplicación en red que tiene como función principal llevar el control de un centro educacional en donde se da capacitación a personas, en su mayoría, de otras organizaciones. El diseño está elaborado para una arquitectura Cliente/Servidor, el sistema operativo de red que propongo es Windows NT, la herramienta de diseño que utilicé es ERWIN y las herramientas de desarrollo que presento son Visual Basic y SQL Server.

En el primer capítulo definiré lo que es una red y algunos otros términos, así como también platicaré un poco acerca de lo que es Visual Basic, SQL

Server, Windows NT, y de la aplicación tratada, con el objeto de introducirnos al tema del trabajo.

La arquitectura Cliente/Servidor (*client/server*) es una red de comunicaciones, en la que el Cliente es la máquina o computadora solicitante y el Servidor es la máquina proveedora. Esto implica que existe un software especializado en ambos extremos. Por ejemplo, en un sistema de base de datos para trabajar en red, la interfaz de usuario reside en la estación de trabajo (Cliente) y las funciones de almacenamiento y recuperación residen en el Servidor.

Una aplicación desarrollada en la arquitectura Cliente/Servidor tiene muchos beneficios que se ven reflejados en la productividad de la empresa (en caso de ser utilizada en un negocio), en comparación de otros tipos de arquitecturas, como por ejemplo: la velocidad de transmisión, la disminución del tráfico de datos, seguridad, etc.

En el segundo capítulo hablaré sobre este tipo de arquitectura, lo que es, sus características y funciones de sus componentes.

El tercer capítulo trata de la combinación de la Arquitectura antes mencionada con Windows NT. Y por último, el cuarto capítulo es la exposición de la propuesta de diseño de una aplicación para un centro de capacitación con arquitectura Cliente/Servidor bajo Windows NT, utilizando Visual Basic y SQL Server.

CAPITULO I

GENERALIDADES

I.- GENERALIDADES

I.1 Conceptos y características

I.1.1 Redes de Computadoras

Una red de computadoras es:

" Un conjunto de computadoras interconectadas entre sí a través de uno o varios caminos o medios de transmisión".¹

Ya que las organizaciones actuales suelen estar dispersas en varios puntos del país, o la información de la organización es necesaria en varios departamentos de la misma, es fundamental contar con una red de computadoras. A parte de que también se pueden compartir recursos, por lo que el uso de éstos se optimiza.

Otro beneficio que nos da una red, es que si una computadora falla, otra puede asumir sus funciones sin que se pierda información.

¹Uyless Black, REDES DE COMPUTADORAS, Protocolos, Normas e Interfaces, cap. 1 pag 1

1.1.2 ERWIN

ERWIN es una herramienta para el diseño lógico y físico de bases de datos. El modelo lógico es el que nos muestra la relación entre entidades en un sistema. El modelo físico comprende el diseño de las tablas que se van a usar en la aplicación, con sus respectivos campos y sus tipos de dato.

Entre las características más importantes de ERWIN están :

- **Facilidad de uso.**
- **Permite hacer el diseño lógico de una aplicación.**
- **Podemos hacer el diseño físico a partir del diseño lógico.**
- **Genera código de los manejadores de bases de datos más importantes desde el diseño físico, por ejemplo los triggers (reglas para conservar la integridad de los datos).**
- **Permite traer, desde un manejador de base de datos, tablas que podemos convertir en diseño lógico y pasarlas a otro manejador de base de datos.**

1.1.3 Windows NT

Desde su creación, las computadoras personales han usado sistemas operativos de 16 bits bajo una variedad de acrónimos que van desde el CP/M, el ZDOS, el TRSDOS hasta el contemporáneo MS-DOS/Windows OS. Se comenzó con los chips 8086 y se continuó con el 80286 y el 80386 SX que sólo soportaban operaciones de 16 bits.

Más recientemente el 80386 DX apareció, el cual ya soportaba operaciones a 32 bits, aunque por compatibilidad aún soportaba la base del sistema operativo de 16 bits.

Por las imposiciones y limitaciones que ofrecía el sistema operativo de 16 bits, el 80386 DX, luego el 80486 y posteriormente el pentium, no podían mostrar sus grandes ventajas sobre sus predecesores. Algunas pruebas fueron hechas tales como operaciones en modo protegido de Windows y controladores de memoria extendidos de DOS que utilizaban el potencial de estos CPU's. De alguna manera hubo mejoría, pero los resultados aún estaban lejos de lo óptimo.

La introducción del OS/2 fue el primer hallazgo importante. Por primera vez un sistema operativo fue introducido y no hacía el intento de proveer compatibilidad con CPU's obsoletos, en vez de eso fue diseñado **explícitamente** para operar solo en CPU's compatibles con 32 bits, esto es el 80386 DX, 80486 o el pentium.

Actualmente existen algunos sistemas operativos que trabajan a 32 bits, Windows NT es uno de ellos.

Windows NT es un sistema operativo real de 32 bits, basado en la versión 3.1 de Windows. Aún cuando los sistemas operativos de 32 bits ofrecen gran cantidad de ventajas sobre el sistema operativo de 16 bits (MS-DOS con o sin Windows 3.1) sólo dos de ellas son realmente relevantes.

La primera ventaja es que la memoria es lineal, al menos hasta 4 Gb. Anteriormente, con un sistema operativo de 16 bits la dirección máxima de

memoria posible era de 64 k aún cuando algunos esquemas de direccionamiento proveían métodos para extenderse más allá del límite de 64 k. Sin embargo, aún con esta virtud, la estructura del DOS imponía su propia limitación, y tuvieron que implementarse algunos otros esquemas para direccionar a memoria alta (más allá de 1 Mb). Pero aún cuando algunos sistemas operativos han proveído medios para direccionar varios Mb de memoria, estas provisiones han sido lentas y poco funcionales.

Desde luego probablemente no va a pasar mucho tiempo antes de que los 4 Gb de memoria se vean restringidos y algunos otros esquemas serán creados para direccionar segmentos de memoria extendida y una vez que las restricciones inherentes en el sistema operativo Windows NT se vean arcaicas y sean criticadas por no proveer soporte para los nuevos y más rápidos CPU's y periféricos. Windows NT tendrá que ser reemplazado por un nuevo Sistema Operativo. Por el momento Windows NT nos provee con memoria que puede direccionar el rango completo de memoria soportado por la mayoría de las máquinas.

La segunda ventaja es que el proceso de datos en pedazos de 32 bits es más rápido. Desde luego que este cambio significa que las instrucciones de 32 bits están disponibles en lugar de las instrucciones de 16 bits, pero más que reemplazar a 64,000 instrucciones posibles con un conjunto de 4 billones de instrucciones, la diferencia real es un cambio en como estas instrucciones son manejadas.

El ambiente de Hardware

Primero y más importante Windows NT no operará en sistemas 8080, 8086 u 80286; requiere una plataforma mínima de un 386 con al menos 16 MB de RAM, de cualquier forma un procesador más rápido y RAM adicional son de mucha ayuda.

Un segundo requerimiento es espacio en disco duro. Windows NT requiere un mínimo de 70 MB de espacio en disco duro, además de 20 MB adicionales para el archivo de intercambio. Desafortunadamente, los discos duros de alta capacidad aunque están más baratos que hace algunos años son aún uno de los componentes más caros en una computadora. Sin embargo estos discos viejos, pueden ser montados como esclavos aún cuando sean fabricados por diferentes compañías.

También aún cuando utilizarías de compresión del disco, tales como el stacker, se volvieron populares en los últimos años, tales utilerías no son compatibles con Windows NT por la simple razón de que requeriría controladores de 16 y 32 bits para operar en ambos sistemas (el Windows NT y el DOS). Aún cuando la versión 6.0 del DOS proveía una utilería de compresión de disco el controlador correspondiente a 32 bits nunca apareció, así el sistema de compresión del MS -DOS fue incompatible con Windows NT.

El ambiente del software

Tal vez la mayor ventaja de Windows NT sobre versiones anteriores de Windows, es el cambio de un sistema operativo de 16 bits a 32 bits. Windows 3.x operaba como un subprograma bajo DOS, Windows NT ofrece su propio sistema operativo de 32 bits para permitir a los usuarios cambiar entre el Windows NT y el DOS. La opción de arranque dinámico o flexible es soportada y puede ser usada para seleccionar el arranque en cualquiera de los sistemas operativos.

La facilidad de uso de los sistemas operativos, como es el caso de Windows NT, con una interfaz gráfica, hacen que los usuarios de redes no especializados puedan resolver algunos problemas e incrementar su productividad.

Existen dos Sistemas Operativos Windows NT: Windows NT Workstation y Windows NT Server. Hablaré acerca de Windows NT Server.

Windows NT Server es un sistema operativo de red que puede funcionar como Servidor de archivos, impresoras, aplicaciones, etc. Este sistema se instala en servidores, en los cuales se puede instalar en forma de *PDC (Primary Domain Controller)*, que es el servidor que maneja las cuentas de los usuarios y por lo tanto administra la red, y en forma de *BDC (Backup Domain Controller)*, que, como su nombre lo dice es una copia o respaldo del servidor que maneja las cuentas, pero únicamente de la Base de Datos y no de las aplicaciones o del Hardware que el *PDC* contenga.

Windows NT tiene una estructura interna que está formada por varias capas:

Capa de abstracción del hardware (HAL-Hardware Abstraction Layer)

Básicamente, tiene como función aislar al Ejecutivo y al kernel (los cuales explicaré más adelante) de las diferencias que puedan existir entre diferentes plataformas de hardware.

Microkernel

El microkernel es el corazón del sistema operativo. Se hace cargo de realizar las funciones básicas necesarias para que este pueda operar, se ocupa de la calendarización de *threads*, de la sincronización de múltiples procesadores y, en general, proporciona al Ejecutivo una serie de servicios internos para poder realizar sus funciones.

Manejadores de Dispositivos

También llamados *drivers*, son aquellos módulos de software que se encargan de "hablar" directamente con los dispositivos de entrada/salida de la computadora, o en general con el hardware de la misma.

Sistema de E/S

La función que realiza el sistema de entrada/salida es la de comunicarse con los controladores de dispositivos, aislando al resto del sistema operativo de las particularidades de cada tipo de dispositivo. Este

Este sistema es en gran medida responsable de que Windows NT sea independiente de los dispositivos de salida. Por ejemplo, el mecanismo de acceso a video, a disco y a impresora será exactamente el mismo. También este sistema se hará cargo del manejo de memoria caché y de la estructuración, manejo y presentación al usuario de la información contenida en los sistemas de archivos.

Ejecutivo

El Ejecutivo es la parte más compleja del sistema operativo. Tiene como función administrar los recursos, los permisos de los usuarios, ofrece procesamiento multitarea, controla el acceso a dispositivos de E/S, creación y manejo de objetos, ofrece a los procesos que corren en modo usuario, los servicios del Ejecutivo que necesitan para poder tener acceso a los recursos del sistema.

Subsistemas protegidos

Se dividen en dos tipos: subsistemas de ambiente y subsistemas integrales, los primeros corren en modo usuario y los segundos en modo privilegiado. El modo usuario es el que tienen las aplicaciones, y se refiere a los permisos que le son otorgados, los cuales tienen muchas restricciones; el modo protegido es el que tiene el sistema operativo, el cual controla toda la red.

Los subsistemas de ambiente son básicamente los ambientes operativos que proporciona Windows NT. Cada uno es responsable de sus propios APIs (*Application Program Interface*), y una misma aplicación no

puede utilizar APIs de diferentes subsistemas de ambiente. Un subsistema de ambiente puede proporcionar servicios sobre cualquier proceso, siempre y cuando se proporcione el identificador del proceso y que el usuario tenga los permisos correspondientes.

La figura 1.1 muestra las capas de Windows NT, mencionadas anteriormente.

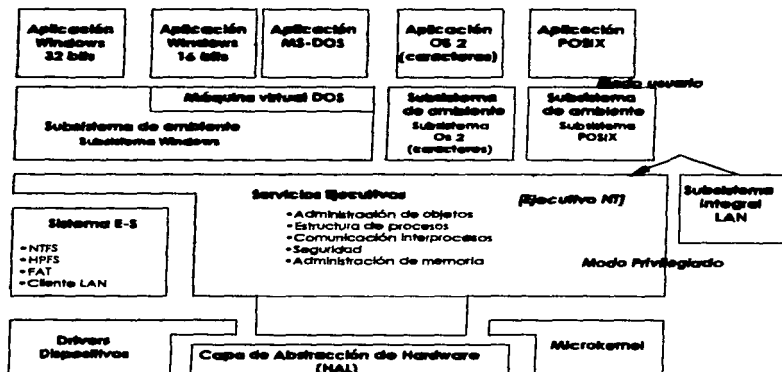


Figura 1.1

Los subsistemas integrales consisten en realizar funciones tales como el manejo de sesiones, el manejo y administración de la seguridad y de las

cuentas de usuarios. Este subsistema proporciona servicios básicos de administrador de red. Las capacidades de administración a través de la red, como por ejemplo el manejo de dominios, servicios de réplica automática, respaldos, etc.

Algunas de las características de Windows NT son:

Ventajas:

- Soporta hasta 4 procesadores de forma simétrica
- Tolerancia a fallas
- Sistema operativo gráfico
- Soporta hasta 256 sesiones de manera remota
- Puede convivir con otros ambientes operativos como Macintosh, NOVELL, UNIX, etc.
- Trabaja con servicios de HTTP, FTP y Gopher
- Es multitarea
- Es menos costoso con respecto a otros sistemas operativos de red como es el caso de UNIX.
- El número de servidores instalados con Windows NT crece día con día.

Desventajas:

- Su servicio de directorios no es tan confiable como el de otros sistemas operativos. Aunque la versión 5, según Microsoft, resolverá este inconveniente.
- Debido a su poco tiempo en el mercado, es menos confiable que UNIX u otros sistemas operativos.

1.1.4 SQL SERVER.

Muchas organizaciones han descubierto que son dependientes del manejo de información para las más fundamentales operaciones del negocio, así como para la toma de decisiones estratégicas. Por ejemplo, procesamiento de pedidos, control de inventarios, contabilidad, etc.

Por lo anterior, las empresas necesitan de Manejadores de Bases de Datos poderosos, que cumplan con todos los requerimientos de la organización.

Microsoft SQL Server es un Sistema Manejador de Base de Datos Relacional (RDBMS- Relational Data Base Management System) inteligente, basado en la Arquitectura Cliente/Servidor, soporta aplicaciones de procesamiento de transacciones en línea y de soporte en la toma de decisiones.

La seguridad y garantía del procesamiento de transacciones en SQL Server es muy buena para aplicaciones de toma de decisiones. SQL Server posee capacidades avanzadas como por ejemplo: Triggers, Procedimientos que se encuentran almacenados, Reglas y Valores por default. Tiene una tecnología de llamado a procedimientos remotos (RPC- Remote Procedure Call), que hacen de SQL Server una herramienta muy importante para el desarrollo de aplicaciones "back-ends" en estaciones de trabajo, minicomputadoras y mainframes.

SQL Server tiene una arquitectura diseñada para el alto rendimiento y para el proceso multitarea, asegura un rendimiento consistente y estable, aún cuando el número de usuarios aumenta.

En 1993, salió una versión especial de SQL Server especialmente diseñada para el sistema operativo de red Windows NT . SQL Server para Windows NT ofrece un rendimiento excepcional, seguridad y escalabilidad, haciéndolo muy adecuado para grandes corporaciones.

Algunas de las características de SQL Server son:

- Administración remota de los recursos del sistema.
- Arquitectura de 32 bits.
- Un completo y robusto manejo de transacciones con capacidad de completar o revertir (*roll-forward and roll-back*) que permiten la recuperación automática de transacciones al reinicio del sistema.
- Capacidad para generación automática de copias de seguridad.
- La ejecución de procedimientos almacenados en forma remota y procesamiento de transacciones en dos fases (*two-phase commit*) permiten la distribución de la información en servidores geográficamente dispersos.
- Un robusto soporte de tolerancia a fallas.
- El reforzamiento centralizado de la integridad de los datos minimiza el mantenimiento, programación y reduce los costos asegurando que los datos sean correctos.
- Interoperabilidad dinámica con la red local.
- Monitoreo gráfico del rendimiento del sistema.

- Interfaces de programación remota como: *DB-Library* y *ODBC(Open Database Conectivity)*

1.1.4 Visual Basic.

Visual Basic es un sistema de desarrollo diseñado especialmente para crear aplicaciones gráficas de una forma fácil y rápida. Es un lenguaje de alto nivel, y está basado en Basic.

Podemos crear aplicaciones en Visual Basic usando objetos ya predefinidos: ventanas (también llamadas formas) y controles, los cuales nos permiten crear una interfaz amigable con el usuario. Cada objeto, formas y controles, están ligados con un código que permanece inactivo hasta que se de un suceso que lo active. Por lo anterior se puede decir que Visual Basic es un lenguaje orientado por eventos.

Las formas y controles los podemos personalizar para hacerlos más acordes a la aplicación que estemos desarrollando. Esto hace el desarrollo de aplicaciones mucho más rápido y fácil porque, como ya mencioné, usa objetos existentes en vez de escribir todo.

Visual Basic dispone de una gran cantidad de controles que se encuentran disponibles en el software, pero también existen controles externos que Visual Basic puede manejar a través de bibliotecas, lo que lo hace muy poderoso.

Muchas personas no consideran a Visual Basic como un lenguaje basado en objetos, ya que, según los expertos, aún no cumple con las características de este tipo de lenguajes, las cuales son: herencia, polimorfismo y encapsulamiento.

La programación orientada a objetos (OOP), se refiere a más que simplemente usar objetos existentes durante el desarrollo.

1.1.5 SACE

SACE es una aplicación para un centro donde se capacita al personal de otras organizaciones. El diseño de esta aplicación se llevó a cabo en ERWIN y está pensado para una arquitectura Cliente/Servidor.

Se trata de una aplicación que tendrá como función registrar a todos los clientes en una base de datos, ya sea a través de una persona que administre la base de datos o por el mismo cliente a través de Internet. Esta base contendrá información de los cursos, los calendarios, horarios, nombres, e instructores que los impartirán, así como también información acerca de los instructores como por ejemplo su experiencia, sus certificaciones, etc.

La aplicación se desarrollará en Visual Basic y como manejador de la base de datos SQL Server. El sistema operativo propuesto es Windows NT.

La base podrá ser consultada por el administrador de la aplicación, el cual informará a las personas que quieran saber algo sobre los cursos, ya sea personalmente o por teléfono. Otra forma de consulta será por Internet,

donde el cliente podrá consultar todo lo referente a los cursos y en donde se podrá registrar sin necesidad de que el administrador de la aplicación (en este caso la secretaria) lo registre.

CAPITULO II

LA ARQUITECTURA CLIENTE/SERVIDOR

II.- LA ARQUITECTURA CLIENTE/SERVIDOR

Antes de que la Arquitectura Cliente/Servidor surgiera, se tenían mainframes con varias terminales tontas (TTY), las cuales tenían únicamente la función de interfaz entre el usuario y la mainframe, sin llevar a cabo ningún tipo de proceso; todos los procesos los realizaba la mainframe.

Después empezaron a tener auge las computadoras personales (PC's), las cuales, a diferencia de las TTY, tenían capacidad de procesar información, capacidad de almacenamiento, etc., lo cual las hacía máquinas inteligentes, y mejor aún, empezaron a ser accesibles en cuanto a costo; mientras que las TTY empezaban a resultar costosas en cuanto a los beneficios que con ella se obtenían, a su mantenimiento, a su eficiencia y eficacia, etc.

Por lo tanto las PC's comenzaron a reemplazar a las TTY. El mejoramiento de la tecnología propició que máquinas "pequeñas" compitieran en productividad con las mainframe. Por lo que muchos mainframes han sido reemplazados por computadoras más chicas, que tienen características similares, aunque no tienen la gran capacidad de almacenamiento que tiene una mainframe.

Al darse cuenta de que las nuevas terminales de trabajo tenían la capacidad de procesar información, decidieron distribuir la carga entre el servidor y las terminales de trabajo. A este tipo de sistema se le llamó Cliente/Servidor.

A pesar que la tendencia en los últimos cinco años ha sido reemplazar aplicaciones que se tenían en mainframe por Cliente/Servidor, no en todos los casos ha resultado, por ejemplo hay algunas aplicaciones que corren en mainframe que manejan grandes cantidades de datos que en otros tipos de arquitectura sería difícil igualar, además, las aplicaciones mainframe están evolucionando en sistemas más nuevos, más interactivos y más dinámicos. Aunque hay ciertas aplicaciones más pequeñas que funcionarían mejor en una Arquitectura Cliente/Servidor.

II.1 Definición

Un sistema Cliente/Servidor se compone de dos diferentes sistemas: uno actúa como usuario y el otro aporta y administra la información. Sin embargo, estos dos sistemas son englobados en uno solo.

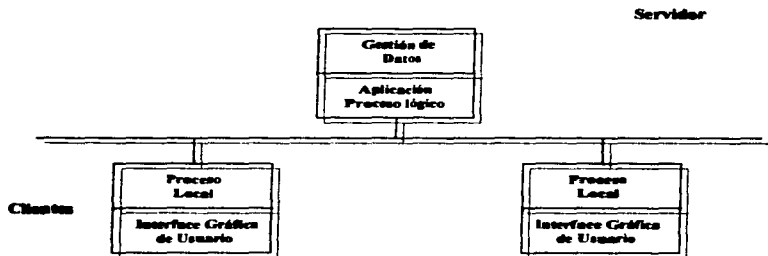


Figura II.1

Una forma de configurar la red en un sistema Cliente /Servidor, y la que generalmente se usa, es que existe un servidor o ente que administra la red, y estaciones de trabajo o entes que realizan la petición. Puede haber más servidores: de archivos, de páginas WEB, etc. Aunque también puede ser que en un solo servidor estén contenidos todos estos servicios, pero esto no es muy recomendable, solo en casos de que la red no sea muy grande o no se maneje gran cantidad de información. También pueden estar contenidos en una misma máquina el Cliente y el Servidor.

Arquitectura Cliente/Servidor

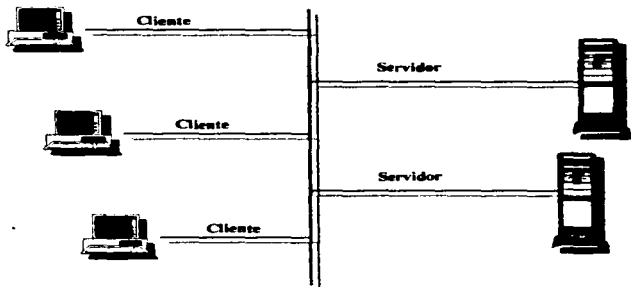


Figura 11.2

El sistema Cliente/Servidor es distribuido, es decir, los procesos son realizados tanto en el servidor (o servidores), como en las estaciones de trabajo.

Comúnmente un servidor provee de servicios a varios sistemas Cliente, aunque puede haber un solo Cliente.

Una simple base de datos no constituye un sistema Cliente/Servidor, aunque estos sistemas (Cliente/Servidor) frecuentemente usan una base de datos para ejecutar la actividad del servidor. Las aplicaciones diseñadas para utilizar Access, Microsoft FoxPro, Paradox by Borland u otras bases de datos de escritorio no son sistemas Cliente/Servidor, aún cuando esas bases

de datos residan en un servidor de red. Estos son ejemplos de aplicaciones de base de datos en red porque todo el proceso es efectuado por el Cliente.

11.2 ¿Cómo funciona un sistema Cliente/Servidor?

Los sistemas Cliente/Servidor fueron originalmente desarrollados para proveer un mejor desempeño, al trasladar algo del proceso del Cliente al servidor. Esto debería mejorar el desempeño, pero hacía poco por el costo. La arquitectura Cliente/Servidor permite alcanzar una proporción significativa desempeño sobre precio: por ejemplo, en vez de comprar once súper PCs, colocando un disco duro muy grande en una de ellas usándola como servidor y correr una aplicación de base de datos en red en los 10 Clientes restantes, podemos comprar 10 PCs medias y un muy buen servidor.

La figura 11.3 muestra como está configurado un Sistema Cliente/Servidor típico. En este diagrama, el cableado y el foco (centro, HUB) se muestran para darle claridad. Las redes varían en el nivel de hardware, así que sus componentes consistirán en mas que un simple hub y algunos cables.

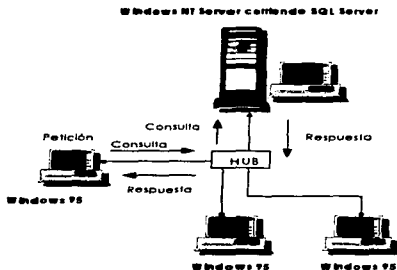


Figura 11.3

En la figura 11.3 el Cliente Microsoft Windows 95 genera una consulta SQL. La consulta se manda a través de la red al Servidor Microsoft Windows NT corriendo SQL Server. El SQL Server procesa entonces la consulta y regresa los datos al Cliente Windows 95.

La operación del sistema Cliente/Servidor mostrada es diferente a una aplicación de bases de datos de escritorio en el modo en que son procesados los datos. Usando una base de datos de escritorio como Access, la tabla entera de la base de datos requerida en la consulta sería enviada al Cliente para proceso. El Cliente procesaría los datos y entonces mostraría el resultado en pantalla.

Una base de datos de PC típica es bastante menos eficiente al correr en una red debido a la cantidad de datos enviados por la red y porque la carga de la aplicación no se divide entre el Servidor y la PC de escritorio. El único trabajo que el Servidor hace en este caso es manejar las tareas habituales de Servidor de archivos al proporcionar acceso a la base de datos desde la PC.

La características de automatización OLE (*Object Linking and Embedding*) y Automatización Remota de Visual Basic 4 y 5 cambian el modo en que un sistema Cliente/Servidor puede ser construido y utilizado. El correr aplicaciones de Visual Basic 4 o 5 en Windows 95 o Windows NT provee una poderosa combinación de características para sus clientes, completa con recursos expandidos, multitareas, un sistema de archivo recuperable (*Windows NT File System o NTFS*), y más.

La Automatización Remota puede ser usada para mantener a la vista el inventario a través de un Sistema de Alarma Ejecutivo. Por ejemplo: El cliente A manda una petición al servidor para ver la cantidad del inventario. Esto activa un chequeo periódico del valor total del inventario mediante un programa en el servidor, sin requerir intervención posterior del cliente. El servidor envía una alerta cuando el valor del inventario excede \$2 millones y también dispara (activa) un programa de Automatización Remota diferente en el cliente, el cual despliega el estado del inventario y pide acción de un administrador.

La petición del cliente B utiliza una aproximación distinta. El cliente B pide un listado de todo el inventario desde el servidor. Si esta consulta fuera corrida interactivamente, alentaría al servidor, al cliente y a la red al recuperar, esperar por y transportar los datos. En vez de esto, el cliente B emite la petición llamando un procedimiento archivado en el SQL Server. Este procedimiento archivado utiliza entonces *MAPI* (*Messaging Application Program Interface*) para enviar la respuesta a la cuenta de Microsoft Mail del Cliente.

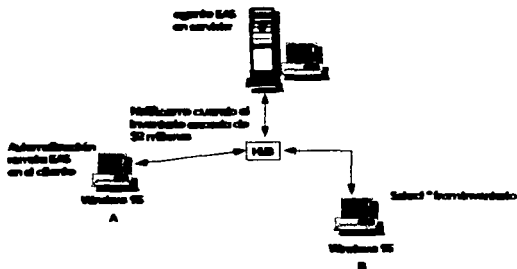


Figura II.4

Uno de los mejores modos de construir un sistema Cliente/Servidor que sea compatible con Conectividad de Base de Datos Abierta (*Open Database Connectivity ODBC*) es usar una base de datos local de Access, con el servidor conteniendo tablas adheridas. Una de las razones por las que esto trabaja tan bien es que la base de datos de Access contendrá las estructuras internas (campos e índices) para las tablas adheridas a la base de datos ODBC, reduciendo la cantidad de trabajo que debe ser hecho cada vez que es procesada una petición a la base de datos.

Access entiende las capacidades ODBC de la base de datos remota y puede pasar todas las peticiones al servidor para ejecución excepto cuando una consulta usa funciones o características soportadas por Access pero no por la base de datos final.

II.3 Funciones de los componentes

II.3.1 Función del Cliente

La función del cliente es la de interactuar con el usuario final. Es lo que se llama una aplicación *front - end*. A través del cliente el usuario puede acceder a la información, aplicaciones, etc., que se encuentran en el servidor o servidores. Este acceso lo puede hacer mediante diversas Interfaces, como por ejemplo Visual Basic y SQL Server.

II.3.2 Función del Servidor

La función del servidor es la de proveer aplicaciones, información, etc., al cliente para que éste pueda operar con el usuario.

II.3.4 Función de la Red

La red tiene diversas funciones: conectar a los clientes y servidores, controlar el flujo de datos que viajan a través de ella, controlar que la información viaje en forma correcta, etc.

II.4 Modelos de la Arquitectura

II.4.1 Presentación Distribuida

Este tipo de modelo se refiere a la forma de presentación de la información a los usuarios, y trata de que ambos, cliente y servidor, den un formato en grupo, es decir, el servidor envía los datos semiformateados al cliente, el cual se encarga de darle un formato final.

II.4.2 Presentación Remota

En este tipo de modelo, el servidor manda alguna parte lógica al cliente, entonces, la mayoría del manejo lógico de la aplicación reside en el cliente, pero el servidor sigue manejando la integridad de los datos.

II.4.3 Distribución Lógica

En este modelo, los sistemas Cliente/Servidor dividen la lógica de la aplicación entre el cliente y el servidor. El cliente controla el flujo de la aplicación mediante el uso de aplicaciones GUI (Graphical User Interface), mientras que la base de datos y la integridad de la información siguen siendo controladas por el servidor.

11.4.4 Datos Remotos

Aquí, el cliente se encarga de manejar toda la aplicación lógica y su presentación al usuario, en tanto que el servidor solamente proporciona los datos.

11.4.5 Datos Distribuidos

Este modelo se caracteriza porque los datos están contenidos en varios servidores de bases de datos y el manejo de los datos se realiza a través de un cliente. Aquí se puede utilizar la técnica de la replicación, aunque puede haber problemas con la integridad, consistencia y seguridad de los datos.

CAPITULO III

LA ARQUITECTURA CLIENTE/SERVIDOR JUNTO CON WINDOWS NT

III.- LA ARQUITECTURA CLIENTE/SERVIDOR JUNTO CON WINDOWS NT

III.1 Características de la combinación Cliente/Servidor - Windows NT

Windows NT está basado en una serie de modelos básicos. Uno de ellos es la arquitectura Cliente/Servidor, ya explicada en el capítulo anterior.

Éste, normalmente se asocia con redes de comunicaciones en las cuales existen computadoras con nodos clientes y nodos servidores. Un mismo nodo puede en ocasiones realizar ambas funciones al mismo tiempo.

El principio es idéntico cuando se habla de un sistema operativo. En este caso existen procesos clientes y procesos servidores.

En la siguiente figura III.11a aplicación es el proceso cliente, y el subsistema de ambiente el proceso servidor. Ambos se comunican entre sí utilizando RPC (Remote Procedure Calls; llamadas a procedimientos remotos), que constituye una serie de servicios de comunicación que proporciona el ejecutivo NT, que es el núcleo del sistema.

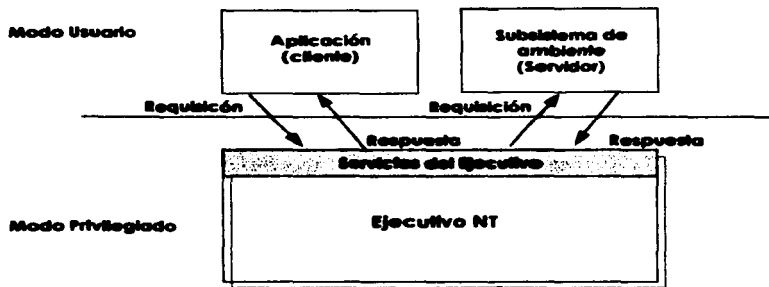


Figura III.1

El proceso cliente envía una petición al proceso servidor, a través del Ejecutivo. El servidor la recibe, la atiende, y envía la respuesta. Todo en forma completamente transaccional. Un subsistema de ambiente para establecer un paralelo es como una "vista" de base de datos. Es decir, es el proceso que construye y presenta una interface al usuario a través de la cual, éste obtiene acceso a los servicios del sistema.

Podría decirse que cada subsistema de ambiente presenta una visión o una realidad virtual del sistema. Dicha visión cambia un poco dependiendo de la forma de trabajar de cada sistema operativo que la originó, y se compagina con la realidad del Ejecutivo NT a través de los servicios que el mismo proporciona a cada subsistema.

Cabe mencionar que una de las consecuencias del uso de este tipo de arquitectura, en el caso de Windows NT, ha sido que el sistema operativo está estructurado por capas en forma completamente modular. Como en el caso del modelo OSI en comunicaciones, cada módulo es completamente independiente de los demás. Por lo tanto, sólo es necesario definir muy cuidadosamente las interfaces de comunicación entre los módulos, y será entonces posible en un momento dado "sacar" un módulo y reemplazarlo por otro nuevo sin afectar al resto del sistema operativo.

La capa del Ejecutivo corre en modo privilegiado, y tanto la aplicación como el subsistema de ambiente corren en modo usuario. Correr en modo privilegiado significa que las aplicaciones no tienen acceso directo a los procesos del ejecutivo mas que a través de una serie de servicios que se denominan Servicios del Ejecutivo, y que son el único mecanismo que poseen las aplicaciones que corren en Windows NT para obtener acceso a los recursos del sistema.

Las implicaciones del uso de este modelo son muchas. Cada subsistema de ambiente corre en un ambiente completamente aislado de los demás, en modo usuario y completamente aislado de sus aplicaciones, lo que significa que una aplicación que falla no nada más no puede detener el sistema, sino que ni siquiera puede detener al subsistema de ambiente. Y si es este último el que falla, no podrá a su vez detener al sistema.

Esto quiere decir que si una aplicación falla, lo único que se afectará serán sus propias instancias de ejecución. Y lo único que fallará a los ojos del usuario será la ventana o ventanas que tenga abiertas dicha aplicación. El

resto de las aplicaciones y subsistemas de ambiente seguirán corriendo de forma normal.

Otro resultado que se deriva del uso de este modelo es que cada subsistema de ambiente corre en realidad en modo nativo, no es una emulación, lo cual garantiza un buen rendimiento al correr aplicaciones.

Un sistema operativo de este tipo, se presta fácilmente a la integración de sistemas de cómputo distribuido, ya que estos operan esencialmente siguiendo un modelo cliente/servidor, con lo cual la red de comunicaciones se convierte en una extensión natural del sistema operativo, en este caso de Windows NT.

III.2 ¿ Por qué Usar Visual Basic y SQL Server para un sistema Cliente/Servidor?

Visual Basic es una herramienta muy poderosa. Las razones son muchas, pero todas ellas surgen de la integración de Visual Basic con otras aplicaciones basadas en Microsoft Windows, es fácil de usar, y el gran poder que provee.

En Visual Basic podemos construir muy buenas aplicaciones. La Automatización Remota, combinada con las características de automatización del estándar OLE, hacen de Visual Basic un desarrollador de ambientes poderoso. Remote Data Objects (RDO) y el Remote Data de control también hacen a Visual Basic una poderosa herramienta para el desarrollo en arquitectura Cliente/Servidor.

Se puede tener control sobre el código fuente al evitar perder líneas de código al trabajar en red con otros programadores con el SourceSafe, este software nos "aparta" formas, las cuales se convierten de escritura para quien la "apartó" y de solo lectura para los demás programadores de la red.

Visual Basic provee módulos de clase que tienen la capacidad de construir rápida y fácilmente bibliotecas reusables.

Las bases de datos de los sistemas Cliente/Servidor típicamente proveen aspectos poderosos no encontrados en bases de datos tales como Access, Foxpro, Paradox, o dBase.

CAPITULO IV

DISEÑO DE UNA APLICACIÓN PARA UN CENTRO DE CAPACITACIÓN UTILIZANDO ARQUITECTURA CLIENTE/SERVIDOR.

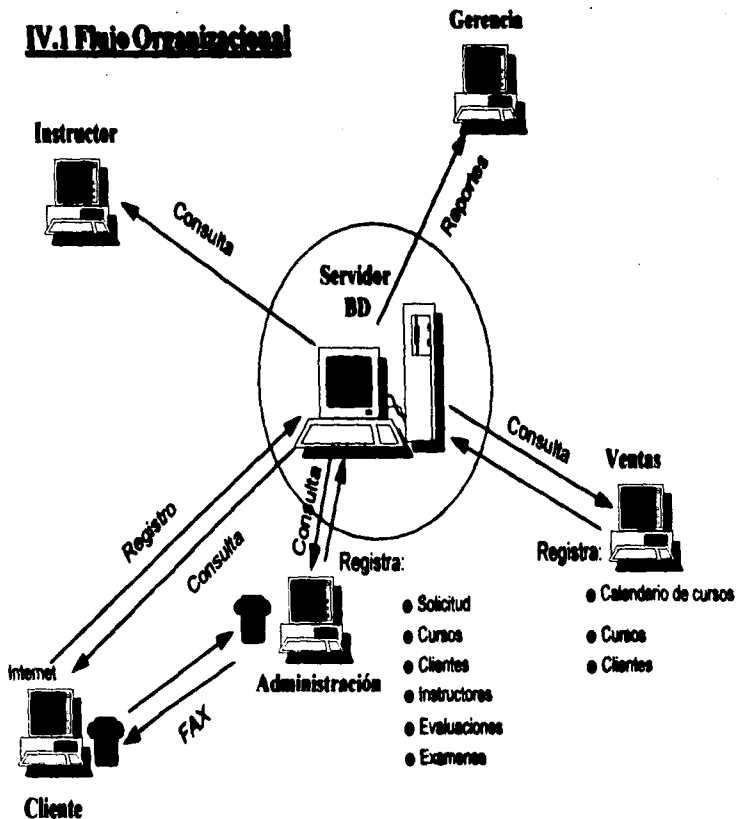
IV.- DISEÑO DE UNA APLICACIÓN CON ARQUITECTURA CLIENTE/SERVIDOR BAJO WINDOWS NT, UTILIZANDO VISUAL BASIC Y SQL SERVER

Requerimientos del sistema:

- El sistema debe permitir registrar clientes, cursos e instructores en una base de datos contenida en un servidor, en el cual se podrá acceder a través de Internet o a través de la red del centro educacional.
- Cuando un instructor quiera buscar información acerca de algún curso, podrá consultar esta base de datos.
- Cuando la persona encargada de dar información (en este caso la secretaria) desee o necesite consultar la base lo podrá hacer en cualquier momento. También es la encargada de alimentar al sistema.
- También podrán consultar dicha base las personas que se encargan de las ventas de los cursos, para así poder dar la información correcta y oportuna. Ellas también podrán registrar cursos, calendarios de cursos y clientes.
- Cualquier persona puede llamar y pedir información a la secretaria en forma oral o por medio de un FAX, o bien, podrá consultar en Internet la página que contiene la información de la base de datos.
- Las personas que se quieran registrar lo podrán hacer mediante una llamada telefónica a la secretaria o por medio de Internet.
- El sistema debe permitir realizar reportes para la gerencia, para la toma de decisiones.

Estos requerimientos son esquematizados en la siguiente página.

IV.1 Flujio Organizacional



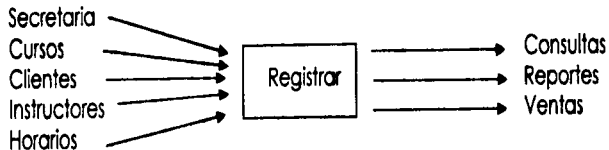
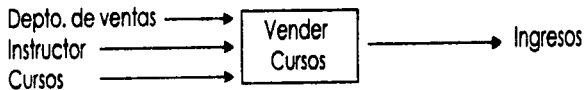
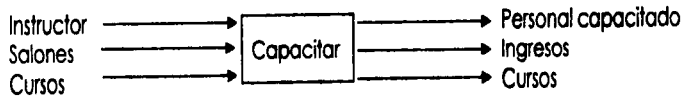
IV.2 El Modelo de Procesos

Este modelo (página siguiente) nos presenta las principales actividades que se desarrollan en el Centro de Capacitación, así como las entradas y salidas para estas actividades.

Las entradas se refieren a lo que se necesita para poder realizar la actividad, en el caso de CAPACITAR, para que esta se pueda realizar se necesita de un instructor, que existan salones y que haya cursos que dar. Obtenemos por salidas después de pasar por el proceso de capacitar, personal capacitado, ingresos, y más cursos.

Este tipo de modelo nos ayuda a encontrar todas las entidades que vamos a necesitar. El modelo que presento es de las actividades más importantes que se llevan a cabo, pero este diagrama se puede hacer tan complejo como uno quiera, y se puede introducir con las actividades menos importantes de la organización.

IV.2 MODELO DE PROCESOS



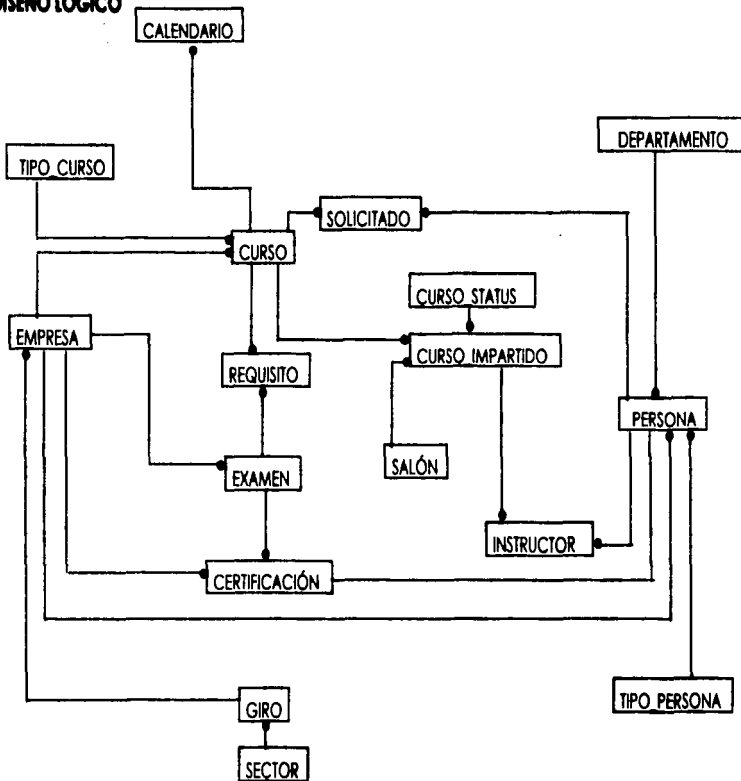
IV.3 Diseño Lógico

El diseño lógico nos muestra el modelo de Entidad-Relación (diagrama de la página 44). Nos ayuda a identificar cada una de las entidades que participan en la aplicación, así como conocer con que otras entidades se relaciona y que tipo de relación es.

Por ejemplo, en las entidades que presento CURSO se relaciona con CALENDARIO, ya que todos los cursos deben estar programados para cierto día y hora. También CURSO se relaciona con REQUISITO porque todos los cursos deben cumplir ciertos requisitos.

Los círculos negros se refieren a la relación uno a muchos, del lado donde se encuentra es la entidad muchos. Por ejemplo, un departamento puede tener muchas personas.

IV.3 DISEÑO LÓGICO



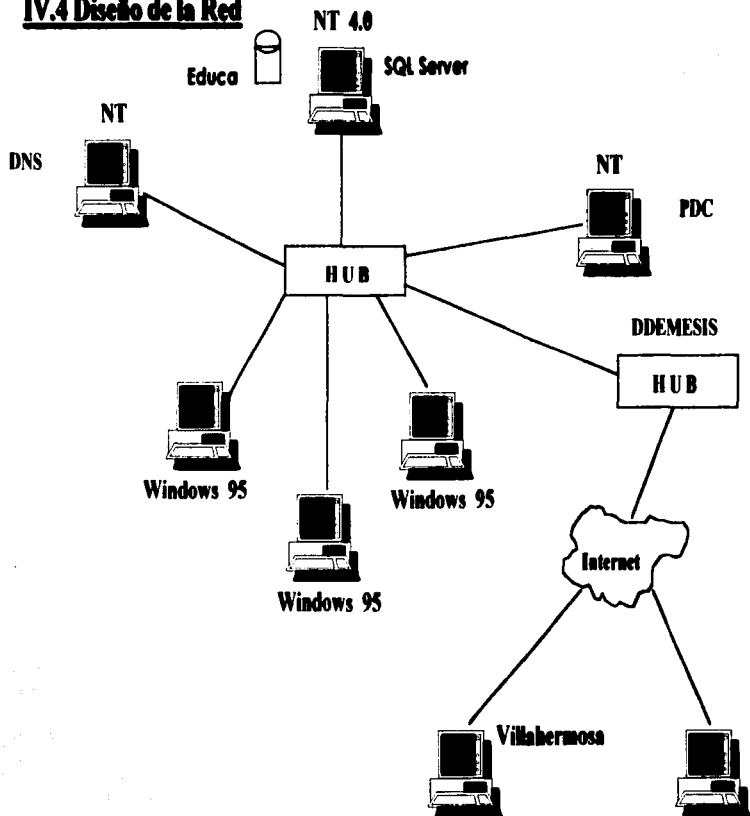
IV.4 Diseño de la Red

En el centro de capacitación se tienen varias máquinas, de las cuales se pueden tomar tres máquinas como servidores. En el primer servidor se almacena DNS que son los dominios de las máquinas. El segundo contiene SQL Server, en donde será almacenada la base de datos Educa. Y por último el tercero es el PDC, que contiene las cuentas de los usuarios.

En las estaciones de trabajo puede haber Windows 95, ya que es más familiar al usuario. Todos están conectados a través de un HUB. Se podrán realizar consultas desde una sucursal de la organización en Villahermosa o desde cualquier usuario a través de Internet. Estas conexiones se harán al HUB de la organización y posteriormente al HUB del centro de capacitación, el cual es una área de la organización.

Lo anterior está representado en la página siguiente.

IV.4 Diseño de la Red



IV.5 Diseño Físico

El modelo o diseño físico, como mencioné anteriormente, es la presentación de las tablas que van a contener los campos, los cuales van a tener la información necesaria para el buen funcionamiento del sistema.

La tabla CALENDARIO va a contener la información de la fecha en que se van a iniciar los cursos, así como la clave del curso para poderlo identificar en la tabla CURSO y así poder "jalar" la información de esta última tabla.

La tabla TIPO_CURSO tiene dos campos los cuales van a tener la clave del tipo de curso, para poder identificar a esta tabla y el nombre del tipo de curso, que pueden ser Sistemas Operativos, Paquetería y de desarrollo de aplicaciones, principalmente.

CURSO_STATUS es una tabla que va a contener información acerca del estado en que se encuentran los cursos, es decir, si ya concluyó, si aún no empieza, si está impartándose, si se canceló, etc.

CALENDARIO

Curso_Id (FK)
Fecha_Inicio

TIPO_CURSO

Tipo_Curso_Id
Tipo_Curso_Nombre

CURSO_STATUS

Status_Id
Status_Dsc

EMPRESA contiene toda la información referente a la empresa a que pertenecen las personas que toman los cursos y también los instructores, que en algunos casos son externos.

La tabla SOLICITADO contiene información de los cursos ya concretados, es decir, de los cursos que ya están confirmados y listos para darse. En esta tabla se registra la fecha de inicio, la fecha final, el costo, etc.

CURSO, Aquí está el catálogo de cursos posibles a impartir por el centro de capacitación; el nombre, la versión, el costo, la duración, etc.

EMPRESA

Empresa_Id
Giro_Id (FK)
Empresa RFC
Empresa_Nombre
Empresa_Razón
Empresa_Domicilio
Empresa_Teléfono
Empresa_Fax
Empresa_Email

SOLICITADO

Solicitado_Id
Persona_Id (FK)
Curso_Id (FK)
Ponencia_costo
Ponencia_Fecha_Ini
Ponencia_Fecha_Fin
Ponencia_Hora_Ini
Ponencia_Hora_Fin
Ponencia_Cupo

CURSO

Curso_Id
Tipo_Curso_Id (FK)
Empresa_Id (FK)
Curso_Nombre
Curso_Version
Curso_Costo
Curso_Tamaño
Curso_Duracion

En CURSO_IMPARTIDO vamos a tener información acerca de los cursos que ya se impartieron y de los que se están impartiendo.

Con PERSONA podemos saber los datos de la persona que está tomando el curso y de la persona que lo está impartiendo.

La tabla de CERTIFICACION va a contener los nombres de las personas que han hecho exámenes de certificación, cuantos, a que empresa pertenecen, sus resultados, etc.

CURSO_IMPARTIDO

Impartido_Id
Status_Id (FK)
Persona_Id (FK)
Salon_Id (FK)
Solicitado_Fecha_Ini
Solicitado_Fecha_Fin
Impartido_Hora_Ini
Impartido_Hora_Fin
Impartido_Cupo
Impartido_Comentarios

PERSONA

Persona_Id
Departamento_Id (FK)
TipoPer_Id (FK)
Empresa_Id (FK)
Partic_RFC
Persona_Nombre
Persona_Apell_Pat
Persona_Apell_Mat
Persona_Domicilio
Persona_Telefono
Persona_Fax
Persona_Email
Persona_Sexo

CERTIFICACION

Certificacion_Id
Persona_Id (FK)
Examen_Clave(FK)
Empresa_Id (FK)
examen_resultado
examen_fecha
Certificacion_Status

En REQUISITO conoceremos, como su nombre lo dice, los requisitos o lo necesario para poder realizar un examen de certificación, así como los requisitos para poder introducir un curso en el catálogo de cursos.

DEPARTAMENTO nos va ayudar para localizar a una persona más fácilmente o para poder conocer en que área trabaja para poder formar grupos con el mismo nivel, es decir, si la persona necesita conocer lo básico de algún tema o si ya se es experto.

SALON contendrá los nombres de los salones para poder organizar los cursos y que estos no se encimen en un mismo salón, también para identificar en que salón se encuentra un curso específico.

REQUISITO

Examen_Clave (FK)
Curso_Id (FK)

DEPARTAMENTO

Departamento_Id
Departamento_Nombre

SALON

Salon_Id
Salon_Nombre

La tabla EXAMEN tendrá la clave del examen, el nombre y de que organización proviene, por ejemplo si es de Microsoft.

Para saber a que se dedica la empresa a la que pertenecen los clientes o los instructores (en el caso de que sean externos), podemos consultar GIRO.

SECTOR nos presentará la información acerca del nombre del sector en que trabaja la empresa.

EXAMEN

Examen_Clave
Empresa_Id (FK)
Examen_Nombre

GIRO

Giro_Id
Sector_Id (FK)
Giro_Nombre

SECTOR

Sector_Id
Sector_Nombre

En INSTRUCTOR conoceremos los cursos impartidos por un instructor, los comentarios sobre dicho instructor, así como lo que costó contratarlo, si proviene de otra compañía.

Por último, TIPO_PERSONA nos va ayudar a identificar si la persona es instructor o es cliente.

INSTRUCTOR

Impartido_Id (FK)
Persona_Id (FK)
Instructor_Comentarios
Instructor_Costo

TIPO_PERSONA

TipoPer_Id
TipoPer_Descrip

CONCLUSIONES

La arquitectura Cliente/Servidor es muy eficiente, ya que disminuye el tráfico de la red, también nos permite trabajar con recursos que están contenidos en otras máquinas, por lo que obtenemos ahorro, tanto de espacio en disco como en costo. Una ventaja más que tiene esta arquitectura es que se reparten las tareas entre las estaciones de trabajo y el o los servidores, lo que hace que los procesos sean más rápidos.

Al estar contenida la base de datos en varios servidores ayuda a que un sólo servidor no se sature al intentar acceder muchas máquinas, a la base de datos. Esto funciona muy bien cuando la información que se maneja es muy grande o cuando la organización así lo requiere por el constante uso de información. En el caso que presento la base de datos está contenida en un solo servidor porque la información manejada no es mucha.

Windows NT se presta fácilmente a la integración de sistemas de cómputo distribuido, como es el caso de la arquitectura Cliente/Servidor, por la amplia gama de características que posee.

SQL Server es un manejador de base de datos muy poderoso y muy robusto, por lo que es muy confiable aunque sea poco amigable, pero esto se compensa al usar una interfaz gráfica como Visual Basic. Visual Basic es un lenguaje de desarrollo muy eficaz, ya que podemos realizar aplicaciones muy complejas de manera fácil y rápida, que en otros lenguajes sería difícil programar; esto en cuanto al programador. En lo que se refiere al usuario

también le resulta más fácil entender como funciona una aplicación, porque es una interfaz de windows común, lo que resulta muy amigable.

Con respecto al objetivo planteado, diseñar una aplicación antes de entrar a la fase de desarrollo resulta muy conveniente, ya que al realizar un análisis y luego plasmarlo en un diseño, nos facilita enormemente la programación, porque tenemos un panorama general de lo que realmente se quiere y se necesita.

El diseño realizado sirve para organismos que se dediquen principalmente a impartir cursos y que deseen llevar un control de sus clientes, instructores, exámenes, etc. En este caso fué diseñado para un centro de capacitación específico, lo cual no quiere decir que no pueda ser tomado como base para otros diseños.

Un diseño nos permite conocer más rápidamente las entidades con las que vamos a trabajar, así como sus relaciones. Por lo anterior, cuando realizamos los modelos del diseño, podemos desarrollar más fácilmente tablas y aplicaciones completas.

Finalmente, es importante realizar un prototipo antes de empezar con la programación, con el fin de presentarlo al usuario y que éste de su aprobación en base al cumplimiento de sus necesidades o que nos haga las observaciones pertinentes encaminadas a la mejora del sistema.

BIBLIOGRAFÍA

- 1.- Diccionario de Computación,
5ª edición
Edit. McGraw Hill.
México, D. F. 1995, 934 p.
- 2.- Black, Uyles. Redes de Computadoras.
Edit. Macrobite ra-ma.
México, D. F. 1993, 414 p.
- 3.- Ceballos, Fco. Javier. Visual Basic: Aplicaciones para Windows.
Edit. ADDISON - WESLEY IBEROAMERICANA, ra-ma.
México, D. F. 1995, 420 p.
- 4.- Solomon, David/Woodbeck, Daniel. Microsoft SQL Server 6.
Edit. SAMS.
USA, 1996, 950 p.
- 5.- Echenique, José Antonio. Auditoría en Informática.
México D. F. 1995, 204 p.
Edit. McGraw Hill.
- 6.- Stalling, William. Computer-communications Standards, Local Area Network.
2ª edición
Edit. Howard W. SAMS & Company.
USA, 1995, 279 p.

- 7.- Senn, James A. Análisis y Diseño de Sistemas de Información.
2º edición
Edit. McGraw Hill.
México, D. F. 1994, 942 p.
- 8.- Michael Galos. Supporting Microsoft Windows NT 4.0, Core Technologies.
Edit. Microsoft
USA 1996, 520 p.
- 9.- Appleman, Daniel. Barlow, Chris. Visual Basic. The Professional resource for expert techniques and advice.
Edit. Que
USA 1996, 637 p.
- 10.- Spencer, Kenneth / Miller, Ken. Client/Server Programming with Microsoft Visual Basic.
Edit. Microsoft Press
USA, 1996. 347 p.
- 11.- Salido, Fco. Javier. "Windows NT". RED. N° 55, abril, 1995, p. 22-33.
- 12.- Amézaga, Avot. "El aumento de tráfico en la red". RED. N° 76, enero, 1997, p. 32-34.