55

2EJ

UNIVERSIDAD NACIONAL AUTONOMA DE MEXICO

FACULTAD DE INGENIERIA

SISTEMA DE CORREO ELECTRONICO PARA UNA RED DE AREA LOCAL (LAN)

T E S I S

Que para obtener el Título de:

INGENIERO EN COMPUTACION

p r e s e n t a

LUIS MARTIN HEREDIA BARRAGAN

Director de Tesis: M C. Francisco Nava Morales

México, D. F. *1995*

FALLA DE ORIGEN

TESIS CON
FALLA DE ORIGEN

Universidad Nacional Autónoma de México
UNAM
Biblioteca Central
Dirección General de Bibliotecas de la UNAM

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.

SISTEMA DE CORREO ELECTRÓNICO PARA UNA RED DE AREA LOCAL (LAN)

Contenido

Agradecimientos.....vi

Objetivo.....viii

Motivación.....ix

Capítulo 1. Redes de Computadoras.....1

1.1 Introducción.....2
1.2 Historia de las redes de computadoras.....2
1.3 Componentes de las redes de comunicación de computadoras.....5
1.4 Propósitos de una red de computadoras.....7
1.5 Clasificación de las redes de computadoras.....8
1.5.1 Criterio de clasificación basado en la transferencia de mensajes.....8
1.5.2 Criterio de clasificación basado en la distancia entre estaciones (geográfica).....14
1.5.3 Propagación y retardos en la transmisión de mensajes.....18
1.6 Arquitectura de redes de computadoras.....19
1.7 Modelo ISO/OSI (Introducción).....19
1.7.1 Modelo de referencia de ISO.....21
1.7.2 Capas del modelo OSI.....27
1.8 Estándares en redes de computadoras (generalidades).29
1.9 Aplicaciones de redes de computadoras.....32
1.10 Ejemplos de redes públicas y privadas.....32

Capítulo 2. Correo Electrónico.....33

2.1 Introducción.....34
2.2 Generalidades del correo electrónico.....35
2.3 Sistema individual de correo electrónico.....37
2.4 Correo electrónico de red.....39
2.5 Servicios del correo electrónico (voz).....40
2.6 Resistencia al cambio.....41
2.7 Limitación del medio.....41
2.8 Aplicaciones.....42

Capítulo 3. Redes de Area Local y Token Ring.....43

3.1 Introducción.....44
3.2 Especificación.....44
3.3 Tipos de LANs.....46
3.4 Componentes físicos de una red de área local.....50
3.4.1 Capa física de una LAN.....51
3.4.2 Sistema de cableado.....53
3.4.3 Unidades de acceso y repetidores.....55
3.5 El token.....56
3.5.1 El frame de datos.....59
3.6 Protocolos de alto nivel.....62
3.7 Datagramas y Sesiones.....62

Capítulo 4. Plataforma de Programación Presentada por el NetBios.....64

4.1 Software de soporte a la red.....65
4.1.1 IBM PC LAN Support Program.....66
4.2 NetBios.....68
4.2.1 Nombres en NetBios.....69
4.2.2 Datagramas en NetBios.....70

4.2.3 Sesiones en NetBios.....70
4.2.4 Bloques de Control de Red (NCBs).....71
4.2.5 Frames en NetBios.....73
4.3 Programación con NetBios.....75
4.3.1 Invocación al NetBios.....76
4.3.2 Estructura del NCB.....77
4.3.3 Comandos del NetBios.....78

Capítulo 5. Implementación del Sistema de Correo Electrónico.....79

5.1. Modelado del sistema de correo electrónico (introducción).....80
5.1.1 Modelo conceptual.....80
5.1.2 Estructura interna del sistema.....84
5.1.3 Descripción conceptual de subsistemas, estructura funcional.....92
5.1.4 Entorno de trabajo para la implementación.....97
5.2. Transmisión de voz.....98

Capítulo 6. Resultados y Conclusiones.....100

6.1 Evaluación del desempeño y utilidad del sistema....101
6.2 Mejoras futuras al sistema (recomendaciones).....102

- **Bibliografía**.....Bb1

- **Apéndice A. Funciones del NetBios usadas en este Sistema**.....A1

- **Apéndice B. Bloque de Control de Red (NCB) y Códigos de Error del NetBios**.....B1

- **Apéndice C. IBM Token-Ring Network 16/4 Adapter - 25F7367**.....C1

- **Apéndice D. Guía del Usuario**.....D1

- **Glosario**.....G1

Agradecimientos

A mis padres:

Sra. María del Carmen Barragán Espinosa
Sr. Salvador Heredia Moreno

Por su invaluable apoyo en todo momento y por todo lo que de ellos he recibido y que me llena de orgullo.
Por siempre, gracias.

A mis hermanos:

Jesús, Yolanda (y familia, Pedro, Jessica, Jazmín), Salvador, Miguel (y esposa, Luz María), Patricia (y familia, José Luis, Mauricio), Sara.

De quienes, (quizá esto no ha pasado por su mente), he aprendido mucho; agradezco todo su cariño y comprensión y deseo que todas sus ilusiones en la vida sean cumplidas.
Gracias por formar esta singular familia.

A mis familiares

A mi asesor:

M.C. Francisco Nava Morales

Por ser mi guía en el desarrollo de este trabajo.
Con respeto, gracias.

A la Universidad Nacional Autónoma de México

A la Facultad de Ingeniería

A mis profesores y amigos

A todos, mil gracias!.

Objetivo

El presente documento pretende proporcionar al lector, una serie de conocimientos generales en dos temas de interés actuales: Las llamadas *redes de computadoras*, y una de sus múltiples e importantes aplicaciones, el *correo electrónico*, con el fin de servir como elemento de apoyo en el desarrollo profesional del mismo. Se pretende también, considerando la base de conocimiento aquí presentada, obtener como resultado la creación de un *sistema de correo electrónico para una red de área local*.

Motivación

Durante los últimos años, las tecnologías de cómputo y telecomunicaciones han abierto las puertas a sistemas que puedan manejar un flujo de información sin precedente, esto implica un nuevo nivel de "poder" para las organizaciones. Es bien sabido que el conocimiento es por sí mismo poder.

Cada uno de los tres últimos siglos ha sido caracterizado por un cierto tipo de tecnología. El siglo XVIII representa la época de grandes sistemas mecánicos acompañando a la Revolución Industrial. El siglo XIX, la época de la máquina de vapor. Durante el siglo XX, la clave de la tecnología han sido la recopilación, procesamiento y distribución de la información. Entre otros desarrollos, hemos visto la instalación de redes telefónicas mundiales, la invensión del radio y la televisión, el nacimiento y crecimiento sin precedentes de la industria de las computadoras, y el lanzamiento de satélites de comunicaciones.

En los últimos años de este siglo, estas áreas han ido convergiendo rápidamente, y las diferencias entre recolectar, transportar, almacenar y procesar información están desapareciendo.

Así como nuestra habilidad para recopilar, procesar y distribuir información crece, la demanda para procesamientos de información más sofisticados crece aún más.

A través de una serie de tecnologías, se está logrando la integración de poderosas redes entre máquinas. Aunque las redes y las tecnologías que las crean son bastante complejas, las implicaciones son muy simples. Las redes de información ampliarán la habilidad de la gente para almacenar, reunir, preparar y comunicar información importante.

1. REDES DE COMPUTADORAS

Capítulo 1. Redes de Computadoras

1.1 Introducción

El surgimiento de las computadoras y las comunicaciones ha tenido una profunda influencia en la forma en que los sistemas de cómputo están organizados. El concepto del *"centro de cómputo"* como un cuarto con una computadora grande a la cual los usuarios traen su trabajo para procesamiento, está llegando a ser rápidamente obsoleto. Este modelo no tiene uno, sino dos defectos: El concepto de una simple computadora grande haciendo todo el trabajo, y la idea de los usuarios trayendo trabajo a la computadora en lugar de traer la computadora a los usuarios.

Este viejo modelo sirviendo a todas las necesidades computacionales de la organización, está siendo reemplazado por uno en el cual un gran número de computadoras separadas pero interconectadas realizan el trabajo. Estos sistemas son llamados redes de computadoras.

Una *red de computadoras* es un grupo de procesadores intercomunicados, independientes entre sí, con el propósito primordial de compartir recursos. Un sistema con una unidad de control y muchos esclavos no es una red de computadoras.

1.2 Historia de las redes de computadoras

Uno de los grandes proyectos de redes de comunicación de computadoras comenzó en 1969 bajo el patrocinio de la Advanced Research Proyects Agency (ARPA). La meta inicial de la red de ARPA (ARPANET) era interconectar centros de investigación sostenidos por la comunidad de ARPA para lograr dos objetivos: Desarrollar y examinar las técnicas de

comunicación de computadoras, y obtener los beneficios de compartir recursos de la comunidad de ARPA.

Inicialmente había cerca de 20 centros candidatos, cada uno con diferentes computadoras, modelos y sistemas operativos. Se contaba con equipos IBM 360/65, PDP10, UNIVAC 1108, Burroughs 6500 e Illiac.

ARPANET tomó ventaja de 2 tecnologías ya establecidas que tenían efecto en las redes de computadoras: Los sistemas de tiempo compartido y los sistemas de portadora común (ejemplo: la red telefónica). La idea era emular un diálogo de tiempo compartido usando la red telefónica más que la conexión usual punto a punto entre una terminal y una computadora, lo cual era común en un sistema de tiempo compartido. La mayor dificultad fue debida a lo inadecuado de la red telefónica para la tarea de comunicación. Habían 3 problemas principales: Primero, la red telefónica introducía errores excesivos en la transmisión; segundo, un circuito en particular podía fallar causando una falla al sistema; y tercero, la red telefónica introducía excesivo tiempo de respuesta. Lo que se necesitaba era un servicio de comunicación confiable entre computadoras con envíos de mensajes de aquí para allá.

Para vencer estos problemas, ARPANET introdujo los siguientes esquemas: Detección de error y retransmisiones, dos rutas físicamente separadas para colocar en ruta cada mensaje compensando fallas de circuitos, y el requerimiento de que el tiempo de retardo de propagación fuera menor de 0.5 segundos para lograr tiempos de respuesta cortos.

Se consideraron 3 posibles métodos de comunicación: Líneas dedicadas punto a punto, servicio de línea conmutada y servicio de mensajes conmutados (almacén y reenvío). El método de mensajes conmutados fue seleccionado debido a su gran flexibilidad, amplio ancho de banda y costo. Este sistema introdujo nodos en lugares apropiados de la red. Cada nodo en la red tenía un conmutador o centro de almacén y reenvío, con pocas conexiones a otros nodos; así los mensajes eran enviados de nodo a nodo hasta alcanzar su destino.

Los centros conmutadores estaban basados en pequeños computadores de propósito general llamados Interface Message Processors (IMP); manejaban envío de mensajes, verificación de errores en las transmisiones, y comunicación asíncrona a computadoras (Host). La topología ARPANET se muestra en la *figura 1.1*.

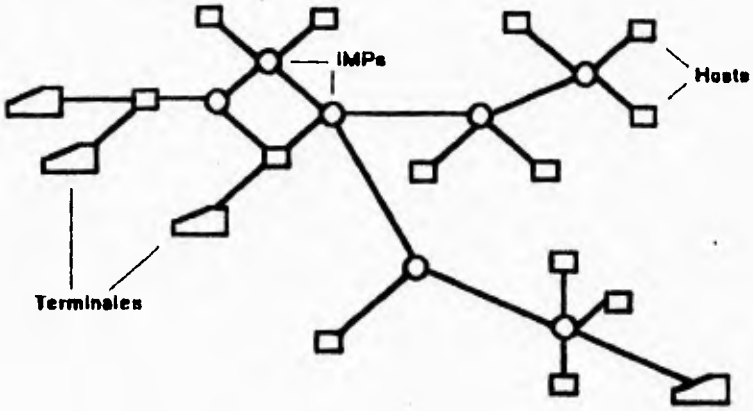


Figura 1.1 Topología ARPANET

Paralelo al desarrollo de ARPANET, un desarrollo similar estaba teniendo lugar en la Universidad de Hawaii a finales de los 60's, el sistema ALOHA. La meta era proporcionar otro sistema de red alternativo y determinar aquellas situaciones en las cuales las comunicaciones de radio son preferibles a las comunicaciones convencionales por alambre (líneas dedicadas por ejemplo).

Ambos, el ARPANET y el ALOHA, fueron desarrollados para interconectar computadoras localizadas "lejos" unas de otras. Basados en la experiencia proporcionada por ARPANET y por ALOHA, las instituciones de investigación así como las compañías de cómputo y procesamiento de datos comenzaron a experimentar con otras técnicas de comunicación. El resultado fue una proliferación de un número de esquemas usado para interconectar computadoras. Entre los sistemas de comunicación

desarrollados más importantes estaban el System Network Architecture (SNA) por IBM, DECNET por DEC y Ethernet por Xerox.

Surgió una situación muy difícil como resultado de la proliferación de diferentes técnicas de interconexión, protocolos e interfases; incompatibilidad de redes en sí. La comunicación entre computadoras de diferentes fabricantes era muy difícil.

(Conceptos como protocolo e interfase se analizarán posteriormente en este documento)

Los desarrollos en estándares de protocolos de comunicación tienen por objeto resolver los problemas de incompatibilidad entre redes. Los fabricantes de computadoras junto con algunas organizaciones y profesionales del campo están trabajando juntos para establecer un protocolo estándar común. Ejemplos de este esfuerzo de multiorganización son Manufacturing Automation Protocol (MAP) y Technical and Office Protocol (TOP) siendo desarrollados por GM y Boeing en los Estados Unidos, Computer Network for Manufacturing Applications (CNMA) en Europa.

1.3 Componentes de las redes de comunicación de computadoras

Una red de comunicación de computadoras es un sistema de computadoras interconectadas y otros dispositivos capaces de intercambiar información. Un componente genérico de la red (computadora o dispositivo) es referenciado como estación de trabajo de la red o simplemente *estación*. Una estación perteneciente a una red, es capaz de comunicarse con cualquier otra estación de la red. En otras palabras, la red representa un apoyo para la *conectividad*.

Un criterio que podemos usar para determinar si cierto dispositivo puede ser una estación de la red es que el dispositivo debe tener

suficiente "inteligencia" o estar programado para acoplarse en una comunicación en forma autónoma sin la asistencia de humanos. Por ejemplo, en un ambiente de manufactura, un robot industrial puede ser considerado una estación de red si tiene la capacidad de enviar y recibir mensajes con otras estaciones autónomas.

Para que dos estaciones se puedan comunicar entre sí debe haber un medio de transmisión sobre el cual las señales transmitidas puedan fluír. Además de este medio de transmisión o facilidades de transmisión, se requiere de algunos dispositivos de comunicación para ser interfase entre las estaciones y las facilidades de transmisión. Sin embargo, dos estaciones con una "liga" física entre ellas (usando una facilidad de transmisión) no tienen la garantía de poder comunicarse entre sí.

Para empezar, ambas estaciones deben estar "listas y en acuerdo" para intercambiar información. Aún después de esto, una estación podría enviar datos más rápido que la estación receptora pueda manejar. Por lo tanto, se requiere también de un conjunto de reglas o *protocolo* para llevar a cabo una comunicación exitosa entre las estaciones.

Una *facilidad de transmisión* es un término general usado para describir una facilidad que transporta señales de datos de una localidad a otra. Se incluyen, en general, repetidores, amplificadores, moduladores, filtros y medios físicos. Pueden involucrar un medio de transmisión o una combinación de medios de transmisión. Algunos ejemplos de medios de transmisión incluyen cables de par trenzado, cables coaxiales, fibras ópticas y espacio libre (para señales de ondas electromagnéticas).

Los *dispositivos de comunicación* son usados para hacer la interfase entre las estaciones y las facilidades de transmisión. Las típicas funciones de interfase son multiplexación, codificación de datos, modulación, conmutado, etc. De acuerdo a lo anterior, dichos dispositivos de comunicación son los multiplexores, codificadores, moduladores, conmutadores. También se les conoce como unidades de interfase de red.

Los protocolos de comunicación son esenciales para asegurar la eficiencia, corrección y transferencia adecuada de información entre las estaciones en la red.

1.4 Propósitos de una red de computadoras

Una red de computadoras presenta una gama muy amplia de funciones posibles a realizar, de ahí radica su gran importancia en el manejo de la información. Como propósitos fundamentales presenta los siguientes:

- Comunicación entre: procesos-procesos, procesos-usuarios, usuarios-usuarios.
- Distribuir la carga de trabajo (procesamiento distribuido).
- Centralizar cierto tipo de actividades, por ejemplo, un procesador dedicado 100% a atención a peticiones de una base de datos.
- Agilización de actualización y consultas de información.
- Evitar tiempos muertos debido a posibles fallas de procesadores individuales (aumentar la disponibilidad y confiabilidad de los equipos actuando en forma continua).
- Evitar o procurar evitar la posible duplicación de la información en varios equipos.

Cada uno de los puntos mencionados representa una importante mejora en eficiencia en el procesamiento de datos; se mejora el tiempo de respuesta de las transacciones en las bases de datos (por ejemplo en una actualización de cuenta bancaria por un retiro), la carga de trabajo en un proceso específico se puede distribuir en "procesos" más pequeños (en el campo de la industria del automóvil por ejemplo), la red puede ser usada como un medio de comunicación,etc.

1.5 Clasificación de las redes de computadoras

Algunos criterios pueden ser usados para clasificar a las redes de computadoras. Pero el seleccionar un criterio apropiado es realmente el punto importante. Un criterio no debiera estar basado en muchos detalles en específico sino en las características principales de una red.

Se tienen dos criterios útiles para una clasificación general de redes de computadoras. Un criterio está basado en la distancia que hay entre dos estaciones cualquiera, y el otro está basado en el método usado para la transferencia de mensajes. Algunos autores consideran otros criterios, como el de la técnica de multiplexación, control de la red, técnica de control de acceso al medio, etc; para la clasificación de redes. La desventaja con estos últimos criterios es que uno necesita estar familiarizado con dichos conceptos para el entendimiento de la clasificación de la red.

1.5.1 Criterio de clasificación basado en la transferencia de mensajes

De acuerdo al método usado para la transferencia de mensajes, se tienen los siguientes tipos de redes:

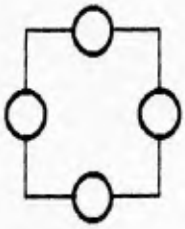
De difusión o broadcast, de relevo o relaying y de conmutado o switching. El primer tipo, usa un esquema de comunicación indirecto. Directo involucra la comunicación entre únicamente dos estaciones, receptora y transmisora; mientras que indirecto involucra la comunicación entre estaciones transmisora y receptora con al menos una estación intermedia entre ambas.

En las redes broadcast, varias estaciones comparten un canal de comunicación común. Los mensajes son transmitidos desde una estación

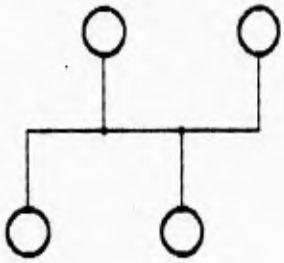
origen a una estación destino vía el canal de comunicación compartido sin el apoyo de una estación intermedia. Este tipo de redes es muy importante dado que la mayoría de las redes de área local (se analizarán posteriormente) pertenecen a este grupo.

Un problema principal ocurre involucrando interferencia en mensajes cuando al menos dos estaciones intentan enviar mensajes vía el canal compartido al mismo tiempo. Por tanto, debe haber una forma de regular el acceso al medio común de tal manera que las estaciones no se interfieran entre sí. (*Figura 1.2*).

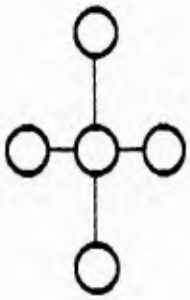
Una red relaying tiene estaciones que interactúan con el medio de comunicación formando un ciclo o anillo. La ruta entre una estación origen y una estación destino puede incluir estaciones intermedias. Un mensaje es "relevado" de una estación a otra hasta que alcanza su destino. La única topología para redes de este tipo es la de anillo, mostrada en la *figura 1.3*.



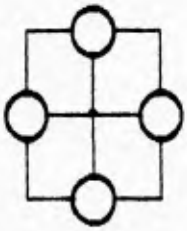
(1) Lazo o Loop



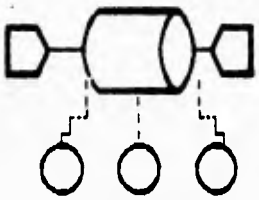
(2) Bus



(1) Estrella



(1) Malla



(2) Broadcast (radiación)

(1) punto a punto

(2) broadcast

Figura 1.2 Tipos de Topologías

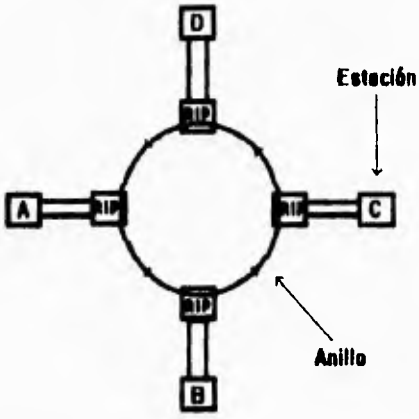


Figura 1.3 Topología de Anillo

La clave para entender este tipo de redes involucra la comprensión de cómo la estación interactúa con el medio de comunicación.

La interfase, llamado procesador de interfase de anillo (RIP), es típicamente un registro de corrimiento como el mostrado en la *figura 1.4*.

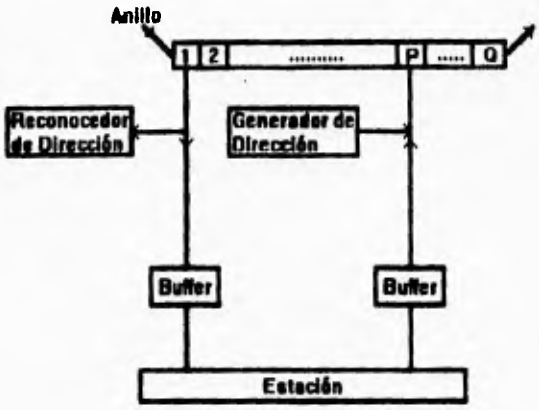


Figura 1.4 Ejemplo de un RIP

El tráfico de datos circula alrededor del anillo como una cadena de bits de un RIP al siguiente. Si un mensaje que entra en un RIP no es direccionado a su estación correspondiente, el mensaje es relevado inmediatamente al próximo RIP a través del registro de corrimiento. Quizá el mejor ejemplo de este tipo de red es el de la red Token Ring de IBM, basado en un protocolo de paso de token (se analizará posteriormente).

Cuando la transmisión de un mensaje entre una estación a otra involucra estaciones intermedias que ejecutan funciones de conmutado, la red correspondiente es referida como una red de conmutado. El conmuteo puede ser ejecutado ya sea usando conmutado de circuitos o conmutadores de almacén y reenvío.

El primer caso requiere que todos los segmentos de la ruta de la estación origen a la estación destino estén disponibles antes de que la transmisión se inicie. *Figura 1.5.*

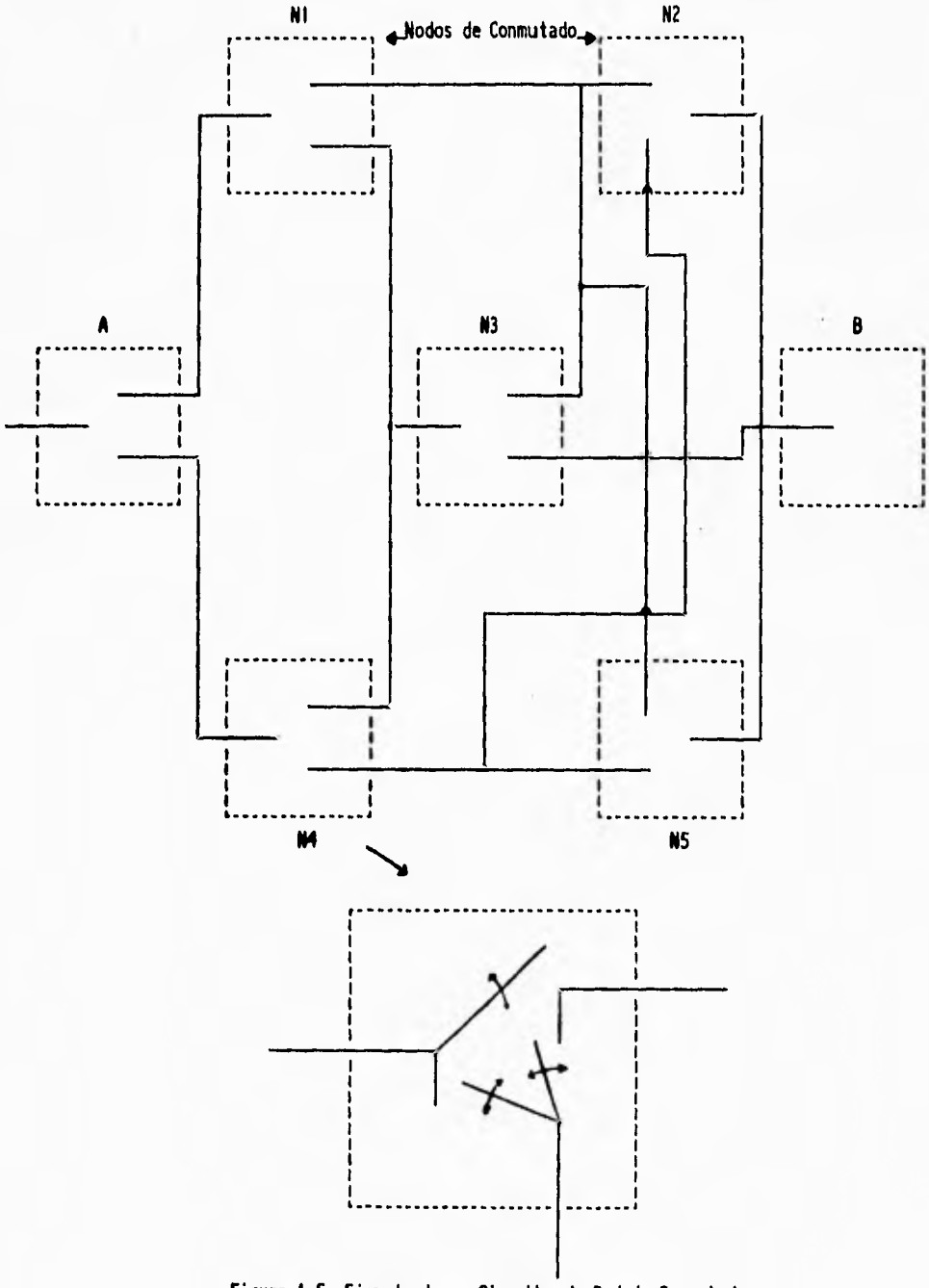


Figura 1.5 Ejemplo de un Circuito de Red de Conmutado

Si una estación A quiere enviar un mensaje a una estación B, primero se debe establecer una conexión física A-B; significa hacer conexiones apropiadas para que A y B estén conectadas.

En el segundo caso, los mensajes son colocados en ruta en cada estación de tal manera que alcancen la estación destino. Las estaciones que ejecutan la función de ruteo son llamadas nodos de conmutación. Como su nombre implica, los mensajes son almacenados temporalmente por las estaciones de conmutado antes de ser reenviados. ARPANET es una red de este tipo. Las redes de almacén y reenvio son llamadas también redes de punto a punto. (*Figura 1.2*).

1.5.2 Criterio de clasificación basado en la distancia entre estaciones (geográfica)

De acuerdo a la distancia que se tiene entre estaciones, se tienen los siguientes tipos de redes:

Redes cercanamente acopladas.- También se les conoce como redes de multiprocesadores, tienen estaciones localizadas una cerca de otra. La distancia entre estaciones es del orden de magnitud de unos pocos centímetros. Típicamente, las estaciones están ligadas a una tarjeta simple que se conecta con la tarjeta madre de la computadora host.

La comunicación entre estaciones se realiza en paralelo (16 bits a la vez) porque los buses internos de la computadora son usados para ésta. De hecho, una red de este tipo, es la única que proporciona comunicación en paralelo entre estaciones. Este tipo de redes tienen aplicación en sistemas que requieren un alto desempeño (ancho de banda grande, precisión, altos rangos de muestreo). Un ejemplo de esta red es uno usado en un robot controlador.

Muchos robots industriales tienen seis movimientos independientes (grados de libertad) para la manipulación. Típicamente, una microcomputadora se usa para controlar cada movimiento independiente, con otra microcomputadora usada para coordinación general y comunicaciones.

Así, el robot controlador consiste de siete microcomputadoras que cooperan conjuntamente para el control de los movimientos del robot. Los mecanismos de interconexión y comunicación en las redes de multiprocesadores están diseñados en una forma sin ningún estándar, aunque, afortunadamente, algunos estándares están comenzando a aparecer. (IEEE488, VME, Multibus II, y otros).

Redes de Area Local (LAN).- Las redes de área local constituyen un elemento principal en este documento. Son caracterizadas por una distancia entre estaciones del orden de magnitud de hasta algunos kilómetros. Típicamente, una LAN se presenta ubicada en una organización "sencilla" como un hospital, una universidad, una planta de manufactura o una oficina. La comunicación entre estaciones se realiza en forma serial (un bit a la vez) para decrementar costos en interconexión.

Las tres aplicaciones de LAN más importantes soportan el desarrollo de automatización en la oficina, el laboratorio y la fábrica. Usando una LAN en ambiente de oficina, un gerente puede revisar un documento financiero creado por una secretaria en otra terminal. Cuando el documento es terminado, la secretaria puede imprimirlo en una impresora disponible en la red.

Redes de Area Metropolitana (MAN).- Son similares a las LAN. Difieren en la intención de su aplicación. Típicamente apoyan la comunicación en cierto tipo de compañías. La distancia entre estaciones va en el rango de algunos metros a decenas de kilómetros.

Redes de Transporte Lejano (LHN).- La distancia entre estaciones es del orden de cientos de miles de kilómetros. Su principal aplicación

está en la interconexión de computadoras localizadas en diferentes ciudades, estados, y aún en diferentes países. Por ejemplo, comunicación de profesores de distintas universidades para el intercambio de ideas de algún tema específico. ARPANET pertenece a esta clasificación.

En general podemos clasificar a las redes en base a la distancia entre estaciones de la siguiente manera. (*Figura 1.6*)

Separación entre CPUs	Ubicación Típica	Nombre Genérico de la Red
< 1 m	Circuitos Integrados en Tarjetas	Máquinas de Flujo de Datos
1 m	Sistema o Gabinete	Multiprocesador
10 m	Sala o Cuarto	Red de Area Local (LAN)
100 m	Edificio	
1 km	Campus o Planta Industrial	
10 km	Ciudad	Red de Area Amplia (WAN)
100 km	Estado o País	
>= 1000 km	País o Continente o Planeta	

Figura 1.6 Clasificación de las Redes de Cómputo Basándose en su Extensión Geográfica

1.5.3 Propagación y retardos en la transmisión de mensajes

Un aspecto muy importante en las redes de computadoras es el desempeño, éste involucra, entre otras cosas, el cálculo del tiempo requerido por un mensaje para viajar a través del medio de transmisión y el cálculo del tiempo necesitado para enviar un número dado de bits una vez que la transmisión ha comenzado.

El tiempo requerido desde que un bit inicial es enviado (puesto en el medio) por el transmisor hasta que dicho bit llega al receptor es conocido como **retardo de propagación**. Los factores influenciando a este retardo son la longitud del medio L entre el transmisor y el receptor, la velocidad v en la cual la energía (eléctrica, óptica o electromagnética) llevando la información viaja en el medio, y otros factores dependiendo si la red es broadcast, relaying o switching.

Vamos a considerar una red conmutada donde el retardo de propagación depende de la configuración actual del procesador de interfase de anillo y los nodos de conmutado respectivamente. El tiempo transcurrido desde que el primer bit de un mensaje es transmitido hasta que el último bit del mismo mensaje es enviado por el transmisor es llamado **retardo de transmisión de mensaje**. Dos factores tienen influencia en este retardo: tasa de datos R (bit/sec), y longitud de mensaje X (bits). La tasa de datos es la velocidad a la cual bits individuales son puestos en el medio por el transmisor. El retardo de transmisión de mensaje queda representado como:

$$Tm = X/R$$

Otra medida de performance es el **retardo de mensaje** el cual definimos como el tiempo elapsado desde que el primer bit de un mensaje es colocado en el medio hasta que el último bit del mensaje alcanza la intefase de red de la estación destino.

$$Dm = Tp + Tm$$

1.6 Arquitectura de redes de computadoras

Una arquitectura de redes de computadoras es una serie de funciones y/o elementos, denominados capas en los cuales quedan comprendidas, bajo una división lógica, los aspectos fundamentales en una red.

Entre capas, del mismo nivel, contenidas en computadoras distintas se logra la comunicación a través de protocolos; un protocolo es una serie de reglas que rigen el intercambio de información.

Al grupo de reglas que rigen el intercambio de información entre capas contenidas en una misma computadora (nodo o host) se denomina interfase (*Figura 1.9*).

La conectividad, que es la capacidad de enlace y de intercambio de información entre equipos, se logra sólo en equipos con arquitecturas iguales. Para lograr la conectividad entre arquitecturas distintas se utilizan los llamados convertidores de protocolos (puentes, compuertas).

1.7 Modelo ISO/OSI (Introducción)

Aunque la separación física de computadoras comunicándose puede variar considerablemente de un tipo de aplicación a otro, en general una red de comunicación de computadoras puede ser representada como en el diagrama de la *figura 1.7*.

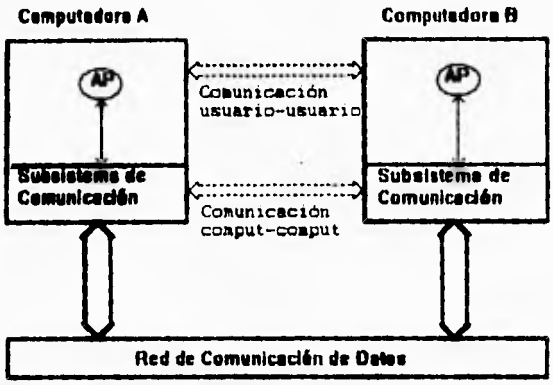


Figura 1.7 Comunicación de Computadoras

En el corazón de cualquier red de comunicación de computadoras está la facilidad de comunicación de datos, la cual puede ser una LAN privada o tal vez un número de redes interconectadas entre sí. Además de dicha facilidad, se requiere de cierta cantidad de hardware y software para cada computadora para el manejo de protocolos apropiados para la red.

La provisión de tales facilidades es sólo una parte de los requerimientos de la red, dado que en muchas aplicaciones, las computadoras pueden ser de diversos tipos; pueden usar lenguajes de programación diferentes, y más importante aún, diferentes formas de representación de datos. También pueden usar diferentes sistemas operativos; de allí que la interfase entre programas de usuario (aplicación), normalmente referidos como procesos de aplicación y los servicios de comunicación puedan ser distintos.

Por ejemplo, una computadora podría ser una pequeña computadora personal, mientras que otra pudiera ser un sistema multiusuario. Esto quiere decir que hace algún tiempo, sólo comunidades de computadoras "cerradas" (del mismo fabricante) podían comunicarse entre sí. System Networks Architecture (SNA) de IBM y Digital Network Architecture (DNA) de DEC son sólo dos ejemplos de ello, y no solucionan el problema de interconectividad universal, o sistemas "abiertos" de interconexión.

En un intento para arreglar este problema, ISO, a finales de los 70's, formuló un modelo de referencia para proporcionar una base común para la coordinación de desarrollos de estándares. El último llamado fue el permitir un proceso de aplicación en el cual cualquier computadora que soportara los estándares aplicables pudiera libremente comunicarse con un proceso de aplicación en otra computadora, soportando el mismo estándar. Este modelo fue llamado ISO Reference Model for Open Systems Interconnection (OSI).

Estaba relacionado con la estructura de software de comunicación que se requería para proporcionar un servicio confiable y transparente de comunicación de datos (el cual fuera independiente de cualquier equipo o convención de un fabricante específico) para soportar un amplio rango de aplicaciones.

Algunos ejemplos de procesos de aplicación que se comunicarían en una estructura abierta son:

- Un proceso en una terminal bancaria para iniciar una operación de retiro (por ejemplo) en un sistema remoto.
- Un programa en una computadora accesando un archivo en un sistema remoto.
- Un proceso en una aplicación de trabajo en una oficina accesando un servicio de correo electrónico.
- Un proceso controlando una pieza de equipo y enlazándose a un sistema supervisor de control de planta.

1.7.1 Modelo de referencia de ISO

Un subsistema de comunicación es una pieza de software compleja. Los primeros intentos para implementar tal software estuvieron basados en un complejo programa no estructurado (normalmente escrito en lenguaje

ensamblador) con muchos componentes interactuando. El resultado de tal sistema de comunicación era difícil de examinar y modificar.

Para saltar este obstáculo, ISO ha adoptado una característica para el modelo de referencia. El subsistema de comunicación está dividido en un número de capas, cada una con una función bien definida. Conceptualmente, estas capas pueden ser consideradas para ejecutar una de dos funciones generales: Funciones dependientes de la red o funciones orientadas a la aplicación.

Esto nos da pauta a tres ambientes operacionales distintos:

1. El ambiente de red, relacionado con los protocolos y estándares de los diferentes tipos de redes de comunicación de datos.
2. El ambiente OSI, el cual abarca el ambiente de red y añade protocolos y estándares adicionales orientados a aplicaciones, para permitir a los sistemas finales comunicarse entre sí de una forma abierta.
3. El ambiente de sistemas reales, el cual recae en el ambiente OSI y está relacionado con los sistemas propietario y servicios de los propios fabricantes; han sido producidos para funcionar en una tarea de procesamiento de información distribuido. Esto se muestra en la *figura 1.8*.

Ambas, funciones (dependiendo de la red y orientadas a la aplicación) son implementadas en un cierto número de capas de protocolo. Los límites de frontera entre cada capa, y aún más, las funciones realizadas por cada una de ellas, han sido seleccionados como resultado de la experiencia ganada en las actividades de los estándares anteriores.

Cada capa ejecuta una función bien definida en el contexto general de la estrategia de comunicación. Opera de acuerdo a un protocolo de intercambio de mensajes bien definido; ambos, datos de usuario e intercambio de información de control tienen correspondencia con una

respectiva capa en un sistema remoto. Cada capa tiene una interfase bien definida entre sí misma y la capa inmediatamente "arriba" de ésta. (*Figura 1.9*).

La estructura lógica del modelo de referencia de ISO se muestra en la *figura 1.10*.

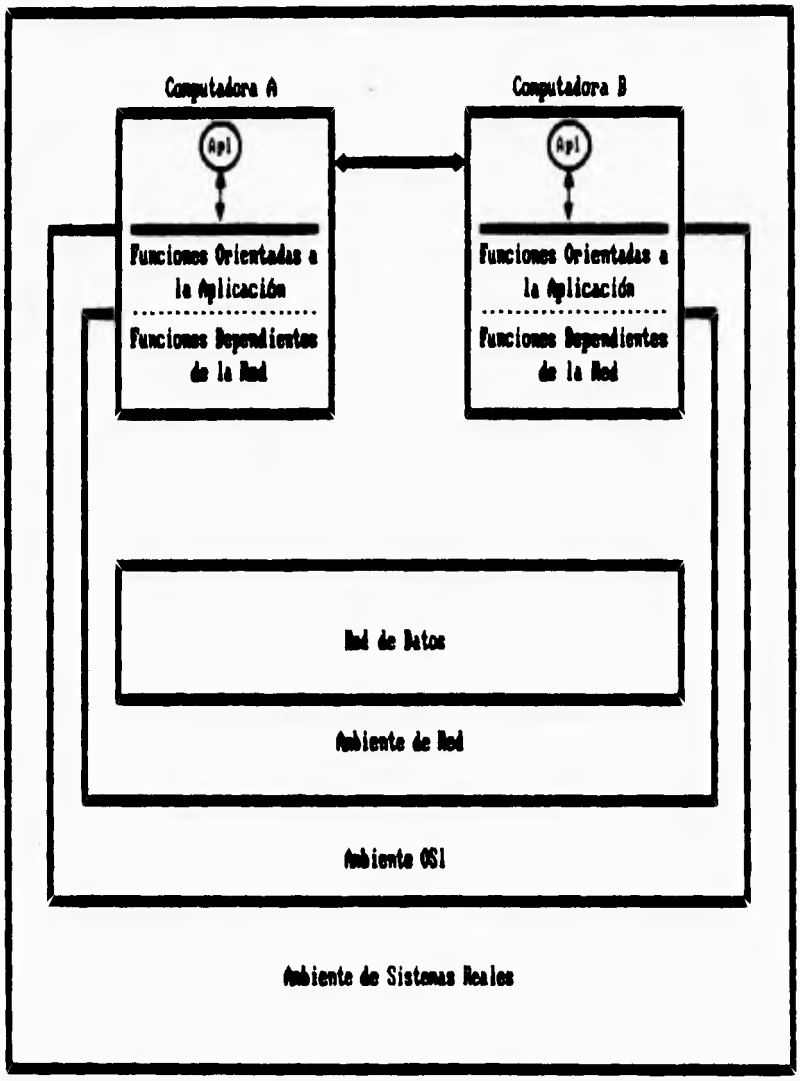


Figura 1.8 Ambientes Operacionales

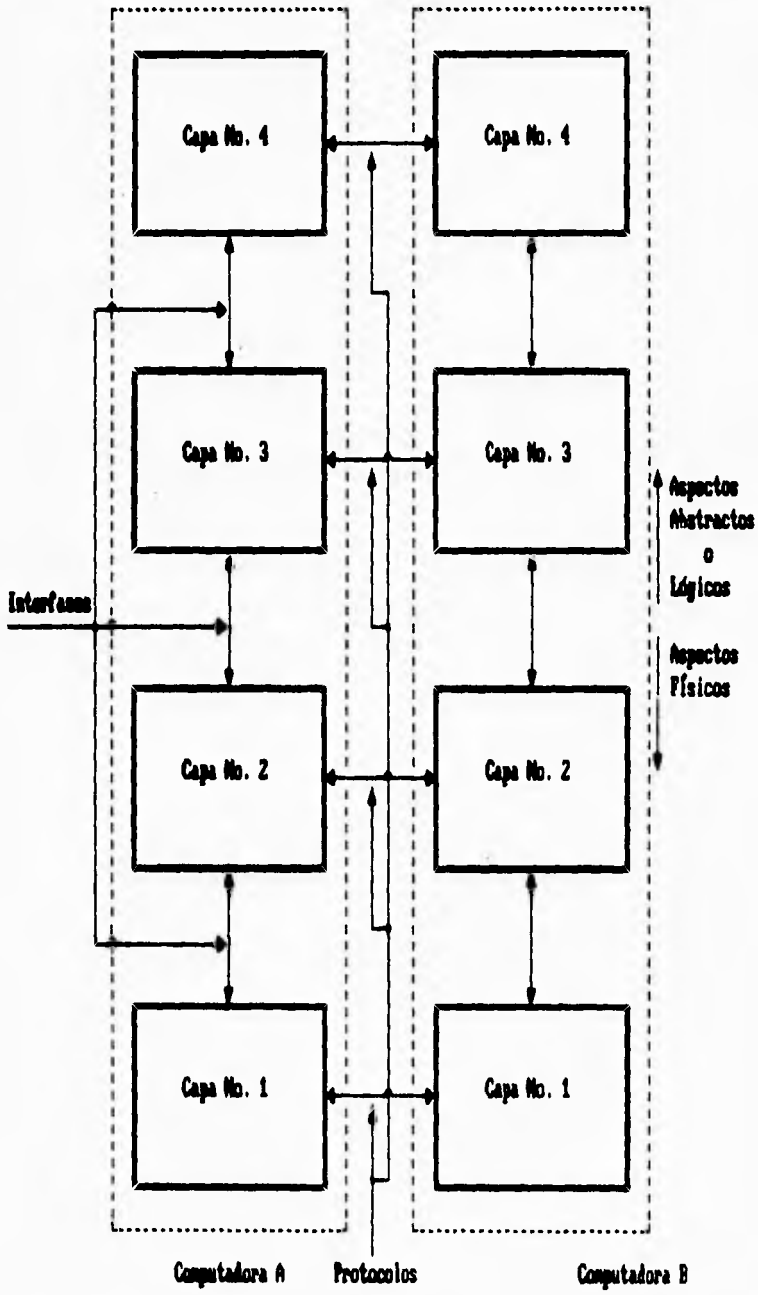


Figura 1.9 Reglas de Intercambio de Información

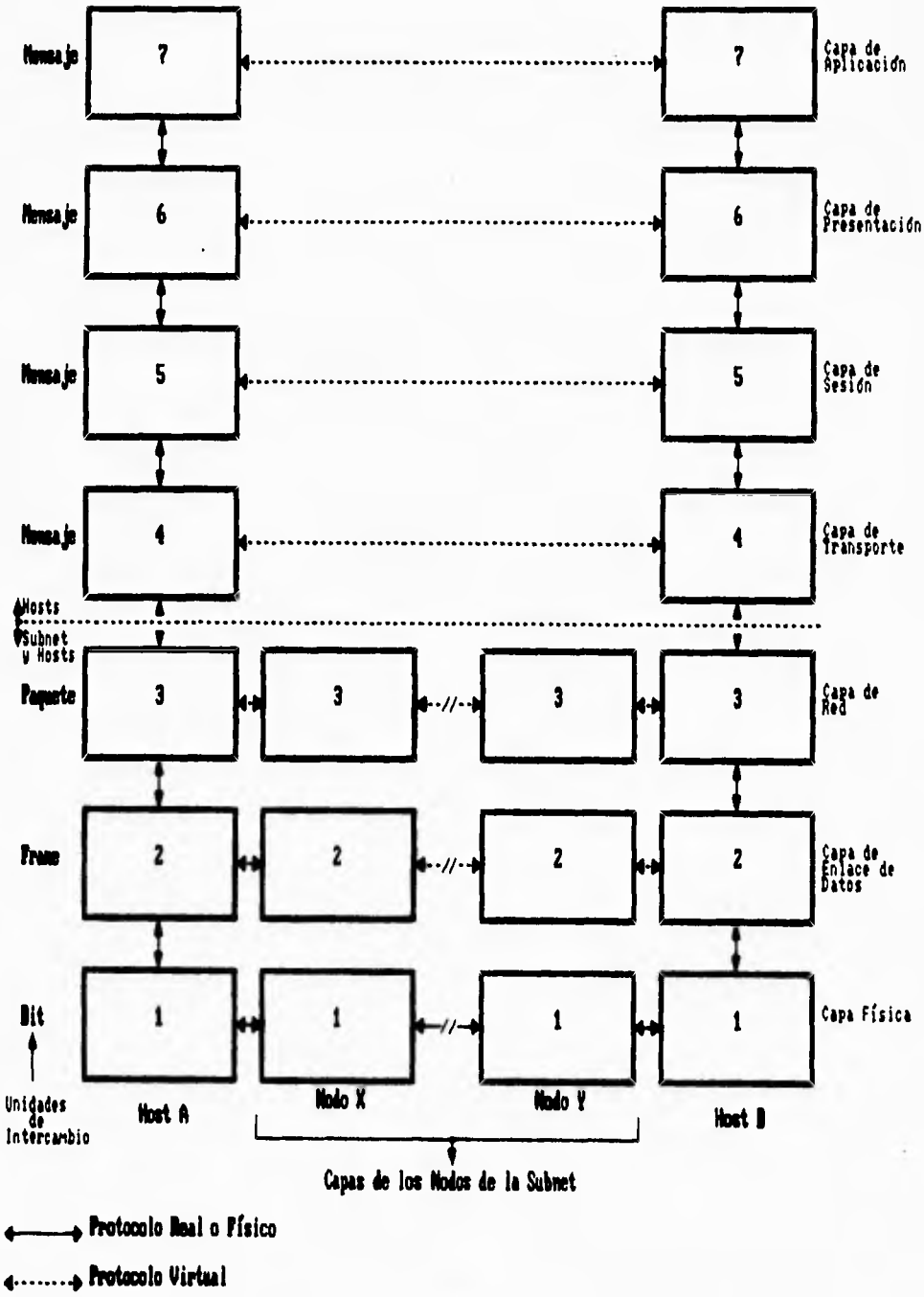


Figura 1.10 Modelo ISO/OSI (7 Capas)

1.7.2 Capas del modelo OSI

Capa No. 7 (capa de aplicación).- Proporciona la interfase de usuario a un rango de servicios de información distribuidos de la red. Se incluyen acceso y manipulación de archivos de transferencia, y servicios de intercambio de mensajes y documentos como el correo electrónico. El acceso a los servicios de aplicación se logra normalmente a través de un conjunto de primitivas bien definido, cada una asociada con ciertos parámetros, los cuales son soportados por el sistema operativo local.

En adición a la transferencia de información, esta capa también provee los siguientes servicios:

- Identificación de las estaciones que intervienen en la comunicación por un nombre o una dirección.
- Determinación de la disponibilidad presente de una estación en una comunicación destinada.
- Establecimiento de autoridad para comunicarse.
- Acuerdo en mecanismos de encriptado.
- Selección de la disciplina de diálogo, incluyendo la iniciación y procedimientos de liberación.
- Acuerdo en la responsabilidad (quién la tiene) para recuperación de errores.
- Identificación de estructuras, conjuntos de caracteres,etc.

Capa No. 6 (capa de presentación).- Está relacionada con la representación (sintaxis) de los datos durante la transferencia entre dos entidades de aplicación correspondientes. Para lograr una verdadera interconexión de sistemas abiertos, han sido definidos un número de formas de sintaxis de datos abstractos para ser usados por las entidades de aplicación junto con sintaxis de transferencia asociada. Esta capa por tanto, negocia y selecciona la sintaxis de transferencia de tal forma que la estructura de los mensajes siendo intercambiados entre dos entidades de aplicación se mantenga.

Para ilustrar los servicios proporcionados por la capa de presentación, vamos a considerar una conversación telefónica tomando lugar entre una persona hablando Francés y otra más hablando Español. Asumamos que cada una usa un intérprete y que el único lenguaje común entendido por ambos intérpretes es el Inglés. Aqui, cada intérprete debe traducir a y de Inglés a su lenguaje local. Los intérpretes representan entidades de la capa de presentación y los dos personajes representan entidades análogas de la capa de aplicación. La sintaxis concreta es el Inglés y el Francés y el Español son sintaxis locales.
En general proporciona una serie de rutinas que dan un formato adecuado a la información generada por aplicaciones o usuarios, incluye algunas conversiones necesarias (por ejemplo caracteres de control de impresora) y traducción de código (por ejemplo ASCII a EBCDIC).

Capa No. 5 (capa de sesión).- La diferencia clave entre las capas de sesión y de transporte es, que la capa de sesión está orientada a procesos, mientras que la capa de transporte está orientada a estaciones. Esto es, la capa de transporte es responsable por la transmisión confiable de mensajes entre dos estaciones, sin considerar la identidad de los procesos residentes en cada estación. La capa de sesión, es responsable por la transmisión confiable de mensajes entre dos procesos, sin considerar la identidad de las estaciones conteniendo los procesos. El propósito de esta capa es proporcionar los medios necesarios para que los procesos tengan un diálogo organizado y sincronizado en su intercambio de datos.

Capa No. 4 (capa de transporte).- Su principal función es aceptar datos de la capa de sesión, dividirlos en piezas más pequeñas, y, pasar estas piezas a la capa de red asegurándose que las piezas lleguen correctamente a su fin. También se incluye el control de errores. Forma la interfase entre las capas superiores (aplicación, presentación y sesión) con las capas inferiores (enrutamiento o de red, enlace de datos y física).

Capa No. 3 (capa de enrutamiento o de red).- El ruteo de mensajes a través de varias rutas posibles es una de sus funciones principales. Resuelve también la congestión y segmentación de mensajes. Para redes broadcast sólo existe una ruta, por tanto no hay "ruteo". La función de ruteo consiste en encontrar una ruta sobre la cual los mensajes puedan ser enviados.

Capa No. 2 (capa de enlace de datos).- Entre sus funciones se incluyen la transmisión libre de errores de frames (tramas de bits que constituyen la unidad de intercambio de esta capa), control de flujo, control de acceso al medio, etc.

Capa No. 1 (capa física).- Trata aspectos eléctricos, mecánicos y de procedimiento para la comunicación entre computadoras.

- Tipos de conectores
- Cableado
- Métodos de modulación/transmisión
- Tasa de transmisión
- Niveles de voltaje, etc.

Proporciona mecanismos eléctricos, funcionales y de procedimiento para tener acceso al medio físico.

1.8 Estándares en redes de computadoras (generalidades)

X.25.- Estándar de la CCITT para subnets. Comprende la definición de 3 primeras capas del modelo ISO/OSI (física, enlace de datos y enrutamiento).

La capa física recibe el nombre de X.21. Para aquellos equipos no compatibles con X.21, también se define un elemento más: PAD (Packet Assembler Disassembler) definido por el estándar X.3.

En México la Secretaría de Telecomunicaciones y Transportes ofrece un servicio de red pública tipo X.25 (TelePac).

SNA.- (Systems Networks Architecture) Es una red, tanto desde el punto de vista arquitectura como por estructura, sumamente compleja. SNA fue liberada en 1974 y ha evolucionado con el propósito de dar una solución a los requerimientos de comunicación de negocios que son muy complejos. En cuanto a su estructura, SNA distingue 4 tipos de nodos: *(figura 1.11)*:

DECNET (DNA).- Es una arquitectura que ha pasado por 5 fases de crecimiento:

- Fase 1. Nace por 1975, se consideran nodos tipo "small".
- Fase 2. Adopción de nodos tipo "full" (permitiendo comunicación con cualquier nodo en la red).
- Fase 3. En esta fase se adopta EtherNet como base de su arquitectura (3 primeras capas).
- Fase 4. Adoptan arquitecturas de 5 capas.
- Fase 5. Se apega al modelo ISO/OSI.

NAU (Network Addressable Unit). Novell Netware (basada en NetBios de DOS). Es una arquitectura muy popular para PCs. En sí, considera capas superiores, equivalentes a las capas 7,6,5 y 4 de ISO/OSI; las capas inferiores las implementa NetBios. Puede adaptarse a diversas estructuras y/o tarjetas de red, como son:

Arcnet
EtherNet
Token Ring

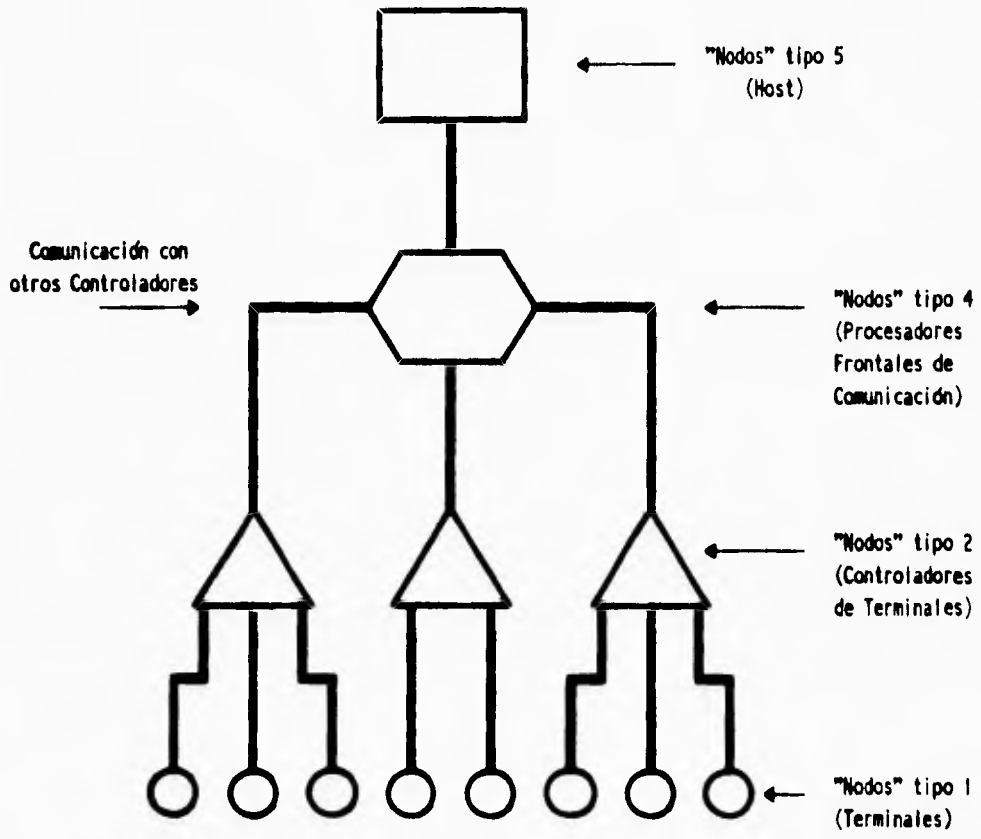


Figura 1.11 Estructura SNA

1.9 Aplicaciones de redes de computadoras

Ya se han mencionado algunos propósitos fundamentales de las redes de computadoras y ahora toca en turno mencionar algunas aplicaciones de ellas en la vida diaria.

- Consultas / actualización a bases de datos remotas distribuidas.
- Areas financieras / mercado.
- Acceso a grandes bancos de datos públicos o privados (consultas, intercambio de programas, etc).
- Situaciones de monitoreo / control de eventos en tiempo real, por ejemplo, monitoreo de condiciones meteorológicas, control de procesos industriales, etc.
- Transmisión de voz e imágenes.
- Correo Electrónico.

1.10 Ejemplos de redes públicas y privadas

Para redes públicas, la administración de la subnet, se da en una compañía u organismo el cual presta los servicios de comunicaciones a los usuarios, los cuales abonan sus hosts. En una red privada, tanto la subnet como los hosts pertenecen a una sola organización y ésta decide todo lo concerniente a la red.

Ejemplos: Redes públicas (CCITT X.25 (Comité Consultativo Internacional de Telegrafía y Telefonía)); redes privadas (BNA (Unysis), SNA (IBM), DECNET (DEC)).

2. CORREO ELECTRÓNICO

Capítulo 2. Correo Electrónico

2.1 Introducción

Una de las pérdidas de tiempo más grandes en una oficina es aquella ocasionada por la "llamada telefónica". El señor X llama a la señora Y, quien está lejos de su escritorio. Tiempo más tarde, Y regresa la llamada pero X está fuera o en otra línea. Y así sucesivamente...

Estudios independientes han mostrado que del 50% al 70% de todas las llamadas teléfonicas de negocios no alcanzan a su destino intencionado en el primer intento. El problema es que la persona que llama y a quien está dirigida la llamada deben estar ambas en sus teléfonos y disponibles para responder al mismo tiempo. Si la persona que llama pudiera simplemente escribir una nota y dejarla en el escritorio de la persona a quien se quiere llamar, el problema pudiera ser evitado.

El correo electrónico proporciona un camino para hacer esto. Además también elimina otro problema muy grande: La explosión de papel de oficina. Las oficinas generan una cantidad enorme de papel de trabajo, la mayor parte de éste en forma de reportes y memorandums internos; arriba del 80% de todos los documentos de negocios son textuales y/o numéricos (no gráficos), y se originan y permanecen en la misma organización.

2.2 Generalidades del correo electrónico

Una de las principales aplicaciones presentes en cualquier red es la llamada correo electrónico. Correo electrónico es un término ambiguo; aunque es un término genérico familiar a la mayoría de la gente en la comunidad de los negocios, no es fácil de definir.

Después de muchos meses de deliberación, la Electronic Mail Association (EMA) adoptó esta definición: El ***correo electrónico*** es un nombre genérico para comunicación no interactiva de texto, datos, imágenes o voz entre un transmisor y receptor(es) designados por sistemas utilizando enlaces de telecomunicaciones.

La definición es típica de las definiciones técnicas: En su esfuerzo por capturar el elemento técnico, de alguna manera, olvida al elemento humano. Aunque el corazón del correo electrónico podrían ser las comunicaciones no interactivas, su alma es la gente comunicándose con gente ya sea directamente o a través de máquinas intermediarias.

El correo electrónico, también conocido como sistema de mensajes basado en computadora (CBMS), es una facilidad que permite a los usuarios el intercambio de mensajes en sus terminales. Los mensajes no necesitan estar en papel a menos que el usuario (transmisor o receptor) desee una copia en papel del mensaje. Algunos sistemas de correo electrónico sirven únicamente a usuarios en una computadora individual; otros proporcionan servicio a través de una red de computadoras.

El correo electrónico tiene que realizar algunas funciones para la sociedad para representar un cierto valor a ésta. Una función principal es añadir un elemento de efectividad a las comunicaciones electrónicas, lo que no es obtenible a través del teléfono. La comunicación, de acuerdo a la teoría básica, consiste de un transmisor, un mensaje, y retroalimentación.

El teléfono presenta todos estos elementos, pero, desafortunadamente, el teléfono no es particularmente efectivo. Primero, una verdad común en la sociedad moderna, es que la mayoría de las llamadas telefónicas no logran alcanzar a los receptores debido a que éstos se encuentran lejos de sus teléfonos.

Segundo, cuando la llamada es completada, el mensaje por sí mismo puede estar "desordenado", el transmisor quizá no ordenó en forma adecuada sus ideas para comunicarlas al receptor. El teléfono es un medio que permite mucha retroalimentación en relación al mensaje, lo cual tiene un impacto en la efectividad de la llamada. Durante el proceso de retroalimentación, por ejemplo, el mensaje intencionado podría no ser liberado completamente o algunos puntos clave podrían ser olvidados.

El correo electrónico tiene los mismos elementos, pero están en una relación diferente. El transmisor puede estimar en un alto grado que el mensaje será recibido. Todo lo que se requiere es que el transmisor tenga conocimiento de que el receptor buscará su mensaje. El contenido no está abierto a discusión. Es almacenado y reproducido con un alto grado de precisión. Aunque está sujeto a interpretaciones variantes, es posible solicitar específicamente lo que uno quiere decir utilizando palabras clave. En comparación con el teléfono, el correo electrónico presenta un nivel bajo de retroalimentación sin distorsionar el contenido del mensaje.

Dado que la información liberada puede ser almacenada por el receptor, independientemente del tiempo y la localización, esto hace al correo electrónico un medio de comunicación altamente efectivo en comparación al teléfono, pudiendo servir como un medio de documentación.

El tráfico no interactivo, puede tener un tiempo de retardo desde su transmisión hasta su liberación sin tener un impacto en el valor de la información. Una gran cantidad de compañías tienen una creciente cantidad de tráfico electrónico no interactivo dentro de sus

organizaciones, incluyendo intercambio de datos electrónico, lo cual involucra, órdenes de compra, y otros documentos enviados entre compañías; formas de negocio electrónico interno incluyendo formas de requisición, órdenes de cambio, inventario, etc.

Todo este tráfico no interactivo, puede ser manejado en la misma red con enlaces usuario-usuario. Esto es una parte integral de la red de trabajo.

Se tienen los siguientes tipos de tecnologías de correo electrónico:

- Telex, TWX.
- Transmisión electrónica de imágenes.
- Comunicación de procesadores de palabra y teletex.
- Sistemas de mensajes basados en computadora. (Como el de este trabajo).
- Sistemas de correo de voz.
- Sistemas de conferencia por computadora.

Por último, cabe mencionar, que el correo electrónico tiene un beneficio directo: Reduce el ciclo de comunicaciones en comparación a los sistemas de correo físicos.

2.3 Sistema individual de correo electrónico

La más simple, y por mucho la más común forma de correo electrónico es la facilidad del sistema individual. Esta facilidad permite a todos los usuarios de un sistema computador compartido intercambiar mensajes. Cada usuario está registrado en el sistema y tiene un identificador único, generalmente el apellido paterno de la persona.

Asociado con cada usuario, se tiene también un buzón. La facilidad del correo electrónico es un programa de aplicación disponible a cualquier usuario firmado en el sistema. Un usuario puede invocar la facilidad del correo electrónico, preparar un mensaje, y *enviarlo* a otro usuario en el sistema. El acto de "enviar" simplemente involucra poner el mensaje en el buzón del receptor. El buzón es una entidad mantenida por el sistema de manejo de archivos, y está en la naturaleza de un directorio de archivos. Un buzón está asociado con cada usuario. Cualquier correo de *entrada* es simplemente almacenado como un archivo bajo el directorio del buzón de ese usuario. El usuario puede ir más tarde y localizar el archivo para leerlo.

El usuario lee mensajes invocando la facilidad del correo electrónico y leyendo los mismos. En la mayoría de los sistemas, cuando el usuario se firma, él o ella está informado si existe correo nuevo en el buzón.

Un sistema de correo electrónico individual ejecuta cuatro funciones:

- Creación.- Un usuario crea y edita un mensaje, generalmente usando una capacidad rudimentaria de edición. La mayoría de los sistemas permiten también al usuario crear un mensaje usando el editor del sistema o un procesador de palabras, y entonces incorporar el archivo resultante como el cuerpo del mensaje.
- Envío.- El usuario designa el receptor(es) del mensaje, y la facilidad almacena el mensaje en la(s) caja(s) apropiada(s).
- Recepción.- El receptor puede invocar la facilidad del correo electrónico para accesar y leer el correo enviado.
- Almacenamiento.- Transmisor y receptor pueden escoger salvar el mensaje en un archivo permanentemente.

2.4 Correo electrónico de red

En un ambiente distribuido, desearíamos ser capaces de intercambiar mensajes con usuarios asignados con otros sistemas. El tipo de correo electrónico anterior es muy limitado como se puede apreciar.

La *figura 2.1* sugiere una arquitectura de sistema interno requerida. Vamos a referirnos a una facilidad de correo de sistema individual como una facilidad de *correo nativo*. Para el correo nativo, se necesitan tres módulos principales. Los usuarios interactuarán con el correo nativo vía terminales; aquí, software de manejo de terminal es requerido. El correo se almacena como archivos en el sistema de archivos; por tanto, se requiere software para tal manejo. Finalmente, debe haber un paquete de correo nativo que contenga toda la lógica para proporcionar el servicio de correo a los usuarios.

Para extender este sistema al *correo de red*, se necesitan dos módulos más.

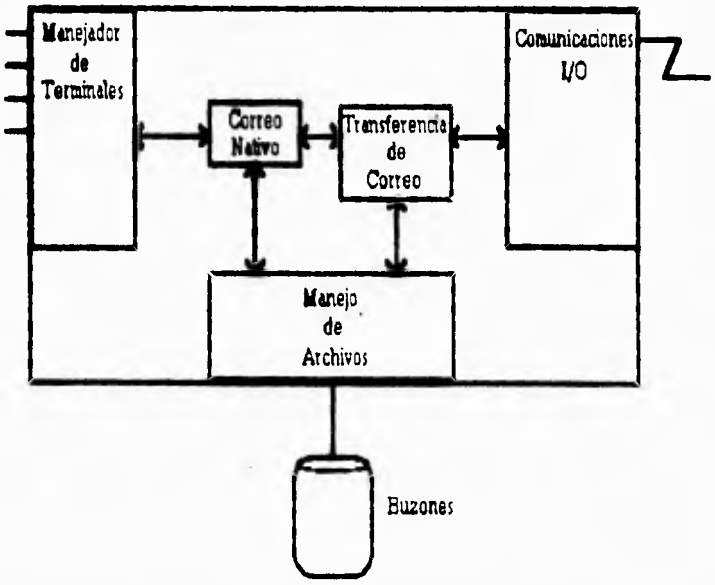


Figura 2.1 Estructura Conceptual del Sistema de Correo Electrónico

Debido a que vamos a comunicarnos a través de la red o del sistema de transmisión, se requiere lógica de entrada/salida de comunicación; en la mayoría de los casos, esto recaería en las capas de la 1 a la 6 del modelo OSI. Lógica de transferencia de correo, la cual "sabe" cómo invocar la función de comunicaciones, también se requiere para especificar la dirección del receptor en la red.

Hay que notar que en la *figura 2.1*, el usuario no interactúa directamente con el módulo de transferencia de correo. Idealmente, la interface de usuario para correo local o remoto debe ser la misma. Si el usuario designa un receptor local, el mensaje es almacenado en un buzón local. Si se designa un receptor remoto, el módulo de correo nativo pasa el mensaje al módulo de transferencia de correo para la transmisión a través de la red. De aquí es pasado al buzón adecuado.

En general, los vendedores ofrecen una versión en particular de su facilidad de correo electrónico, sin embargo, ésta sólo permitirá al usuario enviar correo a usuarios en sistemas del mismo vendedor. Como en otras áreas, se requieren estándares.

2.5 Servicios del correo electrónico (voz)

A finales de los 50's, algunos ingenieros emprendedores integraron una reproductora de cintas con un teléfono para desarrollar uno de los dispositivos electrónicos más simples en el mercado, la *máquina contestadora de teléfono*.

La máquina contestadora nunca ha sido reconocida como un dispositivo de correo electrónico serio, en lugar de eso, ha sido tratada como un consumidor especializado y un producto del negocio de la electrónica. Esto era, sin embargo, el inicio de una rama completamente nueva de la familia del correo electrónico, el *correo de voz*.

2.6 Resistencia al cambio

A pesar de los beneficios del correo electrónico, existe una considerable resistencia a la tecnología, especialmente de gente que no está familiarizada con dichos beneficios. Aunque este tipo de resistencia está decayendo, claramente indica que la gente es el juez de tomar la decisión de usar o no una nueva tecnología. Por tanto, la planeación e implementación del correo electrónico, debe estar relacionada con la educación de los usuarios acerca de sus beneficios y del propio control de uso.

Aunque los sistemas de correo de voz son más fáciles de usar que los sistemas basados en texto, existe aún una considerable resistencia a estos sistemas. Para probar esto, sólo se tiene que comprar una máquina contestadora de teléfono y usarla durante unas semanas. Las probabilidades de que las personas que llamen dejen mensajes son muy bajas debido a que nc les gusta "hablar con máquinas". A pesar de los mejores intentos por parte de los vendedores para distinguir el correo de voz de las máquinas contestadoras, ellos no han encontrado la manera de vencer el problema básico de la gente de no querer hablar con máquinas. Esto puede ser resuelto sólo por el tiempo y la exposición a un nuevo medio. Existen algunas barreras a vencer.

2.7 Limitación del medio

El correo de voz está limitado por el medio de voz, el cual es altamente efectivo en el intercambio de grandes volúmenes de información en conversaciones cara a cara, pero, no particularmente efectivo para un medio de almacena y sigue, excepto para mensajes cortos.

Un sistema de mensajes basado en computadora (CBMS) para texto, por ejemplo, puede ser usado para enviar mensajes de pocas palabras o

documentos de algunas páginas. Poca gente, sin embargo, puede transmitir mensajes con voz largos hacia una máquina efectivamente, y aún menos gente puede escuchar mensajes muy largos.

El flujo PCs, más aún, ha incrementado la necesidad de sistemas de correo basados en texto debido a que el material original es frecuentemente capturado directamente en un formato de computadora. Si el correo de voz llega a ser un subconjunto de un sistema de mensajes integrados que mantenga texto y voz, el sistema de texto llegará a ser anticuado.

2.8 Aplicaciones

Los sistemas de correo de voz son muy fáciles de usar debido al simple manejo de voz y al menú de operaciones del teclado del teléfono. Su capacidad, sin embargo, está limitada debido a la limitación de la voz como un medio almacenado. La gente necesita retroalimentación visual para mantener mensajes largos; esto reduce la efectividad de un sistema de correo de voz a usarlo para mensajes cortos, lo que limita su viabilidad. En general eneral, las razones principales para el uso de un sistema de correo de voz son:

- Eliminación del marcaje telefónico
- Transmisión de mensajes a un grupo, lo cual toma considerablemente menos tiempo que llamar a cada persona individualmente
- Reducción de costos de llamadas telefónicas dentro de una organización

3. REDES DE AREA LOCAL

Capítulo 3. Redes de Area Local (LANs)

3.1 Introducción

Las redes de área local ya fueron mencionadas en el primer capítulo de este documento, sólo falta mencionar que recientemente ha habido mucho interés en ellas. De hecho, algunos sectores de la industria de la computación declararon a 1985 como el año de la red de área local.

Las redes de área local son una de las tecnologías basadas en computadora de más rápido movimiento.

Existen numerosas razones para tener interés en redes de computadoras, en particular en redes de área local; tal vez las más significativas involucran: Compartir recursos, diseminar la información, procesamiento distribuido, automatización de oficinas, laboratorios y ambientes de manufactura.

3.2 Especificación

En una red de área local, las computadoras se comunican entre sí por medio de paquetes de mensajes. Cada paquete contiene información acerca de la dirección del receptor y transmisor respectivos para propósitos de enrutamiento. Una red de área local siempre cuenta con uno o más servidores de archivos y la mayor parte del procesamiento de información ocurre en el procesador local, no sobre el servidor mismo.

Menos formalmente, una red de área local es un grupo de procesadores conectados entre sí por cables. Cada computador contiene una tarjeta adaptadora de red y software de soporte a la misma, así como una dirección única en la red y se conoce como nodo o estación de trabajo.

La comunicación entre las computadoras en la red es a través de paquetes de mensajes llamados frecuentemente frames, dichos frames son enviados y recibidos por el adaptador de red y el software de soporte.

El software de soporte a la red envía frames para muchos propósitos, incluyendo los siguientes:

- Abrir una sesión de comunicaciones con otro adaptador
- Enviar datos a una computadora
- Reconocimiento de la recepción de un frame
- Enviar un mensaje a todas las computadoras
- Cerrar una sesión de comunicaciones

Las comunicaciones entre computadoras son fáciles de visualizar en esta forma; por medio de programación se pide a un protocolo, tal como NetBios , enviar o recibir un mensaje, y el protocolo trabaja con el software de soporte para enviar y recibir los frames y reconocimientos apropiados.

La siguiente figura muestra la capa de un frame típico, claro que redes diversas implementan y definen los frames de forma distinta pero los siguientes datos son típicos en todas las implementaciones (*figura 3.1*).

*** Dirección del transmisor en la red**
*** Dirección del receptor en la red**
*** Identificación del contenido del frame**
*** Registro de datos o mensaje**
*** Código de Redundancia Cíclica para manejo de errores**

Transmisor ID	Receptor ID	Tipo de Frame	Datos/ Mensaje	Código de Red. Cíclica

Figura 3.1 Capa Básica de un Frame

3.3 Tipos de LANs

Se distinguen básicamente en dos categorías: Sensado de portadora y paso de token. Ethernet es un ejemplo de una red de sensado de portadora y Token Ring es un ejemplo de red de paso de token.

El instituto de Ingenieros Eléctricos y en Electrónica (IEEE) define un conjunto de estándares para las características físicas de ambos tipos de redes: IEEE 802.3 (Ethernet) e IEE 802.5 (Token Ring). Por supuesto que algunas redes de área local no se "apegan" con alguno de estos estándares, por ejemplo: ARCnet (Datapoint Corporation, Standard Microsystems Corporation, Thomas-Conrad), VistaLAN/PC (Allen-Bradley, Inc.), STARLAN (AT&T), PC Network (IBM), LANTastic (Artisoft, Inc.), Omninet (corvus Systems, Inc.), ProNET (Proteon).

1) Basadas en sensado de portadora, por ejemplo, (IEEE 802.3 EtherNet).

Cuando un host/nodo desea transmitir un paquete, debe "escuchar" primero el canal (sensado de portadora); si no sensa que exista una transmisión de otro paquete por el canal, envía su paquete; si existe otro paquete circulando en la red, espera a que el canal se libere.

Este protocolo se denomina CSMA (Carrier Sensing Multiple Access). En ciertos casos, dos nodos que deseen transmitir, pueden sensar que el canal está libre; bajo este supuesto, los nodos transmitirán su paquete, entremezclándose en el canal ambos, lo que origina una situación de colisión. La solución a este problema se realiza de la siguiente forma: Los nodos que provocaron la colisión detienen inmediatamente el envío de los paquetes, esperan un tiempo aleatorio (distinto para cada uno de ellos) y reintentan sensar el estado del canal para repetir, de esta manera, el algoritmo (CSMA/CD CD:Colision Detection).

La implementación comercial de este método se conoce como EtherNet. (*figura 3.2*).

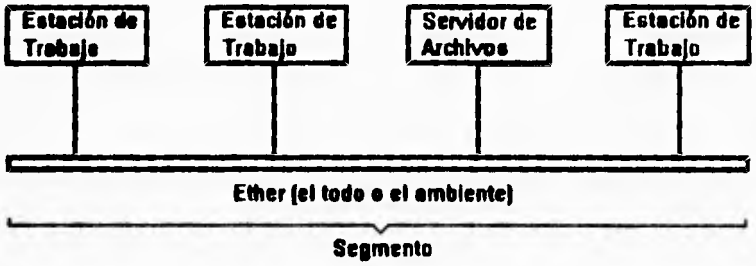


Figura 3.2 Red Ethernet

Características Principales:

- Tasa de transmisión de 10 Mb/s
- Topología típica: bus lineal
- El Ether suele utilizar:
  - -Cable(coaxial RG-58, fibra óptica, par trenzado)
  - -Tranceptores
  - -Terminadores
- Puede tener hasta 1024 nodos (por segmento), varios segmentos pueden acoplarse mediante repetidores
- Los nodos deben tener una separación mínima de aproximadamente 1.5 m.

La *tabla 3.1* muestra algunos estándares en Ethernet, cabe mencionar que el tipo de cableado aquí mostrado será abordado posteriormente.

MEDIO	DISTANCIA-SEGMENTO (Sin Repetidores)	Topologia
Coaxial Delgado	185 m	Bus
Coaxial Grueso	500 m	Bus
Par Trenzado No Blindado	100 m	Bus Interno
Fibra Óptica	1 km	Bus Interno

Tabla 3.1 Estándares en Ethernet

Ventajas:

- Hardware disponible en un gran número de vendedores
- Estándar maduro, con más de 10 años en el mercado

Desventajas:

- La regla de 4 repetidores reduce la distancia máxima del bus
- Los tiempos de respuesta son impredecibles cuando aumenta la carga en la red
- El protocolo CSMA/CD reduce el desempeño de la red
- No se recomiendan más de 100 nodos en una red

2) Basadas en transmisión (paso) de token, por ejemplo, (IEEE 802.5 Token Ring).

Existe un paquete especial circulando a lo largo de la red, denominado token. Una vez que una estación desea enviar un paquete de información, debe "adueñarse" del token, el cual indica al resto de las estaciones que la red está ocupada, (*Figura 3.3*).

Si una estación desea enviar un paquete y se encuentra con el token ocupado, deber esperar a que pase nuevamente éste pero ya liberado. La liberación la realiza la estación que lo ocupó o generó. El paquete de información debe contener información sobre el o los identificador(es) de el (los) nodo(s) destino.

Se puede dar un esquema de autoelección de generación de token, debe reportar al resto de los nodos su elección.

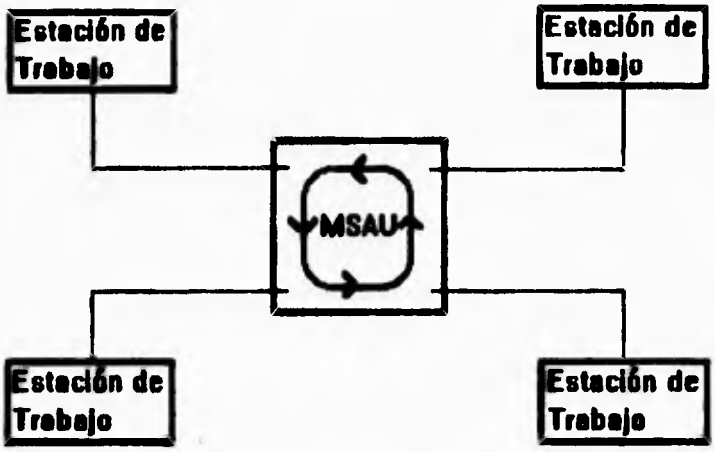


Figura 3.3 Red Token Ring

Especificaciones Comerciales (Token Ring IBM):

- Tasas típicas de transmisión de 4Mb/s, 16Mb/s
- Número de estaciones en un anillo < 256,000
- El cable puede ser coaxial(RG-58, RG-62), par trenzado y fibra óptica
- Cada estación se acopla a la red a través de un adaptador al anillo

La *tabla 3.2* presenta algunos estándares en Token Ring.

MEDIO	DISTANCIA-ANILLO (Máxima)	No. DE ESTACIONES POR ANILLO
Par Trenzado Blindado	330m	260
Par Trenzado No Blindado	100m	104
Fibra Óptica	2 km	----

Tabla 3.2 Estándares en Token Ring

Ventajas:

- El desempeño del Token Ring garantiza tiempos de respuesta confiables
- No existe la colisión de datos
- Velocidad de trabajo flexible, 4 o 16 Mbps

Desventajas:

- El protocolo es robusto, pero requiere de mecanismos complejos para asegurar un correcto desempeño
- Cuando una estación falla, el anillo entra en proceso de recuperación de la misma, lo que disminuye el desempeño de la red

3.4 Componentes físicos de una red de área local

Los componentes de hardware básicos en una red de área local son: Tarjetas adaptadoras de red, cables y unidades de acceso o repetidores.

Dependiendo del tipo y complejidad de la red sobre la que se está trabajando, se pueden incluir puentes, ruteadores, concentradores, etc. No se requiere entender los términos puentes, ruteadores y concentradores para conceptualizar una red de área local para programación, basta con saber que se conocen como convertidores de protocolos y se utilizan para lograr la conectividad entre arquitecturas distintas.

3.4.1 Capa física de una LAN

En general, la capa física de una LAN reside en la tarjeta adaptadora de red. Estas ejecutan 7 pasos principales durante los procesos de envío y recepción de un frame. Cuando ciertos datos están siendo enviados, los pasos siguientes son ejecutados en el orden mostrado; cuando los datos están siendo recibidos, los pasos se invierten:

1. Transferencia de datos.- Los datos son transferidos de la memoria de la computadora a la tarjeta adaptadora o de la tarjeta adaptadora a la memoria de la computadora por el DMA, memoria compartida o entrada/salida programada.

2. Buffering.- Mientras están siendo procesados por la tarjeta adaptadora de red, los datos son mantenidos en un buffer. Este buffer o espacio de memoria temporal da acceso a la tarjeta para un frame entero a la vez y permite a la tarjeta manejar la diferencia entre la tasa de transmisión de datos en la red y la tasa de transmisión de datos de la computadora.

3. Formación de frame.- El adaptador de red tiene que "cortar" (o re-ensamblar) los datos dentro de cadenas de datos manejables. La mayoría de las redes usan un tamaño entre 1 y 4 kilobytes. Se fijan un frame header y un frame trailer al paquete de datos. (En la recepción, tanto el header como el trailer son eliminados).

4. Acceso al cable.- En una red CSMA/CD, el adaptador de red asegura que la línea está desocupada antes de enviar datos. En una red token ring, el adaptador espera hasta que obtiene el "token" de tal manera que lo pueda usar.

5. Conversión paralela y serial.- Los bytes de los datos en el buffer son enviados o recibidos a través de los cables en forma serial,

un bit tras otro. La tarjeta adaptadora hace esta conversión antes de la transmisión o después de la recepción.

6. Codificación y decodificación.- Se forman las señales eléctricas que representan los datos siendo enviados o recibidos. La mayoría de los adaptadores de red usan el código Manchester que tiene la ventaja de incorporar información del tiempo dentro de los datos usando períodos bit. En lugar de representar un 0 como la ausencia de electricidad y un 1 como su presencia, tanto el 0 como el 1 son representados por cambios en la polaridad en relación a períodos de tiempo muy cortos.

7. Envío y recepción de pulsos.- Los impulsos codificados eléctricamente representando los datos (frame) son amplificados y enviados a través del cable. (En la recepción, los impulsos son decodificados).

Por supuesto, la ejecución de todos estos pasos toma sólo una fracción de segundo. Las tarjetas adaptadoras de red y el software de soporte reconocen y manejan errores que pueden ocurrir debido a interferencia eléctrica, colisiones (en CSMA/CD), o mal funcionamiento del equipo.

Los diferentes tipos de adaptadores de red varían no sólo en el método de acceso y protocolo sino también en lo siguiente:

- Velocidad de transmisión
- Tamaño de buffer
- Diseño del bus (8-bit, 16-bit o Micro Canal)
- Velocidad del bus
- Compatibilidad con diversos CPUs,etc.

3.4.2 Sistema de cableado

Los sistemas de cableado para las redes de área local varian ampliamente en apariencia, características, propósito y costo.

Los tres caminos más populares para ligar computadoras conjuntamente en una red de área local son: El sistema de cableado de IBM, el sistema de distribución de premisas AT&T y el concepto de cableado de DEC llamado DECconnect.

Los sistemas de cableado mencionados pueden ser categorizados dentro de 3 tipos de cable distintos:

- Par trenzado (cubierto y no cubierto)
- Cable coaxial
- Fibra óptica

El *par trenzado* lo forman, como su nombre implica, alambres aislados con un número de trenzas por pie. Trenzar los alambres reduce interferencia eléctrica (atenuación). Cubiertos se refiere a la cantidad de aislamiento alrededor del alambre y por tanto su inmunidad a errores de corrupción de datos. Los no cubiertos son más familiares a nosotros, son frecuentemente usados por la compañía de teléfonos. Este tipo de cableado es económico y sus rangos típicos de transmisión van de 300 bps a 1 Mbps.

El *cable coaxial* es común a nuestra vida diaria, frecuentemente se encuentra conectado a la parte posterior del aparato televisor o en los equipos de audio. Hay delgado y grueso. El grueso ofrece mayor inmunidad al ruido y es más difícil dañarlo y requiere de un conector "vampiro" para conectarse a la red de área local. El delgado maneja distancias más cortas para la señal pero usa un conector BNC, es de menor costo y ha llegado a ser el estándar oficial.

Consiste de un alambre conductor central rodeado por un aislamiento dieléctrico y una malla conductora, y finalmente cubierto con un aislante externo. El conductor central transporta la señal mientras que la malla conductora sirve como la referencia a tierra.

Los rangos típicos en este tipo de cableado van de 1 Mbps a 50 Mbps. Ethernet y MAP usan cable coaxial como medio físico.

Finalmente, la *fibra óptica* usa luz para acarrear información. Puede ser usada para enviar gran cantidad de datos a distancias muy grandes a altas velocidades, pero es cara y difícil de instalar y mantener. El cable de fibra óptica consiste de un núcleo de fibras de diámetro medido en micras, rodeadas de material sólido y cubiertas por una funda protectora. Primero, las fibras fueron hechas de vidrio y después de plástico.

El origen de luz para la fibra óptica es de un LED o diodo emisor de luz; la información es generalmente codificada variando la intensidad de luz. Un detector al final de la fibra óptica convierte la señal recibida en impulsos eléctricos. Entre más delgada es, es más cara.

Aún no es un medio tan desarrollado para LANs como el cable coaxial pero aparece muy prominente para el futuro. Maneja un ancho de banda grande y es de baja inmunidad al ruido. Su rango de transmisión va hasta algunos cientos de Mbps.

La *figura 3.4* muestra los tipos de cable mencionados.

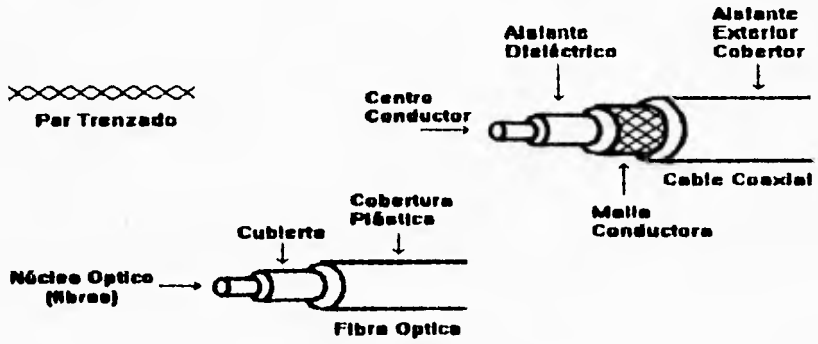


Figura 3.4 Sistema de Cableado

3.4.3 Unidades de acceso y repetidores

En una red de manejo de token, los cables de las estaciones de trabajo están centralmente conectados a una unidad de acceso multi-estación (MSAU o MAU), la cual mantiene información referente a qué estaciones de trabajo en la red son vecinas y cuál está adelante y cuál atrás de la secuencia.

Es un trabajo pesado, la mayoría de las MSAUs no necesitan ser conectadas a una fuente de energía eléctrica. La excepción a esto es una MSAU que se maneje para grandes distancias o bien que se use cable de par trenzado no cubierto en redes de área local de alta velocidad; en estos casos una fuente de energía eléctrica ayuda a la señal regenerándola.

Una MSAU tiene 8 puertos para conectar de 1 a 8 dispositivos Token Ring. La MSAU tiene 2 puertos adicionales etiquetados RI (Ring-In) y RO (Ring-Out) que son usados para ligar varias MSAUs juntas (en una cadena) cuando se tienen más de 8 estaciones en la red de área local.

Toma algunos segundos "abrir la conexión del adaptador" en una red Token Ring; durante este tiempo, la MSAU y la tarjeta adaptadora

ejecutan un pequeño diagnóstico de verificación después del cual la MSAU establece a la estación de trabajo como una nueva vecina en el anillo. Después de ser establecida como una estación de trabajo activa, la computadora es ligada a ambos lados de sus vecinos (de acuerdo a la posición en la red definida por la MSAU); en su turno, la tarjeta adaptadora Token Ring acepta el token, lo regenera y lo envía a través de la MSAU en la dirección de su vecino siguiente en la red.

En una red Ethernet, el número de conexiones y sus distancias interviniendo son factores limitantes. Los repetidores son usados para regenerar la señal cada 500 m aproximadamente. Si dichos repetidores no fueran usados, la señal seguramente tendría errores. Debido al tiempo manejado para la detección de colisión, sólo cinco segmentos de 500 m y cuatro repetidores pueden ser colocados en serie antes de que el retardo de propagación llegue a ser más grande que el máximo período de tiempo permitido para detectar colisión, de otra manera, no se detectaría ésta.

En general, los límites en la distancia y número de estaciones especificados en IEEE 802.3 Ethernet e IEEE 802.5 Token Ring están siendo sobrepasados a través de tecnologías de fibra óptica, repetidores inteligentes, puentes activos, enrutadores y puentes.

3.5. El Token

En principio, un token es el medio por el cual cada estación de trabajo puede reconocer cuando su turno para poder transmitir ha llegado. Un token circula el anillo continuamente hasta que una estación de trabajo tiene un frame para enviar. Por otra parte, los frames de datos (se verán posteriormente en este documento) pueden contener mensajes que las aplicaciones envían a otras sobre el anillo, y también algunas veces contienen mensajes internos usados privadamente entre las tarjetas adaptadoras de la red para propósitos del manejo del anillo. (*Figuras 3.5 y 3.6*).

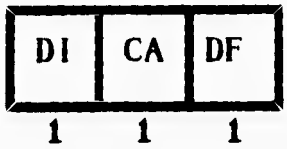


Figura 3.5 Formato del Token

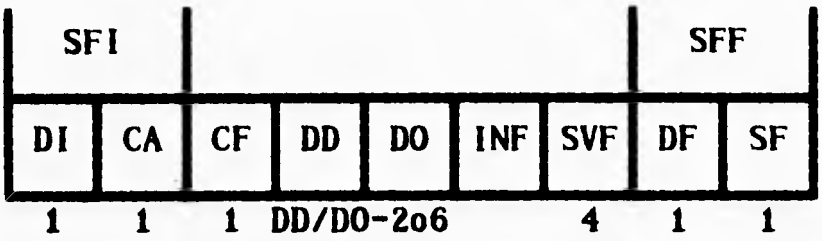


Figura 3.6 Formato del Frame de Datos (Token Ring)

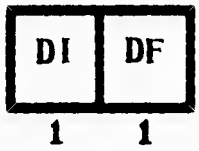


Figura 3.7 Formato de Secuencia de Aborto

* Longitud de cada campo en bytes

Existe otro elemento llamado "secuencia de aborto" el cual es usado para interrumpir o terminar la transmisión actual (*figura 3.7*). Un token está formado de 3 bytes (24 bits) de longitud y contiene 3 campos:

- Delimitador Inicial (DI)
- Control de Acceso (CA)
- Delimitador Final (DF)

El delimitador inicial (DI) es el campo que está al inicio del token (de igual forma en un frame). Consiste de una secuencia especial de bits usada para lograr "transparencia" de datos.

Hay 4 símbolos (no de datos) de 1 bit cada uno y 4 bits "normales" 0 en el campo, en total 1 byte.

El siguiente campo es el de control de acceso (CA) el cual está subdividido en 4 tipos de datos:

PPP	T	M	RRR

En el cual PPP representan a los bits de prioridad, T es el bit de token, M es el bit monitor y RRR son los bits de reservación. Cada token maneja una prioridad que va del 0 al 7 (con el 7 siendo la prioridad más alta). Una estación de trabajo puede usar la red (cambiar un token a un frame) sólo si dicha estación de trabajo recibe un token con una prioridad menor o igual a la prioridad propia y ésta a su vez "coloca" sus bits de prioridad para indicar la prioridad actual del frame o token. El bit de token tiene el valor de 0 para un token, de 1 para un frame.

El bit monitor es puesto a 0 por una estación de token transmitiendo un token o frame. Los bits de reservación trabajan mano a mano con los bits de prioridad. Una estación de trabajo puede colocar su prioridad en los bits de reservación (si su prioridad es más alta a la prioridad actual de los bits de reservación) con el propósito de poder usar el token.

El campo final del token es el delimitador final (DF). Contiene una combinación única de 1s y símbolos especiales (no de datos). Además de marcar el final del token, contiene 2 subcampos: El bit de frame intermedio y el bit de error detectado. Dichos bits se discutirán posteriormente.

3.5.1 El frame de datos

Un frame consiste de varios grupos de campos: La secuencia de frame inicial (SFI), la dirección destino (DD), la dirección origen (DO), los datos (INFO), la secuencia de verificación de frame (SVF) y la secuencia de frame final (SFF). Juntos, estos campos forman un registro mensaje para el manejo de información en el anillo.

Cada frame comienza con una SFI la cual contiene los siguientes 3 campos:

- DI
- CA
- Control de Frame (CF) que es un byte conteniendo 2 subcampos, tipo de frame e identificador de control:

FF	CCCCCC

Los 2 bits de tipo de frame tienen un valor de 00 para frames MAC y 01 para frames LLC. Los frames LLC contienen datos orientados a aplicaciones tales como mensajes PC a PC y los frames MAC son usados para manejo interno por el software de la red. (Bits 10 y 11 son reservados).

El identificador de control identifica el tipo de manejo de anillo del frame. (*Tabla 3.3*).

La dirección destino (DD) puede ser de 2 o 6 bytes de longitud. Con una longitud de 2 bytes, el primer bit indica si la dirección es una dirección de grupo o una dirección individual. Con direcciones de 6 bytes, el primer bit es también un bit indicador de dirección individual o grupal y el segundo bit expresa si la dirección está asignada local o globalmente.

C	C	C	C	C	C	**Tipos de Frame de Manejo del Anillo**
0	0	0	0	1	1	Token de demanda
0	0	0	0	0	0	Examen de dirección duplicada
0	0	0	1	0	1	Monitor activo presente
0	0	0	1	1	0	Monitor en espera presente
0	0	0	0	1	0	Señal
0	0	0	1	0	0	Eliminar

Tabla 3.3 Bits Identificadores de Control (MAC)

El sobrante de los bits forma la dirección de la estación de trabajo a la cual el frame está direccionado.

La dirección origen (DO) es del mismo formato y tamaño que la dirección destino.

La porción de datos del frame (INFO) puede contener uno de los frames descritos (MAC), o un registro mensaje usado por un protocolo de alto nivel como IPX o NetBios. Este campo no tiene una longitud máxima especificada, aunque se tienen límites prácticos sobre su tamaño basados en los requerimientos de tiempo en los cuales una estación de trabajo puede controlar el anillo.

El campo de secuencia de verificación de frame (SVF) es usado para la detección de error, (CRC).

La secuencia de frame final (SFF) consiste de 2 campos: El delimitador final y el status del frame. En un frame, el DF toma un significado adicional. Además de consistir de un patrón único de impulsos eléctricos, un DF también contiene 2 subcampos de 1 bit. El bit de frame intermedio es puesto a 1 si este frame es parte de una transmisión de frame múltiple, o a 0 para el último o único frame. El bit de error detectado comienza como 0 cuando un frame es originalmente enviado; cada estación de trabajo que pasa el frame, verifica por errores y coloca el bit de error detectado a 1 si encuentra algo mal. Las estaciones de trabajo interviniendo que ven un bit de error detectado a 1, simplemente pasan el frame a través del anillo. La estación de trabajo que originalmente envió el frame, nota que ocurrió un problema y retransmite el frame.

El campo de status del frame (SF) es de 1 byte y contiene 4 bits de reserva (R) y 2 subcampos: El bit de dirección reconocida (A) y el bit de frame copiado (CC).

A C RR	A C RR

Cada uno de los subcampos de 1 bit es duplicado para asegurar la integridad de datos. Cuando se origina un frame, una estación de trabajo transmisora coloca el bit de dirección reconocida a 0; para señalar que ha reconocido su dirección destino, la estación de trabajo receptora coloca el bit a 1. El bit de frame copiado también comienza como 0 pero es puesto a 1 por la estación de trabajo receptora (destino) cuando ésta copia los contenidos del frame en su propia memoria (cuando recibe los datos). Los datos son copiados sólo si el frame se recibió sin error.

Otra situación sucede cuando la dirección destino es reconocida pero el bit de frame copiado no es puesto a 1. Esta le dice a la estación de trabajo transmisora que el frame fue dañado en el camino. Una secuencia de aborto consiste de un delimitador inicial seguido por un delimitador final, y señala que la transmisión actual de un frame o token está siendo cancelada.

3.6 Protocolos de alto nivel

Algún día, tal vez, el modelo OSI será el único protocolo que diferentes fabricantes de redes de área local usen. Hasta entonces, encararemos con un caleidoscopio de protocolos propietarios de IBM (NetBios, APPC), Novell, Inc. (IPX/SPX), DEC, AT&T, Sun (NFS), Apple (Appletalk/AFP), etc. Aún el Departamento de Defensa de los Estados Unidos desarrolló su propio protocolo, TCP/IP, para ser usado sobre grandes redes; TCP/IP es también popular por ligar conjuntamente redes de vendedores diversos.

Los protocolos de IBM y Novell son dos de los más populares del mercado, representan fundamentalmente diferentes alcances en el manejo de comunicaciones en una red de área local. IBM vende cuatro productos de Software primarios que implementan su propio protocolo de LANs. Tres están basados en DOS (PC LAN Support Program, PC LAN Program, y Advanced Program-to-Program Communications [APPC]), y el cuarto (OS/2 LAN Server) es usado bajo OS/2. Novell actualmente ofrece NetWare 286, NetWare 386, y su propio producto APPC.

Protocolos de alto nivel como son: IPX, SPX, NetBios, etc; son directamente programables por software. Los mensajes o datos actuales que circulan a través de la red, provienen ya sea de cierto software de aplicación o del software del redirector que realiza peticiones de servicio de archivos de DOS para el servidor de archivos y viceversa. En este caso se tratará con detalle el protocolo conocido con el nombre de NetBios asi como PC LAN Support Program como el Software de soporte a la LAN.

3.7 Datagramas y Sesiones

Los dos tipos de comunicaciones entre PC y PC son: Datagramas y sesiones.

Un datagrama es un mensaje nunca reconocido por el receptor; si un mensaje enviado debe ser verificado, el receptor debe proporcionar un mensaje de retorno. En otras palabras, el transmisor y el receptor deben estar de acuerdo en un protocolo propio para seguridad del uso de datagramas. En algunos casos, el tamaño máximo de un datagrama es mucho más pequeño que el de un mensaje por sesión. Sin embargo, la mayoría de las redes puede enviar y recibir datagramas más rápido que mensajes por sesión. Cada paquete de información debe incluir información sobre el nodo al que va dirigido, información sobre el número de secuencia para poder reconstruir el mensaje, un reconocimiento entre nodos origen y destino para garantizar la integridad de mensajes.

En contraste a los datagramas, una sesión es una conexión lógica entre dos estaciones de trabajo en la cual la recepción del mensaje está garantizada. Se realiza más trabajo, la sesión debe estar establecida, los mensajes son enviados y recibidos y al final del diálogo la sesión debe ser terminada. Los datagramas llegan en un orden distinto al que fueron enviados, su servicio es similar al servicio postal. Las sesiones son similares al enlace telefónico.

4. Plataforma de Programación Presentada por el NetBios

Capítulo 4. Plataforma de Programación Presentada por el NetBios

4.1 Software de soporte a la red

Además de tener las tarjetas adaptadoras de red propiamente instaladas en varias computadoras, y dichas computadoras conectadas entre sí con el equipo y cableado apropiados, se requiere tener software de ayuda a la red para que las computadoras puedan interactuar entre sí. Por supuesto que dicho software debe ser compatible con el hardware con que se cuente. El software de soporte a la red debe:

- Proveer acceso a la tarjeta adaptadora de red
- Habilitar comunicaciones PC a PC
- Emular DOS de tal manera que los discos del servidor de archivos estén disponibles a las aplicaciones que corran en las estaciones de trabajo.

Comunicaciones PC a PC.- Sin considerar cómo está diseñado un protocolo de un vendedor en particular, se tienen funciones y características comunes:

* Inicio de comunicaciones.- Cada protocolo provee los medios para identificar a una estación de trabajo por un nombre, número, o por ambos. Las comunicaciones punto a punto son activadas por una estación de trabajo identificando a otra estación de trabajo con la cual quiere tener un diálogo. La estación de trabajo origen designa el tipo de diálogo: En un datagrama, los frames son direccionados y enviados a su destino sin garantía de verificación de recepción; en una sesión, se establece una conexión que garantiza la recepción de datos.

* Envío y recepción de datos.- Cada protocolo proporciona los medios para las estaciones para el envío y recepción de datos. Se tiene

un límite en la longitud de mensajes en un protocolo específico, y cada participante en la red tiene los medios para determinar el status de un diálogo (por ejemplo, una estación de trabajo podrá inadvertidamente ser desconectada, tal vez alguien pateó sin intención el cable de conexión a corriente y los demás participantes son notificados de que un error ha ocurrido).

* Término de comunicaciones.- El protocolo provee los medios para que los participantes finalicen el diálogo gratamente.

Emulación y redirección de DOS.- Una aplicación corriendo en una estación de trabajo, pide a DOS alguna parte de un archivo de disco pero el software de la red intercepta la petición y la envía al servidor de archivos. El servidor de archivos hace la operación de I/O de disco para obtener la parte del archivo de disco requerida y la envía a la estación de trabajo. El software de la red en la estación de trabajo entrega el contenido del archivo de disco a la aplicación, y esto hace ver como si una copia local de DOS hubiera obtenido el contenido del archivo.

4.1.1 IBM PC LAN Support Program

Un producto prerequisito para LANs Token Ring basadas en DOS es el PC LAN Support Program; éste implementa el software de soporte de red, así como NetBios, en un conjunto de manejadores de dispositivos (devices drivers) (archivos .SYS) que son cargados cuando el sistema es arrancado. Típicamente se usan tres archivos de este tipo.

Aunque la cantidad de memoria usada depende de como se configuren los manejadores, los rangos de memoria comúnmente usados van de 40 a 50 kb. El manejador DXMA0MOD.SYS dirige las peticiones de comunicación a las diferentes subrutinas en las capas del software de soporte a la red. El archivo DXMC0MOD.SYS es la capa de software de soporte del adaptador

que interactúa con el conjunto de chips de IEEE 802.5 en la tarjeta Token Ring. Y el archivo DXMT0MOD.SYS es NetBios.

NetBios acepta los requerimientos de comunicación de un programa de aplicación o del programa PC LAN (IBM). Dichos requerimientos caen dentro de cuatro categorías:

* Soporte de nombre.- Cada estación en la red es identificada por uno o más nombres. NetBios mantiene esos nombres en una tabla; el primer elemento en la tabla es automáticamente el único y permanentemente nombre asignado del adaptador de red. Pueden ser asignados nombres de usuario a la tabla para identificar convenientemente a cada usuario. Dichos nombres pueden identificar una estación individual o en caso especial, a un grupo de estaciones.

* Soporte de sesión.- Una conexión entre dos nombres (estaciones) en la red puede ser abierta, manejada, y cerrada bajo el control de NetBios. Una estación comienza "escuchando" por una llamada, la otra estación "llama" a la primera. Ambas estaciones pueden enviar y recibir datos durante la sesión. Al final, ambas estaciones "cuelgan" la comunicación.

* Soporte de datagrama.- Los mensajes pueden ser enviados a un nombre, grupo de nombres, o a todos los nombres en la red. No hay una conexión punto a punto establecida y no hay garantía de la recepción del mensaje.

* Status de la sesión y adaptador.- Se tiene información acerca de la tarjeta adaptadora de red local, acerca de otras tarjetas adaptadoras de red y acerca de cualquier sesión activa disponibles de NetBios.

4.2 NetBios

NetBios es un protocolo de IBM usado para la transferencia de mensajes de PC a PC. (Novell proporciona un emulador de NetBios con Netware, usa un formato de frame distinto y es incompatible con el NetBios de IBM). NetBios significa Network Basic Input/Output System. Corresponde a las capas de red, transporte y sesión del modelo OSI. Soporta comunicaciones orientadas a datagramas y sesiones.

Un gran número de aplicaciones usan NetBios para comunicaciones en estaciones de trabajo conectadas a una LAN. Las reglas de NetBios no especifican qué Hardware, protocolos de Software o medios físicos van a ser usados; por tanto, NetBios puede ser implementado en otras plataformas. NetBios fue desarrollado inicialmente para las PC's propias de IBM en redes de PC's IBM, pero ha sido adoptado por muchos vendedores en otras plataformas.

La popularidad de NetBios se debe a varios factores:

- Su interfase aisla la aplicación de los procesos de comunicaciones
- Los comandos para Token Ring son los mismos que para Ethernet o PC Network
- El programa de aplicación sólo trata por sí mismo con la interfase de NetBios y deja a NetBios ejecutar las tareas propias de comunicaciones

La mayoría de los comandos de NetBios vienen en dos formas: Wait o No-wait. Cuando se usa la versión de Wait de un comando, NetBios completa la operación antes de regresar a un programa. Si se especifica la versión No-wait, entonces se tiene la opción de poleo (hacer un ciclo hasta que la operación se complete) o dar a NetBios la dirección de una rutina que NetBios invocará cuando el comando sea completado. Cuando se usa la opción de No-wait, el programa debe inspeccionar dos códigos de retorno distintos para determinar si el comando se ejecutó exitosamente:

El primero es el código de retorno inmediato (disponible tan pronto como NetBios regresa a la aplicación) y el segundo es el código de retorno final (el cual mantiene un valor de X'FF' (FF hexadecimal) hasta que la operación termina, en la cual NetBios coloca el valor apropiado).

4.2.1 Nombres en NetBios

Cada estación de trabajo es identificada por uno o más nombres de 16 bytes de longitud. Se mantiene una tabla con dichos nombres en NetBios. En adición a esta tabla de nombres, siempre está presente un *nombre de nodo permanente* el cual está formado tomando 6 bytes de la dirección en la red de la tarjeta adaptadora y colocando un prefijo de 10 bytes de ceros. Este nombre de nodo permanente es único en la red.

Un programa de aplicación puede inspeccionar los nombres en la tabla de nombres y, excepto para el nombre de nodo permanente, puede añadir y borrar nombres según convenga para la aplicación. También se puede añadir un *nombre de grupo* a la tabla, que no tiene que ser único en la red.

El default es de 16 nombres en una tabla pero esto puede variar, es configurable cuando se carga el manejador de dispositivo. Debido a que los nombres siempre son de 16 bytes de longitud, se debe "rellenar" un nombre corto (tal como Jorge) con espacios antes de adicionarlo a la tabla. No se debe tener un nombre que inicie con * o con X'00' ni es buena idea iniciar con IBM (reservado).

Los comandos de NetBios relacionados al manejo de nombres son **Add Name**, **Add Group Name** y **Delete Name**. El comando **Reset** borra todos los nombres de una tabla (excepto el nombre de nodo permanente). Bajo IBM PC LAN Program no es conveniente usar Reset ya que borrará los nombres usados para la redirección de archivos de DOS así como los nombres adicionados.

Para asegurarse de que un nombre siendo añadido a la tabla sea único, NetBios lo busca en la tabla local y después en toda la red.

4.2.2 Datagramas en NetBios

NetBios permite que los datagramas sean enviados a un nombre individual, a un nombre de grupo o a todas las estaciones en la red. Los datagramas en NetBios pueden tener registros de mensajes de hasta 512 bytes de longitud. Los comandos de NetBios usados para enviar y recibir datagramas son **Send Datagram**, **Send Broadcast Datagram**, **Receive Datagram y Receive Broadcast Datagram**.

4.2.3 Sesiones en NetBios

Se puede crear una sesión entre 2 nombres cualquiera en la red. Mientras la sesión está activa, existe la seguridad de que cuando una estación envíe y reciba registros de mensajes, cada uno será transmitido en secuencia y al destino apropiados. Para crear una sesión, una estación realiza un comando **Listen** que especifica que la estación está "escuchando" por una llamada de un nombre específico o de cualquier otro. Otra estación realiza un comando **Call** especificando al nombre que está llamando.

Cuando cada comando se completa en su respectiva estación, NetBios regresa un *número de sesión local* (LSN) a cada aplicación. Cada estación usa el LSN asignado para referirse a la sesión activa. Los registros de mensajes enviados y recibidos durante una sesión pueden ser de hasta 65535 bytes de longitud. Los comados usados son **Send**, **Chain Send** y **Receive**.

Al final del diálogo, ambas estaciones realizan el comando **Hang Up** para cerrar la sesión.

4.2.4 Bloques de Control de Red (NCBs)

Para invocar un comando particular de NetBios, el programa de aplicación construye un *bloque de control de red* (NCB) y después ejecuta una interrupción 5C (hexadecimal). La *figura 4.1* muestra el formato de un NCB, a continuación se describe cada uno de sus campos:

El campo de *comando* es usado para indicarle a NetBios qué comando se desea ejecutar. Si el bit de más alto orden es puesto a "1", el comando se ejecuta en un modo de No-wait.

El campo de *código de retorno* contiene el código de error inmediato (colocado por NetBios cuando comienza a ejecutar un comando).

Después de que un comando **Listen** o un comando **Call** es ejecutado, el campo de *número de sesión local* contiene el LSN asignado a dicha sesión.

NetBios regresa el campo de *número de nombre* después de que un comando **Add Name o Add Group Name** es ejecutado. Este número es usado con los comandos relacionados a datagramas o con el comando **Receive Any**.

En el campo de *dirección del buffer de datos* se coloca un apuntador al buffer de datos asociado con la operación de envío o recepción (segment:offset 4 bytes).

En el campo de *longitud del buffer de datos* se indica la longitud del mismo.

El campo de *nombre remoto* es usado para indicar el nombre de la estación con la cual se desea comunicar una estación dada.

	Longitud en Bytes
Identificador de Comando	1
Código de Retorno Inmediato	1
Número de Sesión Local	1
Número de Nombre	1
Dirección del Mensaje	4 (apuntador)
Longitud del Mensaje	2
Nombre de Computadora Remota	16
Nombre de Computadora Local	16
Tiempo Límite de Recepción	1
Tiempo Límite de Envío	1
Dirección de Rutina POST	4 (apuntador)
Número de Adaptador	1
Código de Retorno Final	1
Área Reservada	14

Figura 4.1 Formato del Bloque de Control de Red

El campo de *nombre local* sirve para identificar cuál de los nombres en la tabla de nombres local se requiere para ser conocido por la aplicación.

Cuando se realiza un comando **Call o Listen**, se usa el campo de *tiempo límite de recepción* para indicar cuántos intervalos de 0.5 segundos serán usados mientras se espera por un comando **Receive** subsecuente para ser completado. Un valor de 0 indica sin tiempo límite.

En el campo de *dirección de rutina POST* se coloca un apuntador a una rutina que NetBios invoca cuando se completa un comando (este campo es significativo cuando está en efecto la opción No-wait), Si este campo se pone en 0 (nulo), se debe hacer un poleo del código de retorno final para determinar cuándo se completó el comando y si lo hizo exitosamente.

El campo de *número de adaptador* indica cúal adaptador de la red se usa (0 para el primario, 1 para el alterno). Una estación de trabajo puede contener hasta 2 tarjetas Token Ring.

El campo de *código de retorno final* contiene un valor de X'FF' mientras se está procesando un comando; después de que éste es ejecutado, el campo muestra si el comando se ejecutó exitosamente.

El *área reservada* en el NCB no es usada por el programa de aplicación. Está reservada por NetBios.

4.2.5 Frames en NetBios

Los frames de NetBios se crean y manejan exclusivamente por él mismo; no se tiene que tratar con ellos directamente, para crear sus frames, NetBios usa los contenidos del NCB construído más lo que él sabe acerca de los nombres y sesiones actuales en efecto. NetBios usa varios tipos de frames.

Cuando se realiza un comando **Call**, NetBios transmite *name query frame* para averiguar si la estación destino está activa en la red. Cuando se realiza un comando **Add Name**, NetBios envía un *add name query frame* sobre la red para preguntar si alguna otra estación está usando el nombre.

Cuando se realiza un comando **Add Group Name**, NetBios envía un *add group frame name query* para averiguar si otra estación está usando el nombre como nombre único. Una estación que está "escuchando" retorna un *name recognized frame* codificado para indicar una respuesta a un name query frame. En esencia, el frame anuncia "Yo existo, y estoy escuchando".

Si una estación reconoce que uno de sus nombres es el mismo que el de un *add name query frame*, la estación retorna un *add name response frame* para indicarle a la estación de trabajo ejecutante que el nombre ya está en uso. Una estación que realiza un comando **Call** envía un *session initialize frame* para establecer la sesión.

Un *session confirm frame* es regresado a la estación que hace la llamada para indicarle que la sesión está establecida. Un *data frame* es enviado cuando se le da a NetBios un comando **Send** a procesar. Un *ack frame* es usado para indicar recepción exitosa.

NetBios envía un *session end frame* cuando se realiza el comando **Hang Up**. Un *datagram frame* es similar a un *data frame*, excepto que un datagram frame no requiere tener una sesión establecida. NetBios transmite un *status query frame* cuando se le da a NetBios un comando **Adapter Status** por un adaptador remoto. El *status response frame* contiene información de la configuración y el status y es retornado por la estación que recibió un status query frame.

4.3 Programación con NetBios

Una manera útil de comenzar a examinar las técnicas de programación específicas para NetBios es ver qué tan similares son a las usadas en la programación de archivos de disco. La *tabla 4.1* compara operaciones con archivos de PC DOS y las funciones equivalentes en NetBios.

Operación de archivos	Equivalente en NetBios
Abrir	Add Name, entonces Call (o Listen)
Leer	Receive
Escribir	Send
Localizar	Ninguno
Cerrar	Hang Up, entonces Delete Name

Tabla 4.1 Comparación de I/O de Archivos con Funciones de NetBios

Usando datagramas en NetBios, se pueden enviar y recibir paquetes de datos como se muestra en la *tabla 4.2*.

Estación de trabajo A	Estación de trabajo B
1. Adicionar nombre de A	1. Adicionar nombre de B
2. Enviar mensaje a B	2. Recibir un mensaje
3. Recibir un mensaje	3. Enviar mensaje a A
4. Borrar nombre de A	4 Borrar nombre de B

Tabla 4.2 Secuencia de un Paquete de Datos para Datagramas en NetBios

Los comandos de NetBios Call, Listen, y Hang Up son las herramientas para el manejo de sesiones. La *tabla 4.3* muestra la secuencia de eventos que se usarían para abrir una sesión, intercambiar mensajes, y cerrar la sesión.

Estación de trabajo A	Estación de trabajo B
1. Adicionar nombre de A	1. Adicionar nombre de B
2. Escuchar	2. Llamar a la estación A
3. Enviar mensaje a B	3. Recibir un mensaje
4. Recibir un mensaje	4. Enviar mensaje a A
5. Colgar sobre B	5. Colgar sobre A
6. Borrar nombre de A	6. Borrar nombre de B

Tabla 4.3 Secuencia de un Paquete de Datos para Sesiones en NetBios

4.3 1 Invocación al NetBios

Para invocar alguna función particular de NetBios, se crea un NCB y se utiliza un apuntador al NCB en el par de registros ES:BX ejecutándose una interrupción 5C (Hexadecimal).

Los elementos de programación aquí presentados se realizaron utilizando el lenguaje de programación PASCAL (TURBO PASCAL 5.5 y 6.0).

```
-----------------------------------------------------------
PROCEDURE int_netbios(var ncb : netcb);
var
   regs : registers;
begin
   regs.ES := seg(ncb);
   regs.BX := ofs(ncb);
   intr($5C,regs);
END;
-----------------------------------------------------------
```

4.3.2 Estructura del NCB

Todo lo que se especifica cuando se invoca la interrupción 5C es la dirección de un bloque de control de red. El NCB le indica a NetBios todo lo que éste necesita saber acerca de una operación particular; el NCB no es transmitido a través de la red, NetBios lo usa como un conjunto de direcciones que especifican lo que se requiere hacer. Para cada operación se declara un NCB por separado, es más fácil manejar uno o dos NCBs para múltiples propósitos que si se usa uno para cada operación.

Veamos a continuación la capa de un NCB:

```
------------------------------------------------------------
type
    byte14 = array[1..14] of byte;
    byte16 = array[1..16] of byte;

netcb = record
    comando : byte;         {* identificador de comando *}
    codreti : byte;         {* código de retorno inmediato *}
            : byte;         {* número de sesión local *}
    numnom  : byte;         {* número de nombre *}
    dirmen  : pointer;      {* apuntador al mensaje *}
    lonmen  : word;         {* longitud del mensaje *}
    nomcrem : byte16;       {* nombre de computadora remota *}
    nomcloc : byte16;       {* nombre de computadora local *}
    tiempor : byte;         {* tiempo límite de recepción en incrementos
                             de 0.5 segundos *}
    tiempoe : byte;          {* tiempo límite de envío en incrementos de
                             0.5 segundos *}
    dirpos  : pointer;       {* apuntador a rutina POST *}
    numadap : byte;          {* número de adaptador (0 o 1) *}
    codretf : byte;          {* código de retorno final *}
    reserva : byte14;        {* área reservada *}
```

```
end;
-----------------------------------------------------------------
```

4.3.3 Comandos del NetBios

Los comandos del NetBios pueden ser usados en dos formas: Wait y No-wait. Ambas formas ya han sido presentadas en este documento. A continuación se presenta una lista de los comandos usados en NetBios con su respectivo identificador de comando.

Comando	**Identificador de comando (Hexadecimal)**
Reset Adapter	32(wait)
Cancel	35(wait)
Get Adapter Status	33(wait), B3(no-wait)
Unlink	70(wait)
Add Name	30(wait), B0(no-wait)
Add Group Name	36(wait), B6(no-wait)
Delete Name	31(wait), B1(no-wait)
Call	10(wait), 90(no-wait)
Listen	11(wait), 91(no-wait)
Hang Up	12(wait), 92(no-wait)
Send	14(wait), 94(no-wait)
Send No Ack	71(wait), F1(no-wait)
Chain Send	17(wait), 97(no-wait)
Chain Send No Ack	72(wait), F2(no-wait)
Receive	15(wait), 95(no-wait)
Receive Any	16(wait), 96(no-wait)
Session Status	34(wait), B4(no-wait)
Send Datagram	20(wait), A0(no-wait)
Send Broadcast Datagram	22(wait), A2(no-wait)
Receive Datagram	21(wait), A1(no-wait)
Receive Broadcast Datagram	23(wait), A3(no-wait)

ESTA TESIS NO DEBE
SALIR DE LA BIBLIOTECA

5. Implementación del Sistema de Correo Electrónico

ESTA TESIS NO DEBE
SALIR DE LA BIBLIOTECA

5. IMPLEMENTACIÓN DEL SISTEMA DE CORREO ELECTRÓNICO

Capítulo 5. Implementación del Sistema de Correo Electrónico

5.1 Modelado del sistema de correo electrónico (introducción)

Toca en turno presentar la implementación del sistema de correo electrónico mencionado en este documento, una manera adecuada de mostrar sus características funcionales y de conformación, es a través de un modelo o prototipo.

El modelo a tratar, será presentado considerando su desarrollo conceptual, su estructura interna y su estructura funcional; elementos clave en su creación.

5.1.1 Modelo conceptual

La *figura 5.1* representa adecuadamente la funcionalidad general de cada CBMS en el mercado, aunque no hay dos sistemas que tengan el mismo conjunto de características o las implemente de la misma manera.

No usan la misma nomenclatura, de hecho, habrá numerosos sinónimos usados en sistemas diversos para la identificación de los comandos. Las únicas partes de un CBMS que son comunes a cada sistema son el **buzón de entrada**, el cual muestra un listado de correo que puede ser leído, y algún **ambiente de entrega de mensajes**, el cual controla la transferencia de los mismos entre usuarios.

Muchos sistemas, por ejemplo, no tienen buzón de salida o aún sistemas de almacén o archivado. Otros más no tienen un editor, el cual es operado en el espacio de trabajo del usuario. Algunos sistemas, por otra parte, tienen todas las estructuras mencionadas, incluyendo un

ambiente que muestra el correo que ya ha sido leído, correo que ha sido enviado, espacio de trabajo para crear y editar mensajes, etc.

Otros más, permiten al usuario crear listas de direcciones de múltiples receptores cada vez que un mensaje es enviado a un grupo específico, (en nuestro caso, esto es controlado con el propio NetBios en la tabla de nombres que maneja), crear formas de negocio electrónicas, boletines, etc.

La *figura 5.2* representa el sistema de correo electrónico tratado en este documento. Se pueden apreciar las estructuras que lo conforman.

Alterno a lo mencionado anteriormente, se tiene un elemento importante en un CBMS como lo es la **interfase de usuario**; cuando se accesa a un sistema de correo, el usuario interactúa con éste a través de la interfase de usuario. Dichas interfases están diseñadas de tal forma que los usuarios puedan ejecutar la mayoría de sus operaciones "fuera de línea" en la estación de trabajo. Estas interfases tienen rutinas especiales que envían y recuperan mensajes automáticamente, cuelgan la línea, y entonces presentan los mensajes nuevos a sus usuarios.

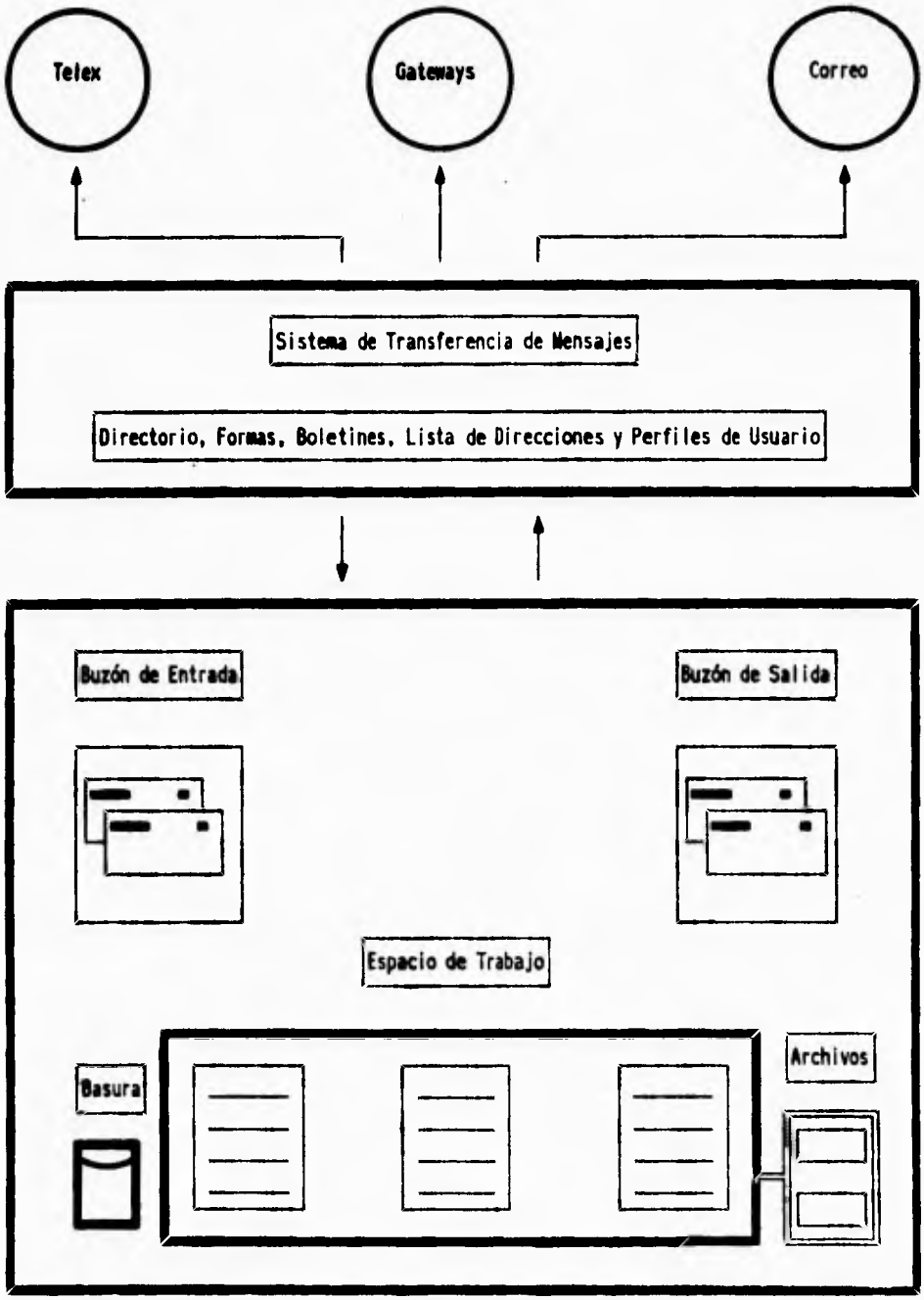


Figura 5.1 Modelo de un CBMS Típico

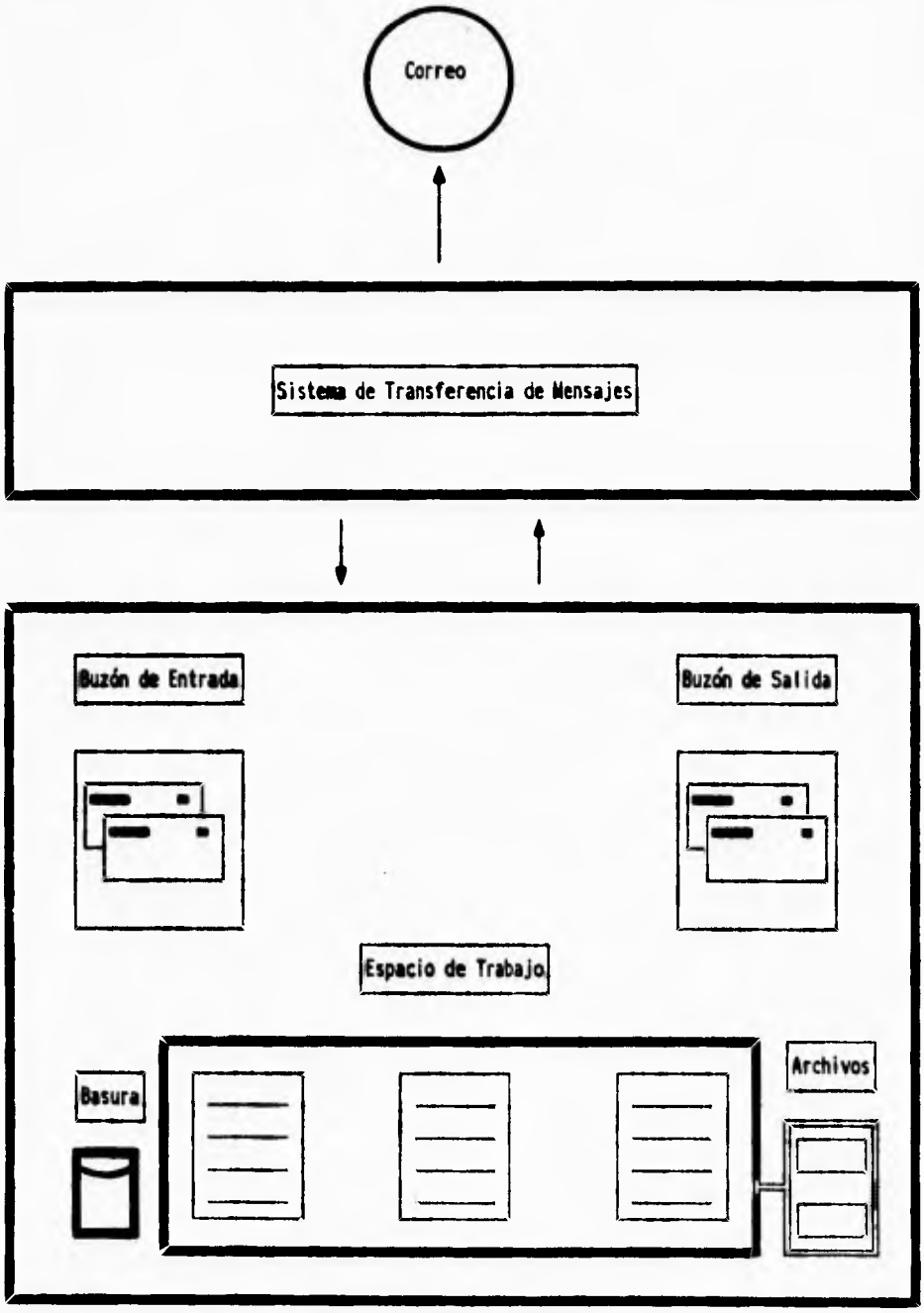


Figura 5.2 Modelo del CBMS Presentado en Documento

5.1.2 Estructura interna del sistema

A continuación se presenta en la figura 5.3, la estructura interna del sistema de correo electrónico, están considerados los siguientes elementos que serán descritos a continuación:

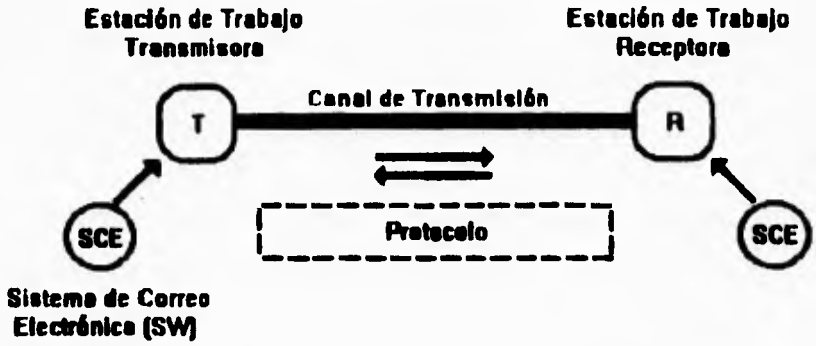


Figura 5.3 Estructura Interna del Sistema de Correo Electrónico

Canal de transmisión.- El tipo de canal utilizado es el conocido como Half Duplex o Semiduplex, en el cual las señales son transmitidas o recibidas en una sola dirección a la vez, ver *figura 5.4*. Para nuestro caso, el tipo de cableado es indiferente como nuestro medio de transmisión, siempre y cuando, dicho tipo de cableado sea un estándar soportado. En este caso se usó cable coaxial como medio de transmisión.

Estaciones de trabajo.- Para este sistema se utilizaron computadoras PCs tipo AT, debemos recordar que se tienen diversos tipos de tarjetas adaptadoras de red con ciertas consideraciones de HW y SW de acuerdo al tipo de estación que se esté usando. Las características del tipo de tarjeta usada en este caso pueden ser vistas en el apéndice C. Sabemos que el desempeño del SW depende del HW en el cual esté trabajando.

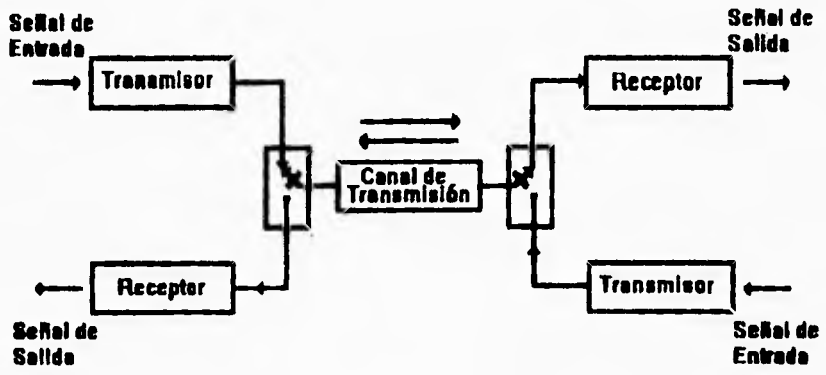


Figura 5.4 Tipo de Canal Semidupiex

Sistema de correo electrónico.- Dicho sistema, constituído por un conjunto de programas escritos en lenguaje de alto nivel, estará presente en cada estación de trabajo involucrada en la red, será descrito con detalle posteriormente en este capítulo.

Protocolo.- Como ya fue mencionado con anterioridad, un protocolo es una serie de reglas que rigen el intercambio de información entre capas, del mismo nivel, contenidas en computadoras distintas. En este caso, el protocolo tratado no puede ser considerado un "protocolo estándar", ya que fue definido tomando en consideración elementos funcionales del protocolo estándar conocido como **Sliding Window (ventana deslizante)**, el cual considera dos estructuras de datos, (ventana de transmisión y ventana de recepción), las cuales poseen identificadores de secuencia de los paquetes de información que han sido transmitidos/recibidos. En dicho protocolo se tiene desinterés en el orden en que se envían o reciben los paquetes de información, no así en nuestro protocolo, ya que el orden de envío y recepción si es importante (desventaja), maneja de manera distinta las estructuras de datos mencionadas.

Por otra parte, por cada paquete de información recibido, se envía un paquete de reconocimiento de recepción (acknoledgement). Existen algunos protocolos estándar utilizados para llevar a cabo el control de

seguridad en el envío y la recepción de cada paquete de información. Por ejemplo, el protocolo **GO BACK N**, en el cual cada paquete enviado requiere de un reconocimiento de recibido, si dicho reconocimiento no llega, la secuencia de envío de paquetes es repetida a partir del paquete de envío cuyo reconocimiento no fue recibido, (ver *figura 5.5)*.

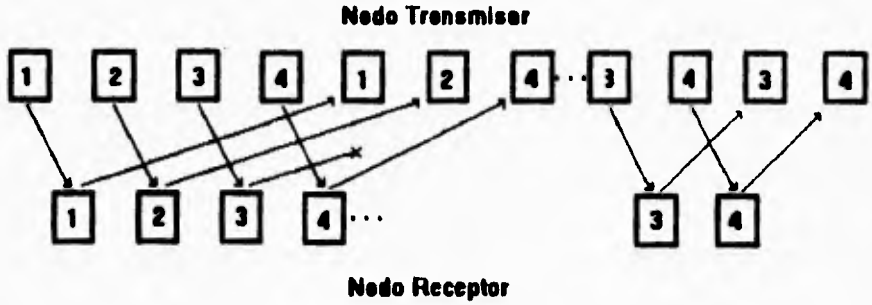


Figura 5.5 Protocolo GO BACK N

El protocolo **SELECT FRAME**, únicamente repetirá el envío del paquete de información cuyo reconocimiento no fue recibido, continuando posteriormente con la secuencia de envío de paquetes ya iniciada, (ver *figura 5.6)*.

Para nuestro caso, en caso de error de reconocimiento no recibido, se tendrá que volver a transmitir nuestra serie de paquetes de información deseada.

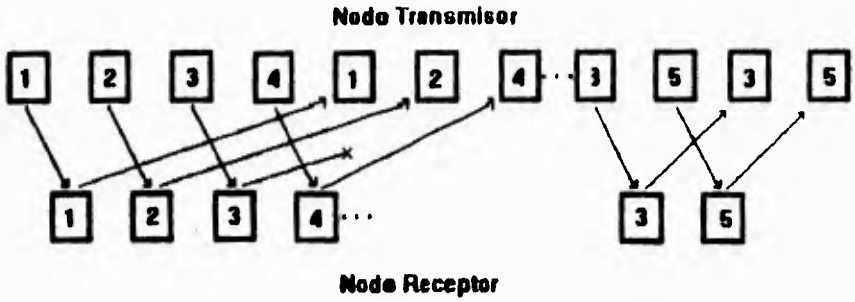


Figura 5.6 Protocolo SELECT FRAME

Dadas las características de funcionalidad del NetBios usando datagramas (nuestro caso), el tamaño máximo del paquete de información que puede ser transmitido en un tiempo dado es de 512 bytes, considerando además que en esta implementación se utilizan algunos bytes para indicar el tipo de paquete que se está manejando, el identificador de secuencia del mismo, así como su longitud; el tamaño final del mensaje posible a enviar es de 500 bytes. A este paquete de información lo llamamos datagrama, siendo una de las estructuras de datos aquí manejada, las cuales serán descritas en su turno.

Considerando, por ejemplo, una nota que se desea transmitir la cual contine 4000 bytes de datos, se debe dividir dicha nota en bloques de 500 bytes y transmitir cada bloque por separado, será responsabilidad de la estación receptora volver a armar la nota adecuadamente para que la recepción sea correcta, auxiliándose de los datos que la estación transmisora está enviando primeramente para dar información clave al receptor, esto es, quién envia la nota, tópico a tratar, número de bloques de la nota, tamaño total.
La estación transmisora es la encargada de dividir la nota a enviar en los bloques correspondientes.

Una manera de presentar lo descrito anteriormente, es auxiliándose del siguiente esquema de envío/recepción de una nota. Nótese la secuencia que rige a dicho proceso, *(figura 5.7)*.

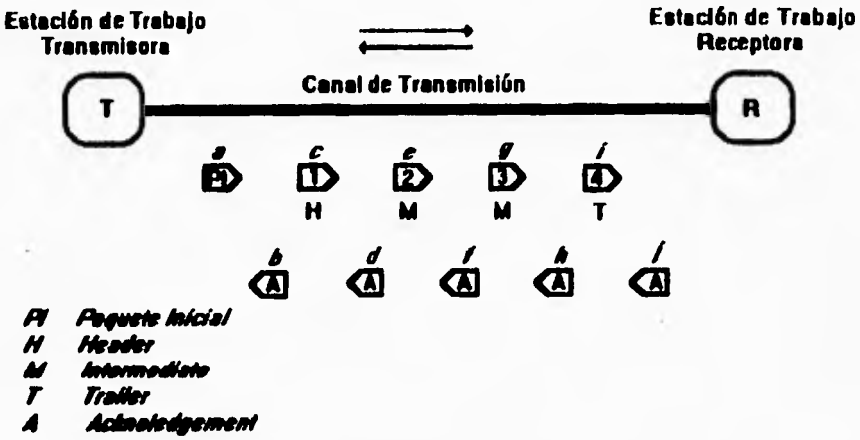


Figura 5.7 Envío/Recepción (Secuencia de Ejemplo)

Para cada paquete de información enviado se tiene un reconocimiento de recepción, el cual asegura una comunicación efectiva. Lo anterior se lleva a cabo siempre y cuando se cumpla el proceso completamente, de lo contrario, se tiene que repetir la secuencia de envío/recepción de la nota en turno.

A continuación se describirán las estructuras de datos que conforman este sistema en su operación, dicho sistema fue desarrollado utilizando el lenguaje de programación PASCAL (TURBO PASCAL 5.5 y 6.0):

Datagrama .- Está formado por 4 campos y representa la unidad básica de envío/recepción de información.

- **Tipo**: Cada datagrama puede ser de tipo A (Acknoledgment), H (Header), M (Intermediate), T (Trailer); su clasificación va en función a su certificación de recibido o bien a su posición dentro del grupo de paquetes de texto del mensaje a transmitir/recibir.
- **Secuencia**: Representa el orden de envío/recepción de cada paquete de texto dentro del mensaje, (0, 1, 2, 3, ...).

- **Longitud**: Determina el tamaño del paquete que se está manejando en un momento dado, su valor queda comprendido entre 1 y 500. (Número de elementos).
- **Texto**: Es el arreglo de caracteres transmitidos/recibidos, va de 1 a 500 caracteres.

A nivel de programación, queda representado de la siguiente forma:

```
arr500 = array[1..500] of char;
Datagrama = record
                tipo      : char;
                secuencia : byte;
                longitud  : integer;
                texto     : arr500;
            end;
```

Paquete de inicio (informativo).- Representa al elemento primeramente enviado antes de iniciar una transmisión de un mensaje. Su función es proporcionar información básica y muy importante para el(la) envio/recepción del mensaje. Esta formado por los siguientes 5 campos:

- **Origen**: Indica el "nombre" del transmisor (a nivel tabla de nombres).
- **Destino**: "Nombre" del receptor (a nivel tabla de nombres).
- **Titulo**: Tópico a tratar en el mensaje, hasta de 35 caracteres.
- **Número de partes**: Número de paquetes a transmitir/recibir que conforman el mensaje, cada paquete puede tener hasta 500 bytes (caracteres) como ya fue mencionado.
- **Número de caracteres**: Número de caracteres totales a enviar/recibir en el mensaje, el limite es de 8000 caracteres, dicho valor fue el deseado en este caso pero puede ser modificado sin presentar mayor problema.

Queda representado de la siguiente forma:

```
cad16 = string[16];
Paquete_de_inicio = record
                        origen    : cad16;
                        dest      : cad16;
                        titulo    : string[35];
                        no_partes : byte;
                        no_caract : integer;
                    end;
```

Registro de datos.- Conforma la unidad de almacenamiento de mensajes, es decir, cada mensaje al estar almacenado, queda representado por un registro de datos. Sus 7 campos son los siguientes:

- **Localidad**: Indica si el mensaje se encuentra ubicado en el buzón de entrada o de salida (I, O).
- **Fecha/hora de envio/recepción**: Como su nombre lo indica, es el dato de tiempo real en el cual se lleva a cabo el (la) envio/recepción del mensaje.
- **Titulo**: Tópico a tratar en el mensaje.
- **Número de caracteres**: Número de caracteres totales a enviar/recibir en el mensaje.
- **Texto**: Texto del mensaje en sí.

A nivel de programación queda representado de la siguiente manera:

```
arrtot = array[1..8000] of char;
reg_data = record;
            loc          : char;
            fecha_unpack : DateTime;
            envia        : cad16;
            recibe       : cad16;
            asunto       : string[35];
            longtot      : integer;
            textotot     : arrtot;
         end;
```

Registro de buzón.- Es el elemento fundamental para llevar a cabo la selección de mensajes en los buzones de entrada y salida. Consta de los siguientes 5 campos:

- **Nombre**: Nombre del archivo (mensaje); (I#######.##M para entrada y O#######.##M para salida). #######.## es tomado de la serie de números obtenida del valor de la fecha y hora empacados, M indica Mensaje.
- **Título**: Tópico a tratar en el mensaje.
- **Origen**: Nombre del transmisor.
- **Destino**: Nombre del receptor.
- **Fecha/hora de envio/recepción**: Dato de tiempo real en el cual se lleva a cabo el (la) envio/recepción del mensaje.

Queda representado de la siguiente forma:

```
Reg_buz = record;
            nombre       : string [12];
            asunto       : string[35];
            envia        : cad16;
            recibe       : cad16;
            fecha_unpack : DateTime;
        end;
```

5.1.3 Descripción conceptual de subsistemas, estructura funcional

A continuación trataremos con más detalle uno de los elementos mencionados en nuestro modelo, el propio Sistema de Correo Electrónico. Aunque un CBMS podría ser visto por los usuarios como un modelo de su sistema de servicio postal, para la computadora no es más que un conjunto de archivos controlados por un programa. El usuario debe dirigir el sistema para ejecutar comandos sobre esos archivos. Un mensaje por sí mismo, consiste de una serie de campos como un registro de una base de datos. La *figura 5.8* muestra los campos contenidos en un mensaje para el sistema tratado en el presente documento.

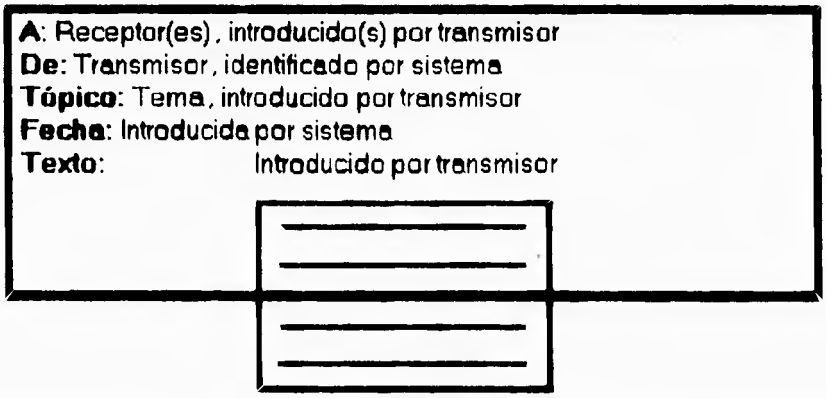


Figura 5.8 Campos dentro del CBMS

Control de mensajes.- Para ahorrar espacio dentro de la computadora, los mensajes son típicamente almacenados sólo una vez, así, cada mensaje es un archivo de computadora individual o un registro dentro de un archivo enorme identificado por un número de mensaje único. Los identificadores o apuntadores, son almacenados en otros archivos que indican a los usuarios quién tiene acceso a los mensajes. De esta manera, el usuario se traslada primero a un "contenedor" específico, como un buzón de entrada, salida, o un archivo personal el cual es creado por el usuario con un nombre particular, y entonces realiza una acción por medio de un comando. Cuando un mensaje es enviado a receptores definidos, por ejemplo, la computadora almacena el mensaje en un archivo y actualiza los archivos de entrada de cada usuario que tenga acceso al mensaje. Dichos comandos le indican a la computadora ejecutar acciones relacionadas a mensajes específicos, por ejemplo, mostrar el contenido del archivo señalado en el buzón de entrada (comando de lectura) o eliminar dicha referencia del espacio de trabajo o de un archivo personal (comando de borrado).

Estructura de directorio.- La estructura de directorio es tan importante como el control de mensajes en un CBMS; el sistema no puede ser controlado dentro de una organización si no cuenta con una buena estructura de directorio para la ubicación y control de acceso de mensajes. Para este sistema, la unidad fundamental de almacenamiento de información (notas) en directorio, es una estructura de datos mencionada, un *registro de datos*. Cada archivo representando una nota, quedará almacenado en un directorio (CORREO) para cada estación de trabajo, identificado como ya fue mencionado en la estructura de datos *registro de buzón*.

Integridad y seguridad de mensajes.- Debido a que la computadora sólo "ve" a los mensajes como archivos, no se tiene integridad y seguridad de los mensajes en forma inherente; deben ser proporcionados como parte del programa CBMS. El factor de seguridad más obvio es que los mensajes pueden ser leídos únicamente por aquellos receptores a quienes van dirigidos y esto debe tener suma importancia en el desarrollo de un CBMS. Por otra parte, la mayoría de estos sistemas han

adoptado la convención de no permitir que los mensajes introducidos al sistema sean alterados, aunque existen otros más que permiten la edición de los mismos. Si un mensaje puede ser cambiado después que ha sido recibido, no hay camino para garantizar que lo que se tiene almacenado no ha sido alterado. Para evitar la oportunidad de revisión pero no prohibiendo la conexión con la preparación de documentos, los mejores CBMS permiten que el texto de mensajes sea colocado dentro del editor pero requieren que sean enviados como mensajes nuevos, garantizando la integridad de los mismos. Esta es la manera en que trabaja el sistema presentado.

Para cualquier usuario, el punto más importante de un CBMS es su funcionalidad; un CBMS consiste de un número de subsistemas que ejecutan ciertas funciones requeridas para emular la forma en que los trabajdores del conocimiento operan en sus escritorios y la forma en que un sistema de servicio postal opera.

Dichos subsistemas son **lectura**, **escritura**, **envío/recepción y administración**, mostrados en la *figura 5.9*.

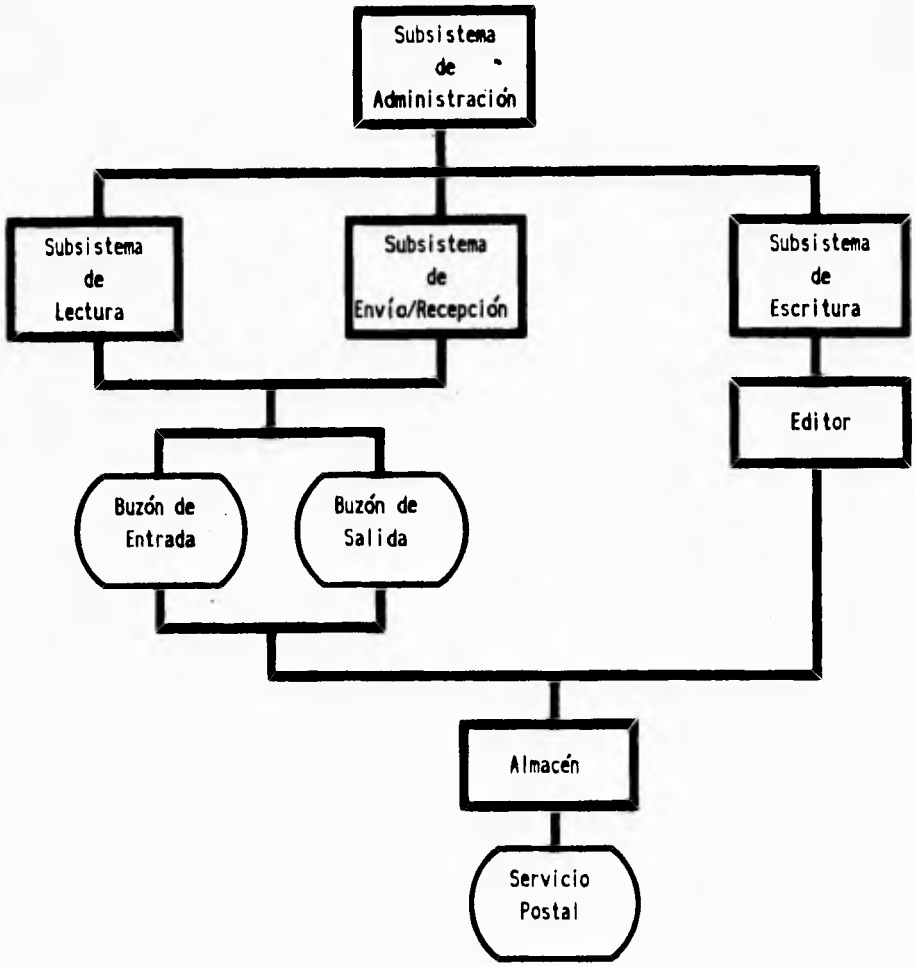


Figura 5.9 Subsistemas Funcionales del CBMS

El subsistema de lectura maneja todas las funciones asociadas con la lectura y control de mensajes, incluyendo, archivado, impresión y borrado de los mismos. El subsistema de escritura controla la creación de mensajes y llama al editor del sistema cuando es requerido; es interfase entre el subsistema de lectura en un lado y el subsistema de envío/recepción por el otro lado. El subsistema de envío/recepción controla la transferencia y recuperación de mensajes. El subsistema de administración maneja el registro de usuarios y otros recursos centrales; controla el sistema de correo y debe ser considerado el subsistema guía.

Subsistema de lectura.- Tiene como elementos principales a los buzones de entrada y salida así como un control de visualización, impresión, almacén y borrado de archivos(mensajes).

Los mensajes incluídos en los buzones mencionados, pueden ser "rastreados" y leídos típicamente haciendo una selección deseada, cada mensaje leído es mostrado en la pantalla de la computadora o bien impreso en papel si así se desea.

Dichos mensajes incluyen información relativa al transmisor, fecha y hora de envío/recepción, tópico particular y el texto del mensaje. Por otra parte, si se desea, el mensaje puede ser eliminado del buzón correspondiente.

Subsistema de escritura.- Este subsistema está asociado con la creación de mensajes. Su elemento principal es el editor, el cual permite la preparación de mensajes que serán enviados posteriormente.

El sistema tratado en este documento cuenta con un editor los suficientemente funcional para la realización de notas incluyendo algunas funciones de búsqueda y reemplazo de palabras, copia y movimiento de bloques de texto, inserción y borrado de líneas, etc. Es un editor programado utilizando PASCAL orientado a objetos.

Subsistema de envío/recepción.- Permite al sistema la transmisión y recepción de mensajes entre las partes deseadas. Para este subsistema también son importantes dos elementos ya mencionados con anterioridad, los buzones de entrada y salida, en ellos tenemos una relación de los mensajes enviados y recibidos.

Cabe mencionar que debe existir una sincronización entre las estaciones interviniendo en una transmisión/recepción de mensajes, y esto lo controla este subsistema.

Subsistema de administración.- Es el encargado de llevar a cabo el control general del sistema de tal manera que su funcionamiento sea el adecuado. Lleva a cabo la correspondiente inicialización de la estación de trabajo para que sea reconocida en la red. Rige la secuencia de los procesos involucrados en el sistema.

5.1.4 Entorno de trabajo para la implementación

Problemas principales encontrados durante la implementación.- Básicamente los principales problemas que se presentaron en el período de creación del sistema, relativos a la programación, fueron los siguientes: Primero, comprender y hacer efectivo el funcionamiento de algunos comandos del NetBios, el vencer este obstáculo significaba dar un gran paso en el desarrollo del sistema al comprender el método de programación de dicho protocolo; segundo, la sincronización en tiempo del transmisor/receptor para el buen funcionamiento del correo electrónico, el sistema, por medio de la interfase de usuario, está "listo" para realizar actividades de sensado de correo internamente "todo el tiempo" mientras esté activo.
Definitivamente no fueron los únicos obstáculos a vencer pero quizá si los más relevantes.

Ventajas / desventajas de utilizar NetBios.- Ya han sido mencionadas las características principales de este protocolo tan conocido y usado en la

actualidad, en el capítulo 4 de este documento así como en los apéndices A y B del mismo. Sólo restaría decir que es un protocolo muy recomendable, dadas sus características, en la programación de aplicaciones para redes de área local.

Ambiente de pruebas utilizado.- El ambiente de pruebas utilizado durante la creación de este sistema consistió de los siguientes elementos de Hardware y Software:

HW	SW
2 Estaciones de trabajo (computadoras AT)	S.O. DOS >= V3.3
2 Tarjetas adaptadoras Token Ring 16/4	PC LAN Support Program V 1.02
1 MSAU	(NetBios)
Cables coaxiales y conectores	Turbo Pascal V5 y V6

Con los elementos anteriores, pertenecientes al IIMAS (Instituto de Investigaciones en Matemáticas Aplicadas y en Sistemas), fue posible la realización del sistema aquí tratado; cabe mencionar que en un principio, no se tenía el material completo y fue necesaria la utilización de la red Token Ring ahí existente para la realización de algunas pruebas.

5.2 Transmisión de voz

Este tópico representa uno de los elementos de interés actuales en el medio de la computación y específicamente en las redes de computadoras; la codificación de voz se define como una representación digital de los sonidos de voz para proporcionar almacenamiento eficiente, transmisión, recuperación y reconstrucción perceptualmente libre de fallas de la voz o señal original.

El sistema al que hace referencia este documento, no presenta esta facilidad dada una serie de características que debe cumplir un sistema de dicha índole. La estructura lógica de este sistema no es compatible con la estructura lógica que debiera tener un sistema que maneje transmisión de voz, primeramente, este sistema no involucra transmisión/recepción de información en tiempo real, que es la forma en que se debe trabajar con la transmisión/recepción de voz dado el tamaño de archivos que pudieran surgir considerando una transmisión de voz de algunos segundos (muy grandes), por otra parte, el control que debiera tener este sistema para manejo de voz sería totalmente distinto al que presenta actualmente, aunque bien pudiera ser un elemento de interés la implementación de esta facilidad en un futuro, sería realmente interesante. Se requiere además de HW y SW adicionales

Existe bibliografía que bien puede ser consultada para obtener información detallada de lo que representa la codificación de voz en tiempo real para canales de comunicación digital, y con esto comprender el extenso e interesante camino que implica la implementación de esta facilidad.

6. RESULTADOS Y CONCLUSIONES

Capítulo 6. Resultados y Conclusiones

6.1 Evaluación del desempeño y utilidad del sistema

Los sistemas de mensajes basados en computadora frecuentemente parecen simples vistos desde fuera, pero presentan complejidad cuando son estudiados a detalle. Los sistemas disponibles en el mercado parecen muy similares, especialmente cuando el usuario únicamente compara sus caracerísticas a manera de una lista de Sí y No. En la práctica, sin embargo, son bastante diferentes en la forma en que implementan sus funciones.

El sistema de correo electrónico aquí presentado representa sólo una de las tantas aplicaciones que hoy en día son utilizadas en las redes de computadoras; en el mercado, existen algunos sistemas operativos para control de LANs, que además de proporcionar un servicio de correo electrónico, presentan también facilidades para compartir recursos (discos, impresoras, archivos, etc), un control de estadísticas de la red, y mucho más (IBM PC LAN Program, Novell/Netware, etc).

Definitivamente el sistema aquí presentado no compite con las aplicaciones del mercado, pero bien puede ser considerado un sistema práctico y funcional para un ambiente de oficinas (pequeño), por ejemplo, sin ser un elemento de programación excepcional.

Fue desarrollado considerando una red Token Ring existente en el Instituto de Investigaciones en Matemáticas Aplicadas y en Sistemas (IIMAS, UNAM), las ventajas y desventajas de este tipo de redes fueron tratadas en el capítulo 3 de este documento, se requirió instalar el HW Y SW respectivo en las PCs utilizadas durante el desarrollo del sistema; por otra parte, se utilizó como protocolo de comunicaciones el NetBios, protocolo de mucha popularidad debido a sus características: Aisla la aplicación del proceso de comunicaciones, se tienen comandos iguales

para redes Token Ring o Ethernet, el programa de aplicación sólo interactúa con la interfase de NetBios y deja a NetBios ejecutar las tareas de comunicaciones, etc.

El haber desarrollado una aplicación como la aquí presentada fue de sumo interés personal, se tiene la oportunidad de idear, en base a estándares establecidos, un protocolo propio, que si bien no mejora quizá a algunos de los aquí mencionados, sí da la pauta a la inventiva personal. Este sistema puede servir como base para desarrollar alguna aplicación deseada considerando NetBios o bien mejorarlo, lo siguiente entonces es emprender el camino hacia la meta. Se cumple con el objetivo planteado.

El sistema creado consiste de dos archivos ejecutables en su funcionamiento: E_MAIL.EXE (sistema de correo electrónico con un tamaño de 47168 Bytes) y EDITOR1.EXE (editor de texto con un tamaño de 93952 bytes). Los programas fuentes: E_MAIL.PAS y EDITOR1.PAS tienen un tamaño de 46405 y 8781 bytes respectivamente.

Una copia del código fuente del CBMS y una copia del código ejecutable del mismo, serán entregadas junto con el presente documento al IIMAS, así como a la División de Ingeniería Eléctrica, Electrónica y en Computación (D.I.E.E.C) de la Facultad de Ingeniería.

6.2 Mejoras futuras al sistema (recomendaciones)

Este tópico lo trataremos como una lista de consideraciones que serán descritas a continuación:

- Mejoras en el protocolo de comunicación implementado, posible utilización de GO BACK N o SELECT FRAME, junto con Sliding Window.

- Creación de un editor propio del sistema, quizá más simple y con esto obtener reducción del tamaño del código del programa.

- Modificación del código del CBMS para hacerlo residente en memoria.

- Mejoras en el control de directorio de mensajes, control de archivos de envio/recepción.

- Implementación de un control de estadisticas de la red.

- Lo que dicte su inventiva.

BIBLIOGRAFÍA

Aspects of Data Communication
John McNamara
Digital Press
1986

Computer Networks
Andrew S. Tanenbaum
Prentice Hall
1988

Data Communications Computer Networks and OSI
Fred Halsall
Addison-Wesley Publishing Company
1988

Dictionary of Computing
IBM
1987

Enciclopedia de Términos de Microcomputación
Linda Gail Christie / John Cristie
Prentice-Hall-Hispanoamericana, S.A.
1986

E-Mail
Stephen A. Caswell
Artech House
Boston-London
1988

IBM Networking and Connectivity Technologies
CTR Computer Technology Research Corp.
1990

Local Area Network Concepts and Products
International Technical Support Organization
Raleigh Center
1994

Local Area Network Support Program V1.02
DXMINFO.DOC FILE
IBM

Network Programming in C
Barry Nance
QUE Corporation
Programming Series
1990

Programmer's Guide to NetBios
W. David Schaderer
Howard W. Sams & Company
1988

Token-Ring Network PC Adapter Technical Reference
IBM

Turbo C The Complete Reference
Herbert Schildt
1988

Turbo Pascal 5
Stephen K. O'brien
Borland Osborne / Mc Graw Hill
1988

Turbo Pascal 6 The Complete Reference
Stephen K. O'brien
Borland Osborne / Mc Graw Hill
1991

Referencias de herramientas de cómputo utilizadas para la elaboración de este trabajo, sistema operativo DOS, procesador de palabras WORD FOR WINDOWS, elaboracion de algunos gráficos FLOW, etc.

APÉNDICES

A. Funciones del NetBios usadas en este Sistema

Reset Adapter ***X'32' (wait)***

Descripción: Este comando da un reset a NetBios, borrando y limpiando toda la información de la sesión, borrando nombres (excepto el nombre de nodo permanente), y dando un reset al número máximo de sesiones y número máximo de NCBs potenciales restantes. Se pueden proporcionar nuevos valores para esos máximos como parte de este comando.

Si se desea alterar los números máximos de sesiones y NCBs restantes con respecto a los valores de omisión o de default, se requiere codificar los nuevos valores en los campos *nsl y numnom* del NCB antes de ejecutar el comando Reset. Si dichos campos contienen un valor de 0 cuando se ejecuta este comando, se tienen valores de default de 6 sesiones y 12 NCBs. Para mejor desempeño se recomienda tener estos valores lo más pequeños posibles.

Si una estación de trabajo tiene 2 adaptadores, este comando da un reset a NetBios de tal manera que sólo afecta al adaptador especificado en el campo *numadap* del NCB.

Cuidado: No usar este comando cuando se está corriendo bajo el programa IBM PC LAN. Este limpia las sesiones PCLP y borra los nombres PCLP, teniendo como consecuencia errores en la red.

Campos de entrada del NCB:

comando	X'32'
nsl	Número máximo de sesiones, 0 por default
numnom	Número máximo de NCBs, 0 por default

A. Funciones del NetBios usadas en este Sistema

Reset Adapter ***X'32' (wait)***

Descripción: Este comando da un reset a NetBios, borrando y limpiando toda la información de la sesión, borrando nombres (excepto el nombre de nodo permanente), y dando un reset al número máximo de sesiones y número máximo de NCBs potenciales restantes. Se pueden proporcionar nuevos valores para esos máximos como parte de este comando.

Si se desea alterar los números máximos de sesiones y NCBs restantes con respecto a los valores de omisión o de default, se requiere codificar los nuevos valores en los campos *nsl y numnom* del NCB antes de ejecutar el comando Reset. Si dichos campos contienen un valor de 0 cuando se ejecuta este comando, se tienen valores de default de 6 sesiones y 12 NCBs. Para mejor desempeño se recomienda tener estos valores lo más pequeños posibles.

Si una estación de trabajo tiene 2 adaptadores, este comando da un reset a NetBios de tal manera que sólo afecta al adaptador especificado en el campo *numadap* del NCB.

Cuidado: No usar este comando cuando se está corriendo bajo el programa IBM PC LAN. Este limpia las sesiones PCLP y borra los nombres PCLP, teniendo como consecuencia errores en la red.

Campos de entrada del NCB:

comando	X'32'
nsl	Número máximo de sesiones, 0 por default
numnom	Número máximo de NCBs, 0 por default

numadap 0=adaptador 1, 1=adaptador 2

Campos de salida del NCB:

codretf Código de retorno final

Código de retorno inmediato: Ninguno

Código de retorno final:

X'0'	Exito
X'3'	Comando inválido
X'23'	Número de adaptador inválido
X'40'-X'4F'	Condición de red no usual
X'50'-X'FE'	Mal funcionamiento del adaptador

Cancel X'35' (wait)

Descripción: Este comando cancela un comando de NetBios por realizarse. Se debe colocar la dirección del NCB cuyo comando pendiente va a ser cancelado en el campo *dirmen*.

Los siguientes comandos no pueden ser cancelados:

Add Name	Send Datagram
Add Group Name	Session Status
Delete Name	Reset Adapter
Otro comando Cancel	Send Broadcast Datagram
Unlink	

Campos de entrada del NCB:

comando	X'35'
dirmen	Apuntador al NCB a ser cancelado
numadap	0=adaptador 1, 1=adaptador 2

Campos de salida del NCB:

codretf	Código de retorno final

Código de retorno inmediato: Ninguno

Código de retorno final:

X'0'	Exito
X'3'	Comando inválido
X'23'	Número de adaptador inválido
X'24'	Comando ejecutado durante la cancelación
X'26'	El comando no puede ser cancelado
X'40'-X'4F'	Condición de red no usual
X'50'-X'FE'	Mal funcionamiento del adaptador

Add Name ***X'30' (wait)***
X'B0' (no-wait)

Descripción: Este comando adiciona el nombre de 16 bytes especificado en el campo *nomcloc* a la tabla de nombres como nombre único. Se recibe un error si cualquier otra estación está usando este nombre individual o de grupo. La cadena del nombre se construye adicionándole espacios a la derecha e insertando un carácter nulo en la última posición.

NetBios asigna un número al nombre que se ha adicionado y regresa este número en el campo *numnom*. Se requiere usar este número en los comandos de datagramas y en el comando Receive Any.

Campos de entrada del NCB:

comando	X'30' o X'B0'
nomcloc	Nombre que se requiere adicionar
dirpos	Apuntador a rutina de correo, o nulo

numadap	0=adaptador 1, 1=adaptador 2

Campos de salida del NCB:

numnom	Número de nombre asignado
codreti	Código de retorno inmediato
codretf	Código de retorno final

Código de retorno inmediato:

X'0'	Sin error inmediato
X'3'	Comando inválido
X'21'	Interfase ocupada
X'22'	Demasiados comandos por ejecutar
X'23'	Número de adaptador inválido
X'40'-X'4F'	Condición de red no usual
X'50'-X'FE'	Mal funcionamiento del adaptador

Código de retorno final:

X'0'	Exito
X'3'	Comando inválido
X'9'	No hay recurso disponible
X'D'	Nombre ya en uso por esta estación
X'E'	Tabla de nombres llena
X'15'	Nombre inválido
X'16'	Nombre ya en uso por otra estación
X'19'	Conflicto de nombre (error interno de NetBios)
X'21'	Interfase ocupada
X'22'	Demasiados comandos por ejecutar
X'23'	Número de adaptador inválido
X'40'-X'4F'	Condición de red no usual
X'50'-X'FE'	Mal funcionamiento del adaptador

Delete Name ***X'31' (wait)***
X'B1' (no-wait)

Descripción: Este comando borra el nombre de 16 bytes especificado en el campo *nomcloc*, de la tabla de nombres. Se recibe un error si el nombre tiene una sesión activa, pero el nombre es borrado después de que sus sesiones son cerradas. Se debe usar el comando Hang Up para terminar cualquier sesión activa antes de borrar un nombre. Si el nombre tiene cualquiera de los siguientes comandos pendientes (Listen, Receive Any, Receive Datagram o Receive Broadcast Datagram) cuando el comando Delete Name es ejecutado, el nombre es borrado pero el comando pendiente termina con un código de error de *nombre fue borrado*. Es preferible usar el comando Cancel para terminar explícitamente un comando por realizarse antes de ejecutar Delete Name.

Campos de entrada del NCB:

comando	X'31' o X'B1'
nomcloc	Nombre que se requiere borrar
dirpos	Apuntador a rutina de correo, o nulo
numadap	0=adaptador 1, 1=adaptador 2

Campos de salida del NCB:

codreti	Código de retorno inmediato
codretf	Código de retorno final

Código de retorno inmediato:

X'0'	Sin error inmediato
X'3'	Comando inválido
X'21'	Interfase ocupada
X'22'	Demasiados comandos por ejecutar
X'23'	Número de adaptador inválido
X'40'-X'4F'	Condición de red no usual
X'50'-X'FE'	Mal funcionamiento del adaptador

Código de retorno final:

X'0'	Exito
X'3'	Comando inválido
X'9'	No hay recurso disponible
X'F'	Nombre en sesiones activas
X'15'	Nombre inválido
X'21'	Interfase ocupada
X'22'	Demasiados comandos por ejecutar
X'23'	Número de adaptador inválido
X'40'-X'4F'	Condición de red no usual
X'50'-X'FE'	Mal funcionamiento del adaptador

Send Datagram ***X'20' (wait)***
X'A0' (no-wait)

Descripción: Este comando envía un datagrama a un nombre único o a un nombre de grupo. El datagrama puede ser de 1 a 512 bytes de longitud. No se requiere tener una sesión establecida antes de usar este comando; sin embargo el comando no garantiza que el datagrama será recibido.

Campos de entrada del NCB:

comando	X'20' o X'A0'
numnom	Número de nombre asignado en la adición de nombre
nomcrem	Nombre de la estación destino
dirmen	Apuntador al buffer de salida
lonmen	Tamaño del buffer de salida (1-512 bytes)
dirpos	Apuntador a rutina de correo, o nulo
numadap	0=adaptador 1, 1=adaptador 2

Campos de salida del NCB:

codreti	Código de retorno inmediato
codretf	Código de retorno final

Código de retorno inmediato:

X'0'	Sin error inmediato
X'3'	Comando inválido
X'21'	Interfase ocupada
X'22'	Demasiados comandos por ejecutar
X'23'	Número de adaptador inválido
X'40'-X'4F'	Condición de red no usual
X'50'-X'FE'	Mal funcionamiento del adaptador

Código de retorno final:

X'0'	Exito
X'1'	Longitud de buffer inválida
X'3'	Comando inválido
X'13'	Número de nombre inválido
X'19'	Conflicto de nombre
X'21'	Interfase ocupada
X'22'	Demasiados comandos por ejecutar
X'23'	Número de adaptador inválido
X'40'-X'4F'	Condición de red no usual
X'50'-X'FE'	Mal funcionamiento del adaptador

Receive Datagram ***X'21' (wait)***
X'A1' (no-wait)

Descripción: Este comando recibe un datagrama (pero no un Broadcast Datagram) enviado por un usuario a un nombre único o a un nombre de grupo en la tabla de nombres local. El datagrama puede ser de hasta 512 bytes de longitud. No se requiere tener una sesión establecida antes de ejecutar este comando. Si el campo de *numnom* es puesto a X'FF', significa que el datagrama puede ser recibido de cualquier usuario. Si el datagrama tiene una longitud más grande que la especificada en el campo *lonmen*, el sobrante del mensaje es descartado.

Campos de entrada del NCB:

comando	X'21' o X'A1'
numnom	Número de nombre o X'FF'
dirmen	Apuntador al buffer de entrada
lonmen	Tamaño del buffer de entrada(1-512 bytes)
dirpos	Apuntador a rutina de correo, o nulo
numadap	0=adaptador 1, 1=adaptador 2

Campos de salida del NCB:

nomcrem	Nombre de la otra estación
lonmen	Número actual de bytes recibidos
codreti	Código de retorno inmediato
codretf	Código de retorno final

Código de retorno inmediato:

X'0'	Sin error inmediato
X'3'	Comando inválido
X'21'	Interfase ocupada
X'22'	Demasiados comandos por ejecutar
X'23'	Número de adaptador inválido
X'40'-X'4F'	Condición de red no usual
X'50'-X'FE'	Mal funcionamiento del adaptador

Código de retorno final:

X'0'	Exito
X'1'	Longitud de buffer inválida
X'3'	Comando inválido
X'6'	Buffer muy pequeño
X'B'	Comando cancelado
X'13'	Número de nombre inválido
X'17'	Nombre borrado
X'19'	Conflicto de nombre
X'21'	Interfase ocupada
X'22'	Demasiados comandos por ejecutar
X'23'	Numero de adaptador inválido

X'40'-X'4F'	Condición de red no usual
X'50'-X'FE'	Mal funcionamiento del adaptador

Apéndice B. Bloque de Control de Red (NCB) y Códigos y Mensajes de Error del Netbios

---

Bloque de control de red (NCB)

```
type
   byte16  = array[0..15] of byte;
   byte14  = array[1..14] of byte;

netcb = record
   comando : byte;        {* comando *}
   codreti : byte;        {* código de retorno inmediato *}
   nsl     : byte;        {* número de sesión local *}
   numnom  : byte;        {* número de nombre *}
   dirmen  : pointer;     {* dirección del mensaje *}
   lonmen  : word;        {* longitud del mensaje *}
   nomcrem : byte16;      {* nombre de computadora remota *}
   nomcloc : byte16;      {* nombre de computadora local *}
   tiempor : byte;        {* tiempo fuera de recepción en inc. de 500ms
                                 *}
   tiempoe : byte;        {* tiempo fuera de envío en inc. de 500ms *}
   dirpos  : pointer;     {* dirección de rutina de correo *}
   numadap : byte;        {* número de adaptador (0 o 1) *}
   codretf : byte;        {* código de retorno final *}
   reserva : byte14;      {* área reservada *}
end;
```

Mensajes de Retorno del NetBIOS

Código	Mensaje
X'0'	'éxito'
X'1'	'longitud de buffer inválida'
X'2'	'código de retorno X'2'
X'3'	'comando inválido'
X'4'	'código de retorno X'4'
X'5'	'tiempo límite terminado'
X'6'	'buffer demasiado pequeño'
X'7'	'código de retorno X'7'
X'8'	'número de sesión inválido'
X'9'	'sin recurso'
X'A'	'sesión cerrada'
X'B'	'comando cancelado'
X'C'	'código de retorno X'C'
X'D'	'nombre local duplicado'
X'E'	'tabla de nombres llena'
X'F'	'sesión activa'
X'10'	'código de retorno X'10'
X'11'	'tabla de sesiones llena'
X'12'	'nadie escuchando'
X'13'	'número de nombre inválido'
X'14'	'sin respuesta'
X'15'	'sin nombre local'
X'16'	'nombre en uso'
X'17'	'nombre borrado'
X'18'	'terminación anormal'
X'19'	'conflicto de nombre'
X'1A'	'código de retorno X'1A'
X'1B'	'código de retorno X'1B'

X'1C'	'código de retorno X'1C'
X'1D'	'código de retorno X'1D'
X'1E'	'código de retorno X'1E'
X'1F'	'código de retorno X'1F'
X'20'	'código de retorno X'20'
X'21'	'tarjeta ocupada'
X'22'	'demasiados comandos'
X'23'	'número de tarjeta inválido'
X'24'	'cancelación realizada'
X'25'	'código de retorno X'25'
X'26'	'cancelación no posible'

Apéndice C. IBM Token-Ring Network 16/4 Adapter - 25F7367

Adaptadores de RAM compartida

Los adaptadores de RAM compartida derivan su nombre del hecho de compartir su propia RAM con la RAM del procesador del sistema. La memoria en la tarjeta adaptadora es "mapeada" dentro de un bloque de memoria no usado, arriba de la línea de 640 KB en una área de memoria más alta. Esta área es de 384KB inmediatos a los 640KB mencionados. La dirección de inicio del área de RAM compartida es determinada por el manejador de dispositivos del adaptador, a menos que el adaptador sea de tipo microcanal, en cuyo caso se determina por la configuración que se determine del diskette de referencia.

En tamaño, la RAM compartida puede ser de 8, 16, 32, o 64KB dependiendo de qué adaptador se esté usando y cómo esté configurado. Las tarjetas adaptadoras con 64KB soportan paginación de RAM, lo que permite al sistema visualizar los 64KB de memoria en la tarjeta en cuatro páginas de 16KB.

Algunas ventajas de la arquitectura de RAM compartida son: Bajos requerimientos de memoria para ambientes DOS, alta base instalada de aplicaciones y manejadores de dispositivos compatibles, control de enlace lógico en tarjeta, etc.

La principal desventaja de esta arquitectura es que cualquier movimiento de datos entre el área de RAM compartida y otras áreas de memoria del sistema deben ser realizado bajo el control directo del CPU del sistema.

Bus ISA

El bus de Industry Standard Architecture (ISA) es el original de la arquitectura bus de PC/AT. Su característica principal son los 16 bits de datos y 23 de dirección permitiendo al bus direccionar 16 MB de memoria. Su ventaja la representa su presencia y disponibilidad en el 75% de las PCs fabricadas hoy en día. Las principales desventajas de esta arquitectura son la baja velocidad del bus, la falta de provisiones para configurar automáticamente adaptadores y resolver conflictos de recursos, etc.

El **IBM 16/4 Adapter** es un adaptador de LAN token-ring de alto desempeño, soporta la operación a 4 o 16 Mbps sobre par trenzado no cubierto o dispositivos de cableado IBM, es el adaptador de elección para sistemas de bus ISA y operando en ambiente DOS

Tabla de Direcciones de ROM - Switches 1-6

Dirección de ROM	Número de Switch					
	1	2	3	4	5	6
C0000	off	on	on	on	on	on
C2000	off	on	on	on	on	off
C4000	off	on	on	on	off	on
C6000	off	on	on	on	off	off
C8000	off	on	on	off	on	on
CA000	off	on	on	off	on	off
CC000	off	on	on	off	off	on
CE000	off	on	on	off	off	off
D0000	off	on	off	on	on	on
D2000	off	on	off	on	on	off
D4000	off	on	off	on	off	on
D6000	off	on	off	on	off	off
D8000	off	on	off	off	on	on

DA000	off	on	off	off	on	off
DC000	off	on	off	off	off	on
DE000	off	on	off	off	off	off

Tabla de Niveles de Interrupción - Switches 7-8

Nivel de Interrupción	Número de Switch	
	7	8
2	on	on
3	on	off
6	off	on
7	off	off

Tabla de Selección de Adaptador, Primario o Alterno, Switch 9

Primaria o Alterna	Número de Switch
	9
Primaria	off
Alterna	on

Tabla de Selección de Tamaño de RAM Compartido, Switches 10-11

Tamaño de RAM Compartido	Número de Switch	
	10	11
8KB	on	on
16KB	off	on
32KB	on	off
64KB	off	off

Tabla de Selección de Rango de Velocidad, Switch 12

16 Mbps o 4 Mbps	Número de Switch
	12
16 Mbps	off
4 Mbps	on

Apéndice D. Guía del Usuario

La aplicación de correo electrónico tratada en este documento está compuesta de 2 programas básicamente: el programa de control de correo y un editor de texto. Dicho programa de correo es arrancado ya sea para el envío de memos o bien para la lectura del correo recibido; a continuación se presentan las imágenes que presenta el sistema durante su ejecución con una breve explicación de su funcionamiento.

Adición de Nombre Local

```
Nombre local (máximo 15 caracteres) ==>  m2
```

Al arrancar el sistema, éste requiere de un nombre local que identificará a la estación de trabajo en turno.

Este proceso toma algunos segundos, al terminar éste, nuestra estación de trabajo es reconocida dentro de toda la red.

```
Correo Electrónico

► Enviar Correo
  Buzón de Entrada
  Buzón de Salida
  Salida del Sistema
```

Use: ↑ ↓

Este es el menú principal del sistema, indica las cantidades de memos existentes en los buzones de entrada y salida para la estación de trabajo en turno, y presenta las funciones a elegir de acuerdo a nuestra seleccción deseada (enviar correo, control del buzón de entrada, control del buzón de salida y salida del sistema); dichas opciones son elegidas con el uso de las flechas del teclado y presionando ENTER.

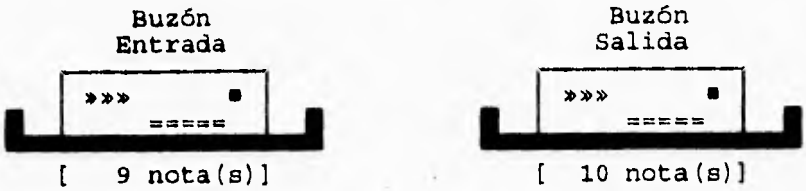
```
Manejo de Archivos

▶ Enviar Archivo Existente
  Editar Archivo Nuevo
```

Use: ↑ ↓ ESC

Al haber seleccionado la opción de envío de correo, se tienen dos nuevas opciones a elegir: el envío de un memo existente o bien la creación de uno nuevo. Se presenta una opción de abandono (ESC) con la cual regresaríamos a nuestro menú principal si así se desea.

```
=Envío de Correo=
Dirigido A (máximo 15 caracteres) ==> m1
Asunto ==> Gold Card
Archivo ==> c:\nota.dat
```

Al decidir enviar un memo, se debe proporcionar el nombre local de la estación de trabajo a la cual va dirigido, el asunto a tratar y el nombre del archivo representando al memo que queremos transmitir a través de la red.




```
Correo Electrónico

  Enviar Correo
▶ Buzón de Entrada
  Buzón de Salida
  Salida del Sistema
```

Use: ↑ ↓

A continuación se muestra otra posible secuencia de actividad seleccionada, correpondiente a la lectura del correo de entrada (recibido por la estación de trabajo en turno).

3-12-1994

Buzón de Entrada

Título:	Origen:	Pag : 1
▸ nota	M1	29- 5-1993
nota1	M1	29- 5-1993
nota2	M1	5- 6-1993
nota3	M1	5- 6-1993
nota4	M1	5- 6-1993
noat5	M1	5- 6-1993
nota6	M1	5- 6-1993
nota7	M1	5- 6-1993
nota8	M1	5- 6-1993

Use: V_er nota B_orrar nota I_mprimir nota Home PgDn ↑ ↓ Esc

Se puede apreciar el contenido del buzón de entrada, se muestran el título del memo, quién lo envió y la fecha de envio/recepción. El memo seleccionado puede ser leido, borrado del buzón o bien impreso si asi se desea.

Origen: M1 F/H (recepción): 29-5-1993 16:50:45 3-12-1994

nota

Pag : 1

Requisitos para obtener THE GOLD CARD

A fin de agilizar su trámite, le recomendamos anexar copias de los siguientes documentos:

* Comprobación de ingresos (comprobante de nómina, contratos de valores, declaración de impuestos).
* Comprobación domiciliaria (boleta predial o recibo de renta a su nombre).
* Identificación (licencia o pasaporte).

Usted no necesita pagar ningún depósito para obtener The Gold Card.
Tampoco pagará depósito por las tarjetas complementarias que otorgue a sus seres queridos.
Unicamente requerirá cubrir la cuota de inscripción de $35.00 U.S. y la cuota anual de $150.00 U.S.
Asimismo, las tarjetas complementarias que usted solicite tendrán un cargo

Use: PgDn PgUp Esc

Lectura de un memo seleccionado; se muestran algunos datos relacionados con el envío y recepción del mismo, y se tiene un control de páginas para visualizarlo. Existe la opción de escape también, con lo que se regresaría a la pantalla anterior del sistema.

Origen: M1 F/H (recepción): 29-5-1993 16:50:45 3-12-1994

nota

Pag : 2

de $95.00 U.S. cada una.
Las cantidades antes mencionadas le serán cargadas en su primer estado de cuenta y se les agregará el importe correspondiente al I.V.A.
Llene la solicitud y envíela hoy mismo para disfrutar cuanto antes los privilegios exclusivos The Gold Card o bien, llámenos al 598-71-22 para que uno de nuestros representantes le atienda personalmente.
Del interior del país, llame sin costo al 91(800)555.→

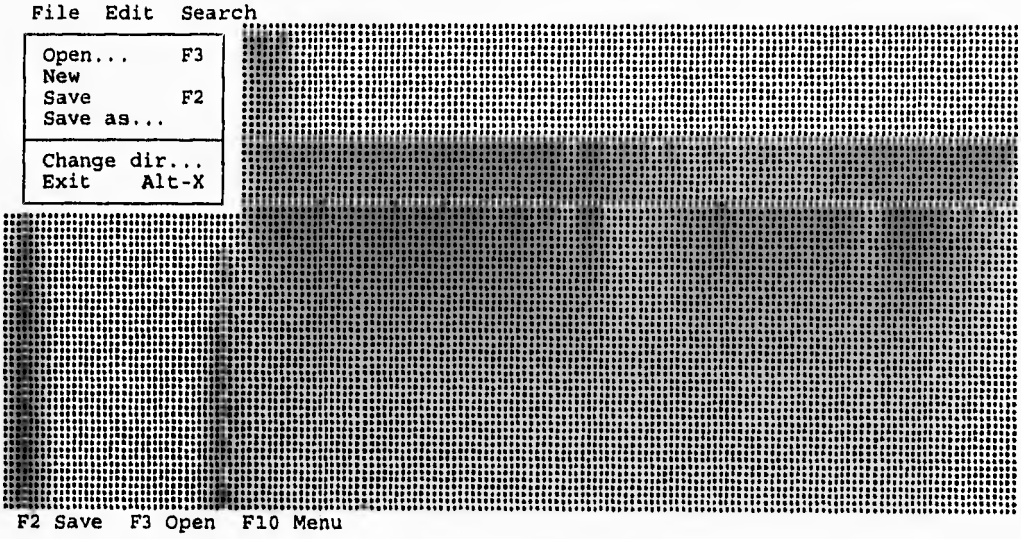
Use: PgDn PgUp Esc

Continuación del memo mostrado...

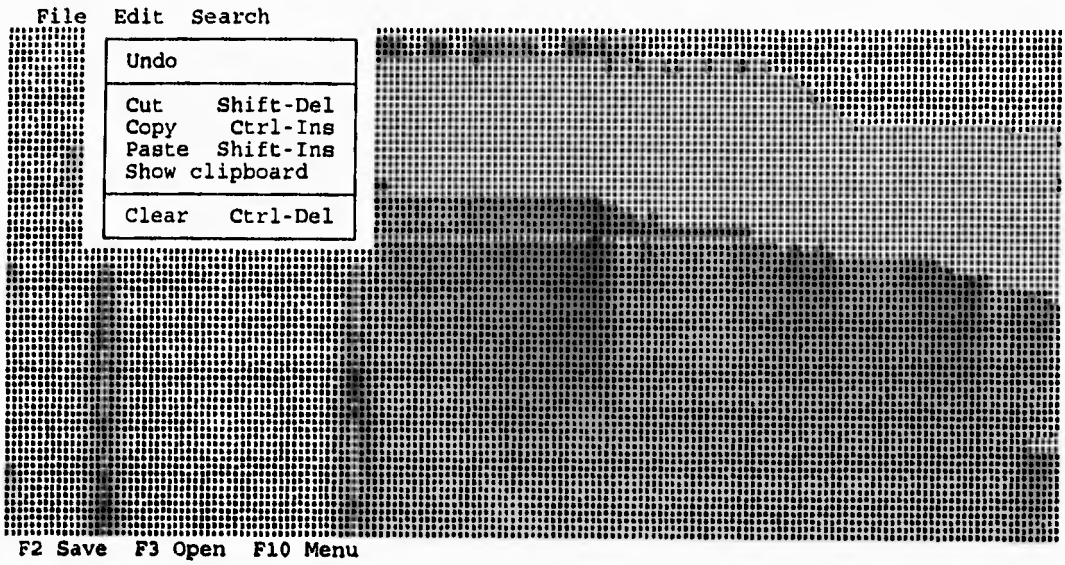
```
Manejo de Archivos

  Enviar Archivo Existente
▶ Editar Archivo Nuevo
```

Use: ↑ ↓ ESC

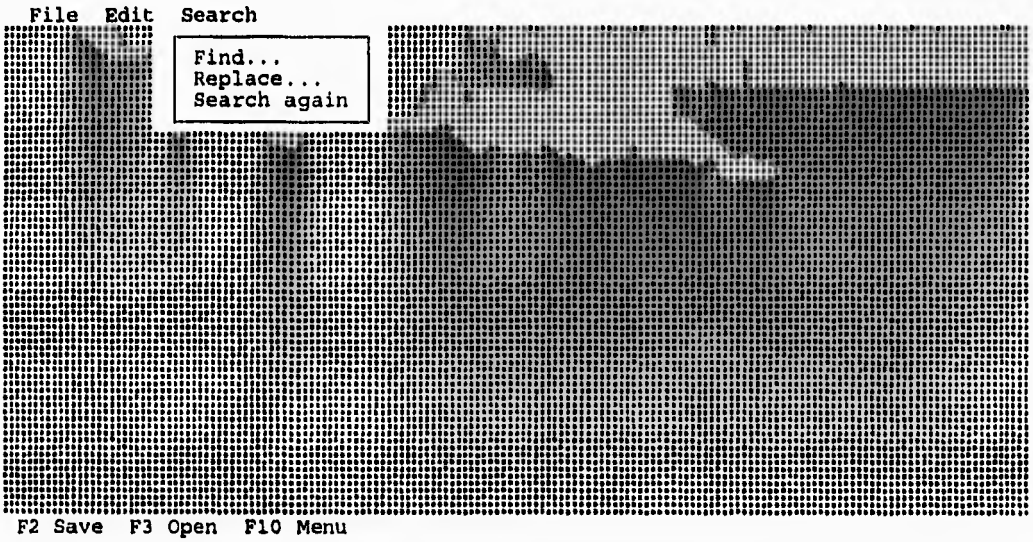
En caso de seleccionar el control del buzón de salida, (correo enviado por la estación de trabajo en turno) las imágenes que se muestran son muy similares a las ya mostradas en el control del buzón de entrada. Toca en turno mostrar la opción de edición de un memo.



El editor manejado por este sistema es lo suficientemente completo para la edición de los memos a transmitir. Está constituido por una serie de ventanas que muestran las posibles funciones a realizar. File...


File Edit Search
Undo
Cut Shift-Del
Copy Ctrl-Ins
Paste Shift-Ins
Show clipboard
Clear Ctrl-Del
F2 Save F3 Open F10 Menu

Edit...


File Edit Search
Find...
Replace...
Search again
F2 Save F3 Open F10 Menu

Search...

```
Correo Electrónico

  Enviar Correo
  Buzón de Entrada
  Buzón de Salida
▶ Salida del Sistema
```

Use: ↑ ↓

El abandono del sistema se logra seleccionando la opción salida del sistema del menú principal. Por supuesto que la mejor manera de conocer las funciones del sistema así como la forma de manejarlo es el uso del mismo. Cabe mencionar que el sistema cuenta con algunos elementos no visibles en este apéndice como son el manejo de mensajes de control de envio / recepción, mensajes de error, verificación de datos, etc.

GLOSARIO

Amplificador.- Dispositivo independiente de energia que produce como salida una reproducción incrementada de las caracteristicas esenciales de una señal de entrada. Por ejemplo, un amplificador puede recibir una señal débil a través de una linea de transmisión e incrementarla (en voltaje o corriente), sin modificar su frecuencia ni forma de onda. Compensa la atenuación (pérdida de potencia) provocada por la degradación inherente a los sistemas de transferencia de energía.

Bridge (puente).- Unidad funcional que conecta dos redes de área local (LANs) usando el mismo procedimiento de control de enlace lógico pero con la posibilidad de usar diferentes procedimientos de control de acceso al medio. Un bridge conecta redes o sistemas de arquitecturas iguales o similares.

CMS.- Sistema de mensajes basado en computadoras.

Concentrador.- Unidad funcional que mejora la eficiencia de transmisión de datos al permitir a las terminales competir por los canales de transmisión y compartirlos.

Datagrama.- Paquete de información independiente que contiene datos suficientes de ruteo entre las estaciones transmisora y receptora en una red.

DMA.- Memoria de acceso directo, proporciona un método para transferir datos entre un dispositivo y la memoria sin ayuda del procesador de la computadora.

Gateway (compuerta).- Unidad funcional que conecta dos redes de computadoras de diferentes arquitecturas.

Interfase.- Grupo de reglas que rigen el intercambio de información entre capas contenidas en una misma computadora (nodo o Host).

IPX.- Internetwork Packet Exchange es un protocolo de alto nivel de Novell, es una adaptación muy estrecha de XNS de XEROX; soporta solo el uso de datagramas. Es más rápido que el protocolo orientado a sesiones (SPX) de Novell también.

ISO.- Organización internacional para estandarización.

Manchester (código).- Código de transmisión en el cual cada bit es codificado como una señal de dos segmentos con un cambio de polaridad ya sea al tiempo de bit o al tiempo de medio bit. La transición a tiempo de bit representa un 0, no transición a tiempo de bit representa un 1. Este esquema de codificación permite una transmisión/recepción más simple y con tiempos de retardo más pequeños por estación que los logrados con códigos de bloque.

MSAU o MAU.- Unidad de acceso múltiple, mantiene la pista de qué estaciones de trabajo en la red son vecinas y cúal está adelante y cuál atrás en secuencia. Lleva a cabo control de red.

NetBios.- Network Basic Input/Output System. Interfase al sistema operativo para programas de aplicación usados en computadoras personales ligadas a una red Token Ring.

OSI.- Open Systems Interconnection.

Protocolo.- Serie de reglas que rigen el intercambio de información entre capas, del mismo nivel, contenidas en computadoras distintas.

Server.- En una red de área local, una estación de trabajo que provee de facilidades a otra estación en la red, por ejemplo, servidor de archivos, de correo, de impresión, etc.

Token.- En una red de área local, el símbolo de autoridad (patrón de bits) pasado entre estaciones de trabajo para indicar que la estación está temporalmente en control del medio de transmisión.

Transceiver.- Dispositivo terminal que puede transmitir y recibir tráfico de información.

TWX.- Teletypewriter Exchange Service.